A. Department goals and objectives are linked to the university and college mission and strategic priorities, and to their strategy for improving their position within the discipline. (1 page maximum)

1. What is the department's mission and is it clearly aligned with the university and college mission and direction?

The department strives to offer quality undergraduate and graduate courses / programs; places a high priority on quality research and effective teaching; seeks to attract highly-qualified students; and strives to attract and nurture high-caliber faculty. This mission is consistent with university goals of ensuring student success, enhancing academic excellence, expanding breakthrough research, and developing / recognizing our people.

2. How does the department's mission relate to curriculum; enrollments; faculty teaching; research/professional/creative activity and outreach? Is it aligned with the college's strategic priorities?

The departmental mission aligns with university and college strategic priorities, although the department is now in financial jeopardy under KSU's new RCM budget model due to developmental growth in faculty numbers under state direction to grow a doctoral program.

3. How does the department contribute to university-wide curricular needs through general education and service instruction?

The department was limited to a single Liberal Education Course (*CS 10051 Introduction to Computer Science*), but may have an opportunity to offer more courses under the new Kent Core curriculum. However, unlike many universities, KSU does not have a university-wide computing requirement, and does not have a large number of engineering majors taking CS courses.

4. How does the department promote diversity?

The department has an active NSF S-STEM grant for broadening participation, and a diverse set of faculty, two of whom are women, and outreach activities to women and other minority students in computing.

5. What is the current standing of the department's programs within the discipline? What goals does the program have in terms of its standing within the University, region, state or nation? (To establish benchmarks, you may wish to use information from the discipline's professional organizations, national rankings, accreditation standards, other comparable programs.)

The National Academies have not updated their rankings of Computer Science programs since 1985. At that time, KSU's Computer Science doctoral program was just beginning, and the program was housed within the Mathematical Sciences Department. The next ranking was initiated in late 2007 but so far has not been released.

B. **Curriculum is relevant, rigorous, current and coherent**. (4 pages maximum)
(Check appendices listing for data that needs to be included)

1. How does the department determine curricular content? How does the curriculum relate to current existing standards, if any, of the discipline?

The department determines its core undergraduate curriculum based upon the latest recommendations of the primary Computer Science professional societies: ACM (Association of Computing Machinery) and IEEE (Institute of Electrical and Electronics Engineers). The curriculum outline from these bodies changes every four/five years, but the Computer Science field and market demand changes even faster. To keep up with these changes, instructors also introduce 'selected topics courses' in fields that are currently needed in the job market, and many of these courses become regular catalog courses once the market need and student demand is proven.

The graduate program requires students to expand their broad knowledge in at least three different areas of computer science. Each area consists of at least one standard course in the area plus several selected topics courses based on research faculty interests. The depth is provided through identifying a concentration area that focuses in one particular subject area and by taking graduate courses, research hours and writing a thesis/dissertation in that area. For the Master's with non-thesis option, students take advanced graduate courses or selected topic courses related to their concentration area followed by an applied industrial project.

2. What internal or external measures of review are employed to ensure that the curriculum is relevant and up-to-date? When were these measures last employed for this curriculum?

We have internal measures and external measures in place for both undergraduate and graduate programs. All four graduate degree programs (MA, MS thesis option, MS non-thesis option and PhD) follow these measures. Internal measures include: evaluation of newly-proposed courses by the departmental curriculum committee; student evaluation of courses and their preparation in Computer Science; an exit survey of graduating students; a plan of study for graduate degree programs / university-wide GPS (Graduation Planning System) for undergraduates; and yearly self-analysis of student evaluation forms by the department under the university AQIP (Academic Quality Improvement Project) program. The external measures for the core and required courses follow the ACM and IEEE guidelines, results of polls from local industries about the fields that are needed, and surveys of alumni. Except for the GPS, all the measures were established in the year 2004 or before, while the GPS was established in 2008. However, a detailed departmental equivalent of the GPS based upon course dependencies has been in place since 1992.

The curriculum committee is composed of tenure track faculty including many senior professors. This committee evaluates the merit of proposed courses before faculty members can teach them, with the courses evaluated on the basis of content and market need (as well as current research areas in the case of graduate courses). Student evaluation of a course is a standard university procedure. The departmental curriculum committee regularly undertakes a self-analysis based upon comments in the student

evaluation forms, and prepares a report for the university about the issues and the measures that have been taken to correct and improve the program.

We also give an exit survey to every graduating student to survey their preparedness in their chosen field, the interaction they had with their instructors and the quality of overall instruction. The entire undergraduate and graduate curriculum is reviewed by external reviewers every five years.

3. Are the curricular offerings structured in a logical, sequential and coherent manner? Is there an appropriate balance between depth and breadth? Does the sequence build from simple to complex? Are basic threads woven throughout?

The department has created detailed descriptions of the logical structure, sequence and coherency for the courses at the undergraduate level as well as the graduate level. In the undergraduate curriculum we have multiple layers of courses: introductory foundational courses (*CS 10051 Introduction to Computer Science*, *CS 10061 Introduction to Computer Programming*, *CS 23021 Computer Science I Programming and Problem Solving*), core foundational courses (*CS 33001 Computer Science II Data Structure and Abstractions*, *CS 35101 Computer Architecture* and *CS 33211 Operating Systems*), upper division required courses, elective courses, and selected topics courses. There is a detailed dependency chart for the courses, first created in 1992, that is continuously updated on a yearly basis by the departmental curriculum committee. This dependency chart is based upon a progression from simple to complex courses. Beginning in 2008, KSU initiated a university wide GPS (Graduation Planning System) Roadmap that does something similar, guiding students through their courses in different semesters based upon the dependency chart. Further, the GPS guides students to graduate in four years, even if they have some backlog in courses compared to the ideal progression.

Introductory foundational courses have coordinators whose task is to provide synchronization at different levels: between the sections and between the lectures and laboratory sessions. These coordinators synchronize the material on a weekly basis.

The graduate course offerings are structured in several coherent research clusters, according to the research interest/activity of the faculty. Students expand their breadth knowledge by taking courses from at least three diverse clusters, and concentrate on depth by taking several research and selected topics in their area of concentration.

4. If consistent with the program mission, does the curriculum adequately prepare students for further study or employment?

Our undergraduate program has both BS and BS/MS degrees, and our graduate program has four different types of degrees - MA, MS with thesis option, MS with non-thesis option, and PhD. Each degree has a different focus.

The BS degree prepares students both for the workforce as well as Masters study. In addition to required coursework, we also have an optional internship program (*CS 33192 Internship in Computer Science*) in which students do practical training in industrial settings to prepare them for the transition to industry. The BS/MS program prepares students who want to go for further study in a shorter period. Under this

program, students are allowed to take graduate level courses in their senior years so that they can finish their MS degree faster.

The graduate program prepares students for multiple career goals. Students who plan non-academic careers or want to enhance their skills beyond the BS level can earn a Master's degree with a non-thesis option. Students who seek positions in research and development (R&D) or plan to proceed into the PhD program can earn a Master's degree with the thesis option. Students who plan to become professional scholars, college and university teachers, or independent research workers in private, industrial or government research institutions can earn a PhD degree.

5. What do graduates of the program say about their preparation?

Our alumni statistics clearly show that our BS and BS/MS graduates are well-placed in the information technology industry. The graduates of our MA, MS and PhD programs are also well-placed in information technology industries, national research laboratories, corporate research centers, and as college and university faculty members. Our graduate students often publish in international conferences or journals before taking up an employment.

We survey our alumni through questionnaires and more recently at an alumni reunion. Our alumni survey results have been quite positive. Through the exit survey we conducted during the past four years, those who responded expressed their satisfaction with their experiences at Kent. Some made some positive comments about course offerings, seminars, and the colloquium series. Approximately 43% graduates strongly agree that their Kent State education prepared them well, and another 50% agree that their Kent State education prepared them well. A recently-graduated student, who is currently an assistant professor in another institution wrote, "The graduate program at KSU improves students' research capabilities and problem solving skills. The courses offered by the program enrich students' experience and prepare them for undertaking a wide range of problems in an efficient manner." Another alumnus wrote, "My courses served me well in giving me content knowledge and applicable skills. Significantly, my courses, research experiences, and dissertation writing gave me exactly the grounding I needed to be effective both as a teacher for undergraduate computer science and a mentor for undergraduate research." Yet another alumnus wrote, "The time, effort, education standard, and support have taught me things that I would never have achieved if I had gone somewhere else." Dr. Liao wrote, "The computer science program at Kent State University prepares me with fundamental knowledge in computer science so that I could adjust myself in this ever-changing computing area. It also offers me a series of seminars and all graduate students get chance to give research talks. Moreover, the opportunity to be a teaching assistant and/or instructor has been very helpful for my career as a faculty member." Dr. Oleg Komogortsev wrote, "The program prepared me to conduct independent research, establish and run a successful research laboratory. The program allowed to develop and hone writing and oral presentational skills necessary for a successful academic career.

6. What do employers of program graduates say about their preparation?

Generally, we have a very high rate of placement for graduates of our BS, BS/MS, MS and PhD programs. Many employers of our BS graduates have hired more than one of our graduates. Most of our students receive job offers before they graduate. In one instance, the employer of a recent graduate wrote, “I am very pleased with his performance in and out of classrooms. In general I found him dedicated, knowledgeable, sincere, and personable.”

7. What are the educational, professional, career, and/or life goals of the students who choose to major in this program?

Computer Science students have a wide-variety of goals and career plans. BS graduates chose information technology industries, self-employment, and higher studies as their goals/professions. MS graduates chose industrial employment, self-employment, corporate research centers, and PhD study. PhD graduates preferred teaching and academics, corporate research centers, and national research laboratories.

8. In what ways does the program contribute to the education of students in terms of general knowledge, critical thinking capacity and other essential cognitive skills?

Kent State University places significant emphasis on students interfacing with other disciplines outside their major, such as physical sciences, logic and mathematics, media and communications, and social studies, and on the study of a foreign language and being exposed to cultural diversity. Our BS students complete 121 credit hours, including 36 credit hours of liberal education, 15 credit hours of mathematics, and 70 credit hours of Computer Science.

Liberal education courses provide general knowledge. The required mathematics courses are two calculus courses and a linear algebra course and are necessary for modeling and problem solving.

Critical thinking and essential cognitive skills are part of the Computer Science program, both at undergraduate level and graduate level, through course projects and labs. Most of the elective courses at the undergraduate level and almost all graduate-level courses have at least one individual or team project, in which students select a challenging project, perform a literature search, formulate a solution, and implement the solution. In addition, at the undergraduate level we have a Capstone Project course in which students undertake a realistic, industrial-level, team project.

C. Faculty quality and productivity.(1 page maximum)

(Check appendices listing for data that needs to be included)

1. Do faculty possess the appropriate background experience and credentials?

All faculty members have a PhD in Computer Science or a related area. Some have industrial experience, and many are recognized as ACM and IEEE senior members or fellows. Faculty research was supported by NSF, DARPA, NASA, CISCO, OBR, etc.

2. Are faculty current in relation to the knowledge base and content of the discipline and curricular offerings? What professional development activities have faculty participated in over the past three years?

Our curriculum is updated regularly as we introduce new courses to reflect current topics in this fast-changing discipline. Our faculty publish in journals and conferences, and present regularly at national and international conferences. They visit companies and academic institutions to expand their knowledge and establish research cooperation.

3. Are the program expectations for faculty involvement in the scholarships of discovery, application, integration and teaching, and public service/academic outreach activities appropriate; and how are these expectations met? Are these expectations consistent with university and college policies regarding teaching assignments, faculty excellence allocations, and other aspects of faculty roles and rewards? (See appendix for vita instructions)

In addition to tenure and promotion evaluations, faculty are evaluated every 5 years for graduate faculty status and for merit based on research, teaching and service. Our expectations are consistent with university and college policies. Over the last 5 years, our faculty published 4 books, 6 book chapters, 66 journal and over 280 refereed conference articles. They gave over 130 conference presentations and 30 invited/colloquium talks. 19 PhD dissertations and 87 MS theses were successfully defended. 102 proposals were submitted to funding agencies and 24 were funded, bringing in about 5.4M dollars.

4. In what way is faculty professional development and growth fostered?

All pre-tenure tenure-track-faculty have received one course release per year, as has each faculty member with grant funding. After seven years, each faculty is usually granted a sabbatical The department funded faculty travel to over 130 conferences/workshops during this time period.

5. In what ways do the department faculty lend their professional expertise to off-campus constituencies?

Many faculty are involved in reviewing papers for CS journals and conferences, serving in professional societies and on program committees (as chairs/members) of conferences, serving as panelists in NSF panels, consulting local industries, etc.

D. Administrative quality and support (1 page maximum).

(Check appendices listing for data that needs to be included)

1. What is the effectiveness of department/program leadership? (department chair, director, coordinator, lead faculty, etc.)

The current Department Chair has served since 2005, first for one year as Interim Chair, and now in the fourth year of a four year term. He was reviewed during the Spring 2009 semester, and the departmental review committee recommended him for reappointment. The Dean has since announced his intention to reappoint him for another term. The department has 5 additional rotating administrative positions — Assistant Chair, Graduate Coordinator, Curriculum Coordinator, and System Coordinator — and the faculty assigned to those positions are generally effective in their roles.

2. Is the current leadership model appropriate for the type of program?

The roles and duties of the departmental administrative Coordinators, along with their appropriate committees, were reviewed recently during the 2008 departmental Handbook revision. Minor changes were made, but in general the department was satisfied with that model. At that time, the faculty also reaffirmed their desire to have a Faculty Advisory Committee (FAC) comprised of the entire faculty, rather than elected representatives.

3. How effective is administrative leadership (chair, dean, assistant/associate dean) in supporting the program?

The past five years have been a period of transition for much of the university. A new President was hired in 2006, and a new Provost was hired the following year. After the retirement of the Dean of Arts and Sciences in 2004, one Interim Dean led the college the following year, and another led for the next three years, with a permanent Dean hired only in 2008. In Research and Graduate Studies, the VP Research and Dean of Graduate Studies was away for a year, and that position was later divided, with several Interim Deans selected to lead Graduate Studies. These many transitions have often resulted in policy shifts and mixed messages.

4. Does the administrative leadership provide for and participate in goal setting, decision-making and resource allotment to build and sustain the program?

The Department Chair has regular meetings of faculty and staff once per month, and usually schedules a one-day onsite “retreat” each year to consider an issue in depth. Major shifts in undergraduate and graduate policy are proposed by the department’s very active Curriculum and Graduate Studies Committees, which typically meet 1-2 times per month, and bring those proposals to the faculty and Chair for approval. Most faculty participate in this democratic process of governance.

E. Teaching/learning environments that facilitate student success. (4 pages maximum) (Check appendices listing for data that needs to be included)

1. What is the program looking for in its students? What kind of students is the program well suited to serve? How does the program define "quality" in terms of admission to the program (when relevant)?

The Computer Science Department at Kent State University is looking for well-motivated, smart individuals, who are willing to learn. The Computer Science program is for those individuals who love to solve puzzles, invent new ways of using computers, or exchange theories about new ideas; who enjoy finding better ways to get things done using computers, are interested in understanding how computers work and how they can make businesses work better and more securely; who want to be involved in building the next generation of computers, software tools, networks, mobile communications, mobile phones, tiny media players, and even high-tech clothing, or creating new and more advanced medical tools; and those who enjoy being a part of the evolution of technology in this fast-paced, ever-changing discipline.

The Computer Science Department has approximately 100 graduate students and approximately 250 undergraduate students. The department covers nearly every aspect of computer science, including: database systems, data mining, distributed systems, parallel processing, interconnection networks, graphics, scientific visualization, image processing, software engineering, algorithms and computational theory, computational intelligence, communication networks, wireless and sensor networks, and multimedia networking, web-based applications, symbolic computation, multimedia languages, information security, and network security.

Kent State University's freshman admission policy differentiates among students with varying degrees of preparation for college studies. The students most likely to be admitted and succeed are those who have graduated with at least 16 units of the recommended college preparatory curriculum in high school, who have achieved a cumulative high school grade point average of 2.5 or higher (on a 4.0 scale), and whose composite ACT score is 21 or better (980 combined critical reading + math SAT score). High school computer science and mathematics classes are the gateway to studying Computer Science at Kent State University.

Students entering our CS Master's program are generally expected to have a Bachelor's degree in computer science or a related discipline, with a grade point average of 3.0 out of 4.0. The GRE is not required but recommended for admission and used as one of the criteria for financial aid consideration. A minimum TOEFL score of 527 (paper base), 197 (computer base), or 71 (IBT) is required for International applicants.

We expect applicants to our graduate program to have mathematical proficiency through Linear Algebra and Discrete Mathematics, and to have taken computer science course work that minimally includes Data Structures, Structure of Programming Languages, Operating Systems, and Computer Architecture.

2. To what extent does the program have articulated learning outcomes (content and skills) for students? By what means are these outcomes measured? (Assessment activities may include surveys, reports from internship or cooperative education

supervisors, student evaluations of instruction or other aspects of the program, focus groups, the review of capstone courses or portfolios of student work, results on standardized tests, licensure and certification examination pass rates, accreditation results, etc.) Do most students achieve them?

The Department of Computer Science at Kent State University offers courses and curriculum leading to the Bachelor of Science degree in Computer Science. This program meets the needs of both the students wishing to pursue a career immediately upon graduation and those students planning for graduate studies in computer science. The curriculum is aligned with the most recent ACM/IEEE-CS curriculum guidelines and as such offers a good balance of practical skills and the underlying knowledge necessary for adapting to the ever- and often-changing needs of technology.

The Bachelor of Science Program in Computer Science at Kent State University is designed to instill in the student a system-level perspective that transcends the implementation details of individual software components, an appreciation of the structure of such software systems, and an understanding of the processes involved in their construction.

Graduates of the program understand not only the theoretical underpinnings of the discipline but also how that theory influences and is applied in practice. The program emphasizes the key themes of abstraction, complexity, and evolutionary change as applied to the development and analysis of software. The program provides a solid foundation that allows the students to maintain their skills in the rapidly evolving field. Students of the Computer Science program develop a wide range of capabilities and skills. Here is a list of cognitive capacities and skills relating to Computer Science that represent the learning objectives of the program: (a) *Knowledge and understanding:* demonstrate knowledge and understanding of essential facts, concepts, principles, and theories relating to Computer Science and software applications; (b) *Modeling:* use such knowledge and understanding in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices; (c) *Requirements:* identify and analyze criteria and specifications appropriate to specific problems, and plan strategies for their solution; (d) *Critical evaluation and testing:* analyze the extent to which a computer-based system meets the criteria defined for its current use and future development; (e) *Methods and tools:* deploy appropriate theory, practices, and tools for the specification, design, implementation, and evaluation of computer-based systems; (f) *Professional responsibility:* recognize and be guided by the social, professional, and ethical issues involved in the use of computer technology; (g) *Communication:* make succinct presentations about technical problems and their solutions; (h) *Teamwork:* be able to work effectively as a member of a software development team.

The CS graduate program is designed to provide prospective and current graduate students with an educational and research environment that fosters personal and intellectual growth, and allows them to accomplish their academic goals and develop a career path.

The program offers four graduate degree programs. (1) The Combined Baccalaureate and Master's degree program is a fast-track to graduate education. It is designed for students without prior undergraduate degrees in computer science. (2) The

Master of Science degree program has two concentration tracks: a non-thesis Master's degree program that supports the needs of persons seeking graduate education for entry into non-academic careers or to enhance and expand their career options; a thesis Master's degree program that fosters close collaboration between research and academic needs and is designed for students who seek academic careers or positions in research and development (R&D). While the latter requires writing a thesis, the former does not require writing a thesis, but the student is required to complete a project and take an oral examination. (3) The Master of Arts degree program provides students with the technical knowledge and skills necessary for success within the information and high technology industries. The program emphasizes breadth of knowledge in advanced computer science topics to augment the student's baccalaureate degree. (4) The PhD degree program promotes research, discovery and integration. It is designed for students interested in becoming professional scholars, college and university teachers, or independent research workers in private, industrial or government research institutions.

We use a number of assessment activities to measure learning outcomes, including:

a. CS II Exit survey, CS Field Test results;
b. Capstone projects, honor thesis for undergraduates;
c. Internship (47 undergraduate students had internship over the past five years);
d. Student Grades;
e. Student evaluations of courses and instructors;
f. Thesis, research projects and OPTs (Optional Professional Training) for master's students;
g. Publications, presentations at conferences, awards for graduate students; etc. (See appendix for a representative listing of student publications, a representative listing of student presentations, and a representative listing of externally awarded student prizes and awards over the past five years).

Most of our students achieve the required skills. Our recent BS, MS and PhD graduates easily find jobs.

3. How are program expectations communicated to students? Are students kept informed of their progress in meeting intended program outcomes?

The program expectations are extensively described on our websites:
http://www.kent.edu/CAS/CS/undergraduate/index.cfm and
http://www.kent.edu/CAS/CS/graduate/index.cfm.

The CS II Exit survey has indicated that students use and visit our web site regularly. They are informed about the program and which course(s) they should take next. At the beginning of each semester, our Graduate Coordinator meets with graduate students in our Masters/Doctoral seminar to discuss expectations and to keep students informed. An academic advisor is assigned to each graduate student during the first week of the semester. Later in the program, a research advisor is chosen by a graduate student. At the undergraduate level, there are 4 tenure track faculty advisors available to assist students. Honors students who write a thesis also select a research advisor to guide them. All of our core undergraduate courses have a faculty coordinator. All of our undergraduate/graduate courses have syllabi available on-line and course web-pages maintained by the instructor. Many faculty post their lecture on those web-pages. The

results of all tests, exams, and homework and project assignments are conveyed in a timely manner to students, and they are kept informed of their progress in meeting intended program outcomes.

4. How is assessment of student learning outcomes used in reviewing and modifying program curriculum, advising, and other program elements, and in evaluating faculty?

Our curriculum is modified, enhanced and updated regularly as we introduce new courses to reflect current topics in this fast changing discipline. Every year the CS I Entry survey, CS II Entry survey, CS II Exit survey, and CS Field Test results, are used in reviewing and modifying our curriculum. The last review of undergraduate core courses was done in Spring 2009 by a subcommittee of the Curriculum Committee. *CS 10051 Introduction to Computer Science* was recently substantially revised to attract more CS Majors, and to synchronize its lecture and lab components. We recently increased the number of tutors for lower-level classes. These new concentrations were developed: Web Design, Game Programming, and Information Security. We take student evaluations into account when evaluating faculty teaching for merit.

5. In what ways does the program evaluate student success following graduation and the program's contribution to that success?

We evaluate student success following graduation and the program's contribution to that success by getting feedback from alumni and their employers, by keeping statistics on placement of our alumni in academia, industry, government institutions, etc. Many of our alumni work in industry. They are employed by Intel Corp., The Goodyear Tire & Rubber Comp., Hyland Software, IBM, Oak Ridge National Laboratory, Synopsys Inc., Alcatel-Lucent, IBM India, Xerox Corp., to name a few of them. A number of our graduates are working in academia. They are employed by Missouri University of Science and Technology, Cleveland State University, Walsh University, The University of Akron, Louisiana Tech. University, Texas A&M University, Texas State University, Wayne State University, Western Washington University, to name few of them. In October 2008, the department held its 2008 KSU CS Graduate Alumni Reunion workshop. A large number of graduates participated in the workshop and gave feedback. Our recent BS, MS and PhD graduates easily find jobs, with many staying in Ohio to work at local companies, such as Progressive Insurance, NASA Lewis, Goodyear, Cisco Systems, Hitachi Medical, Davey Tree, Fed Ex, Ernst & Young, Philips Medical, Sherwin Williams, National Bank, Key Corp, Procter and Gamble, Lexis-Nexis, Bell and Howell, etc.

F. **Resources. (1 page maximum)** (Check appendices listing for data that needs included)

1. In what way does the department maximize the use of its personnel (faculty and staff) resources?

Faculty teach most undergraduate courses, and generally teach one graduate elective per year. Sabbaticals and grant releases are covered by the department. Office staff assist with class and room scheduling; manage the departmental, graduate, and grant budgets; and assist with reports. System Staff serve both the Computer Science and Math Departments and have shared skills as well as individual specializations.

2. In what ways does the department maximize the use of material resources such as space, equipment, operating funds, etc? What information technology and library support is needed for this department?

Space is primarily allocated based on active, supported graduate students. Until recently, faculty needing specialized equipment could request OBR equipment funds in a competitive process twice per year.

3. What strategies does the department employ to develop alternate sources of revenue?

Under the new RCM budget model, Master's students provide more income than undergraduate students (roughly 2x in state subsidy and tuition) so the department may choose to market and expand the non-thesis Masters program. Marketing and expanding summer course offerings could also help, since faculty benefit rates are lower then. More extramural funding would also provide an increase in departmental overhead funds.

4. What additional resources must the department acquire in order to do its job well? What evidence, sufficient to convince deans and provosts, has this department review assembled?

With the loss of the OBR funds and KSU's switch to RCM, the department now has a $2M deficit and faces the loss of some GA funds. It is important that the university realize that the faculty size doubled under state direction to grow a doctoral program. Unfortunately, this came at a time in the computer science boom-bust cycle when undergraduate enrollments were declining nationally, exacerbating the financial crisis.

5. How do the resources available to the department compare to those available at other institutions in the state and nationally?

Compared to other CS PhD departments, KSU CS faculty salaries are close to national norms for Assistant Professors, but are 30% or more below the national median for full Professors. GA stipends are slightly less than national norms. The departmental physical space is about half the national norm for offices, research labs, and instructional labs.

G. Demand (The department should indicate that it is responding to interests from its stakeholders. The review should answer the following questions.)

1. What is the method for projecting student need for the program(s) under review? What is the prediction for student need for the program for both the short-term and the long-term? What is the optimal number of majors for this program? How was that number determined?

The primary national forecast comes from the Bureau of Labor Statistics, released in the last quarter of odd-numbered years. For the past several reports, the forecast has been the same: for the next decade, around 60% of the new jobs plus net replacements in science and engineering will be for computer specialists.

A report from the Ohio Department of Job and Family Services concurs, predicting that computer software engineering and applications will be among Ohio's fastest growing occupations between 2006 and 2016. That report predicts an employment growth rate in that area of 38.9% with an average of 880 annual job openings.

A more local forecast was NorTech's "Northeast Ohio Information Technology Workforce Report", released in March of 2008. That report shows that the need for IT professions in Northeastern Ohio was significantly more than the colleges and universities in the area were producing.

Based on the department's number of faculty and classroom / lab availability, the department could support up to 350 undergraduate majors and 150 graduate students.

2. What is the method for projecting employer demand for graduates of this program? What is the expectation for employer demand for the program for both the short-term and long-term?

See answer to Question 1.

3. Who are the program's major competitors? How is the program under review unique? What community need(s) is it meeting?

The major local competitor is the University of Akron, located only a short distance away. Other local competitors are Case Western Reserve (a private university), and Cleveland State and Youngstown State, both with departments that combine computer science and information systems in a single department. However, KSU is the only public university in the region offering a PhD in Computer Science.

The KSU Computer Science Department is also one of the few departments in the state with a PhD program that is housed in a College of Arts and Sciences at a public university. As a result, the department has strong ties across the sciences, resulting in interdisciplinary collaborations with the departments of Biological Sciences, Chemistry, Physics, Chemical Physics, etc.

See Section E5 for a list of some of the many Ohio employers that hire KSU CS graduates.

Action Plan (3-5 pages)

Summarize in 2-pages or less the strengths and opportunities of your department in comparison to the benchmark departments. Discuss in 3 pages or less what steps the department plans to move your department towards the stature enjoyed by the program(s) you hope to emulate. You will be asked for an interim report in 18 months after the conclusion of this review.

A. Department Strengths

In the tables below, the KSU CS Department is compared to the CS Department at three other universities in Ohio, to two "peer" departments, and to two "larger" departments that we might choose to emulate.

KSU's CS Department is characterized by 18 faculty (19 if Kent State University at Stark is included), with few recent hires and the majority of the faculty at the full Professor level. The Department has slightly over 200 undergraduate majors, or slightly under 300 if the Regional Campuses are included. The Department has slightly fewer than 100 graduate students, with slightly more MS students than PhD students. The Department has GA funds for approximately 24 GAs (aid varies by level) and has 3 RAs on grant support.

	Asst	Assoc	Prof	All	Ugrad	MS	PhD	Grad	GAs	RAs
Kent State University (CS, A&S)	(3) 2	5	11	(19) 18	(291) 206	54	42	96	24	3

With the exception of Wright State University, which will be discussed later, most CS departments in the State of Ohio are not directly comparable. Both Ohio State University and the University of Cincinnati offer a PhD in CS, but Ohio State has a much larger department and the University of Cincinnati has a much smaller one. Like KSU, Ohio University is another state "corner" university, but it does not offer a PhD in CS and has a much smaller department. The only other university in the state offering a PhD in CS is Case Western Reserve University, which is a private university.

	Asst	Assoc	Prof	All	Ugrad	MS	PhD	Grad	GAs	RAs
Kent State University (CS, A&S)	(3) 2	5	11	(19) 18	(291) 206	54	42	96	24	3
Ohio State University	7	14	14	35						
Ohio University	3	5	1	9						
University of Cincinnati (CS, Engr)	0	3	7	10	136	40	40+	80+		

The two universities in this part of the country that have CS departments that seem to most closely match KSU's CS Department are the University of Kentucky and the University of Missouri-Columbia. Both departments have a similar number of faculty, undergraduate majors, and graduate students to the KSU CS Department, but a larger number of RAs. Both are also in a College of Engineering, unlike KSU's CS Department, which is in a College of Arts and Sciences. However, the University of Kentucky has a Carnegie Classification of Very High Research Activity, unlike KSU, which is classified as High Research Activity.

Compared to the KSU CS Department, the University of Kentucky has slightly more faculty, but like KSU has a faculty where the majority are full Professors. They have a similar number of undergraduate majors and graduate students as KSU, though with a slightly higher ratio of PhD to MS students. Their number of GAs is comparable to KSU, though they have many more RAs.

Compared to the KSU CS Department, the University of Missouri-Columbia has a similar number of faculty, though the majority of their faculty are Associate Professors. They have slightly more undergraduate majors and graduate students than KSU, but a comparable PhD to MS ratio. They also have a comparable number of GAs, but like the University of Kentucky, have many more RAs.

	Asst	Assoc	Prof	All	Ugrad	MS	PhD	Grad	GAs	RAs
Kent State University (CS, A&S)	(3) 2	5	11	(19) 18	(291) 206	54	42	96	24	3
University of Kentucky (CS, Engr)	4	5	12	21	200	39	66	105	21	40
University of Missouri-Columbia (CS, Engr)	4	10	4	18	244	58	52	110	26	18

B. Department Areas Needing Improvement

Comparative data on extramural funding was not available, but since federal grants in computer science often support graduate students as RAs, the data above seems to suggest that the KSU Computer Science Department's federal extramural funding is much less than its peers. Delaware Study Comparisons, comparing KSU's CS Department to the university's peers (though not necessarily the department's peers), shows the KSU CS department to have 24%, 24%, and 31% of the research expenditures of those peers for AY 2005-06, 2006-07, and 2007-08, respectively.

C. Planned Department Changes or Developments with Timeline and Budget

The two universities in this part of the country that have CS departments that seem to be one step above KSU's CS department are Wright State University and Wayne State

University. Wright State characterizes the path for growth in undergraduate majors and MS students, while Wayne State characterizes the path for growth in PhD students.

Like Ohio State University and the University of Cincinnati, Wright State University is an Ohio public university offering a PhD in CS. Compared to the KSU CS Department, Wright State University has a similar number of faculty, though more evenly distributed across faculty ranks. They have 36% more undergraduate majors and 41% more MS students, though a similar number of PhD students. Their number of GAs is half that of KSU, though they have more RAs. Like KSU, Wright State University has a Carnegie Classification of High Research Activity.

Compared to the KSU CS Department, Wayne State University has slightly more faculty, though the majority of their faculty are Associate Professors. They have a similar number of undergraduate majors and MS students as KSU, but 90% more PhD students. They also have more GAs and many more RAs. However, Wayne State University has a Carnegie Classification of Very High Research Activity

	Asst	Assoc	Prof	All	Ugrad	MS	PhD	Grad	GAs	RAs
Kent State University (CS, A&S)	(3) 2	5	11	(19) 18	(291) 206	54	42	96	24	3
Wright State University (CSE, CECS)	7	6	6	19	280	76	45	121	12	11
Wayne State University (CS, A&S)	5	12	4	21	191	50	80	130	30	25

Taking the Wright State University path — striving for growth in undergraduate majors and Masters students — is the KSU Computer Science Department's most likely path for growth in the near future. Growth in undergraduate majors could be achieved through a combination of increased outreach and improved retention.

The KSU Computer Science Department's outreach activities waned over the years, but have improved recently. Several departmental faculty have started outreach activities with local middle schools. KSU's Admissions Office recently started encouraging departmental contact with admitted undergraduate students, and calls to those students by departmental faculty seem to be well-received by the students and their parents. The department's presentation on university open-house days has been updated to be more appealing to prospective majors.

The department has also placed a greater emphasis on undergraduate retention in the past couple of years. The outdated lab materials for *CS 10051 Introduction to Computer Science* are being updated this year, following similar updates last year in *CS 23021 CSI: Programming and Problem Solving*. After the loss of the department's three non-tenure-track faculty due to budgetary concerns, four tenure-track faculty were selected as Undergraduate Advisors. More tenure-track faculty were assigned to teach required undergraduate classes this year, with departmental GAs generally teaching only *CS 10001 Computer Literacy* or lab sections of *CS 10051 Introduction to Computer Science* or *CS 23021 CSI: Programming and Problem Solving*. This shift of tenure-track

faculty into required courses freed more GAs for other duties, so the department now provides tutoring for all required CS classes through *CS 33001 CSII: Data Structures and Abstraction*.

Though the department has made many positive steps toward increasing outreach and improving retention, many other steps could be taken as well. The department could choose to build up its very small summer program by offering additional summer courses. However, those courses would have to be planned further in advance than in recent years, and marketed more aggressively, both internally to departmental majors as well as locally to the liberal arts colleges in the area. The department could choose to offer summer camps for local middle school students (as Wright State University does) in an effort to attract those students to computing during their formative years. The department could reach out more specifically to women and other under-represented groups, perhaps forming a local ACM-W Student Chapter to complement the local ACM Student Chapter. Undergraduate Teaching Assistants could be selected to complement Graduate Assistants as tutors and assistants in lab sections of *CS 10051 Introduction to Computer Science* or *CS 23021 CSI: Programming and Problem Solving*.

The Masters program also has potential for growth, particularly the new non-thesis MS option. Though not common in science departments at KSU, most of the department's competitors in Ohio offer a non-thesis Masters degree as well as a thesis-based Masters. The non-thesis option can be particularly attractive for those students looking for a terminal degree at the Masters level and seeking a non-research position in industry. Those students can complete their coursework, and then quickly enter the workforce, as opposed to mostly completing their coursework but then spending the next year or two working on a thesis.

The number of departmental faculty and classroom space is sufficient to support growth in both undergraduate majors and graduate students to the Wright State University level. Undergraduate classes and lab sections are generally less than 20 students, while the department's two first-floor classroom can hold 35 students. If class sizes were to increase by 50%, the department should be able to support as many as 300-350 undergraduate majors on the Kent State Campus. If classroom space could be found outside the Math and CS Building (which is quite problematic), this number could increase further. Similar growth in the number of graduate students could also be absorbed.

Taking the Wayne State University path — striving for growth in PhD students — is much harder with the recent loss of state funding for enhancing the department's doctoral program, though is still a viable path. Though Wayne State's distribution of GA/RA funds is unknown, they have funding for 55 graduate students, meaning they could fund as many of 69% of their 80 PhD students. The KSU CS Department could fund a similar ratio of its PhD students, with funding for 27 graduate students supporting 64% of its 42 PhD students. However, Wayne State has a much higher ratio of RAs to GAs, so if KSU CS faculty could raise their extramural research funding from the current level of 3 RAs to 20 RAs, 24 GAs + 20 RAs could support 2/3 of 66 PhD students — much closer to Wayne State's 80 PhD students.

	A	B	C	D	E	F	G
1	**Majors**	Summer 2009	Spring 2009	Fall 2008	Summer 2008	Spring 2008	Fall 2007
2	Bachelors - Regional		64	51		39	46
3	Bachelors - Kent		152	155		158	200
4	Master		43	49		45	57
5	PhD		44	45		42	39
6							
7	**Enrollment - Main campus**	Summer 2009	Spring 2009	Fall 2008	Summer 2008	Spring 2008	Fall 2007
8	Bachelors	27	503	496	31	531	651
9	Master	0	44	62	0	54	53
10	PhD	0	48	44	0	31	29
11							
12	**Degrees granted**	Summer 2009	Spring 2009	Fall 2008	Summer 2008	Spring 2008	Fall 2007
13	Bachelors	2	15	7	0	16	10
14	Masters	3	6	5	3	8	3
15	Doctoral	0	1	1	2	1	1
16							
17	**Faculty**	Summer 2009	Spring 2009	Fall 2008	Summer 2008	Spring 2008	Fall 2007
18	TT (includes 1 regional F08-Sum09)	19	20	20	19	19	19
19	NTT	0	3	3	3	3	3
20	PT	0	4	7	1	6	8
21	Term						
22							
23							
24							

25	**Tution Scholarship AY 2009-2010**			
26		GA	TF	RA
27	students	19	3	2
28	amount	$116,079	$846	$15,192
29				
30	**Stipend funding AY 2009-2010**			
31		GA	TF	RA
32	students	20	3	3
33	amount	$290,000	$54,000	$33,750

34

35 **Web site:**

36 http://www.cs.kent.edu

	H	I	J	K	L	M	N	O	P
1	Summer 2007	Spring 2007	Fall 2006	Summer 2006	Spring 2006	Fall 2005	Summer 2005	Spring 2005	Fall 2004
2		37	58		57	74		64	82
3		183	208		191	236		235	289
4		65	76		93	98		112	122
5		41	45		44	46		44	49
6									
7	Summer 2007	Spring 2007	Fall 2006	Summer 2006	Spring 2006	Fall 2005	Summer 2005	Spring 2005	Fall 2004
8	54	652	590	43	693	718	33	762	783
9	4	44	66	6	57	110	10	137	200
10	1	30	26	1	28	29	0	29	37
11									
12	Summer 2007	Spring 2007	Fall 2006	Summer 2006	Spring 2006	Fall 2005	Summer 2005	Spring 2005	Fall 2004
13	5	15	5	2	16	11	2	22	18
14	6	8	8	6	4	9	8	4	8
15	2	2	3	0	0	0	1	2	3
16									
17	Summer 2007	Spring 2007	Fall 2006	Summer 2006	Spring 2006	Fall 2005	Summer 2005	Spring 2005	Fall 2004
18	19	19	19	18	18	18	17	17	17
19	3	3	3	3	3	3	3	3	3
20	1	5	6	2	8	9	1	8	9
21	1	1	1	1	1	1	1	1	1
22									
23									
24									
25									
26									
27									
28									
29									
30									
31									
32									
33									
34									
35									
36									

Appendix A

Section A:

Our bachelor's program follows the ACM's IEEE computing curricula of 2001. We had a CS Doctoral Program Review in 1995 and in 2003.

Appendix B

Section B1:
Required and Elective Courses

Sorted by Course Number

Undergraduate Requirements:

CS 10051	INTRO TO COMPUTER SCIENCE
CS 23021	COMPUTER SCIENCE I
CS 23022	DISCRETE STRUCTURES FOR CS
CS 33001	COMPUTER SCIENCE II
CS 33006	SOCIAL AND ETHICAL ISSUES
CS 33101	STRUCTURE OF PROGRAMMING LANGUAGES
CS 33211	OPERATING SYSTEMS
CS 35101	COMPUTER ARCHITECTURE
CS 43005	INTRO TO DATABASE SYS DES
CS 43901	SOFTWARE ENGINEERING
CS 45201	COMP COMM NETWORKS
CS 46101	DSGN & ANAL-ALGORITHMS
CS 49901	CAPSTONE PROJECT

Undergraduate Electives:

CS 10001	COMPUTER LITERACY
CS 10061	INTRO TO COMPUTER PROG
CS 29995	ST: LAB FOR CS 23021
CS 33011	SPEED PROGRAMMING TECHNIQUE
CS 33092	SPEED PROGRAMMING
CS 33192	INTERNSHIP IN COMPUTER SCI
CS 33223	UNIX TOOLS
CS 39995	ST: UNIX TOOLS AND PROG
CS 42201	INTRO NUMERICAL COMPUT I
CS 42202	INTRO NUMERICAL COMPUT II
CS 43006	THEORY OF OBJ-ORIENT PROG
CS 43111	STRUCTURE OF COMPILERS
CS 43202	SYSTEMS ADMINISTRATION
CS 43203	SYSTEMS PROGRAMMING
CS 44201	ARTIFICIAL INTELLIGENCE
CS 45111	PRINCIPLES OF VLSI DESIGN
CS 45231	INTERNET ENGINEERING
CS 46201	AUTOMATA FORMAL LANGUAGES
CS 47101	COMPUTER GRAPHICS
CS 47105	WEB DESIGN & PROGRAMMING I
CS 47106	WEB DESIGN & PROGRAMMING II
CS 47107	WEB DESIGN & PROGRAM STUDIO
CS 47205	INFORMATION SECURITY
CS 49995	ST: GAME ENGINES

CS 49995 ST: GENERIC PROG/LIB DESIGN
CS 49995 ST: HUMAN COMPUTER INTERFACE
CS 49995 ST: INTRO TO BIOINFORMATICS
CS 49995 ST: INTRO TO CRYPTOGRAPHY
CS 49995 ST: MULTICORE COMPUTING
CS 49995 ST: REQUIREMENTS ENGINEERING
CS 49995 ST: SECURE CODING
CS 49995 ST: SENSOR NETWORKS
CS 49996 INDIVIDUAL STUDY
CS 49998 RESEARCH

Graduate Requirements - Master's:
CS 69191 MASTERS SEMINAR - Thesis & Non-Thesis Option
CS 69199 THESIS I - Thesis Option
CS 69299 THESIS II - Thesis Option

Graduate Requirements - PhD:
CS 89191 DOCTORAL SEMINAR
CS 89199 DISSERTATION I
CS 89299 DISSERATATION II

Graduate Electives:
CS 52201 INTRO NUMERICAL COMPUT I
CS 52202 INTRO NUMERICAL COMPUT II
CS 53005 INTRO TO DATABASE SYS DES
CS 53006 THEORY OF OBJ-ORIENT PROG
CS 53111 STRUCTURE OF COMPILERS
CS 53202 SYSTEMS ADMINISTRATION
CS 53203 SYSTEMS PROGRAMMING
CS 53901 SOFTWARE ENGINEERING
CS 54201 ARTIFICIAL INTELLIGENCE
CS 55111 PRINCIPLES OF VLSI DESIGN
CS 55201 COMP COMM NETWORKS
CS 55231 INTERNET ENGINEERING
CS 56101 DSGN & ANAL-ALGORITHMS
CS 56201 AUTOMATA FORMAL LANGUAGES
CS 57101 COMPUTER GRAPHICS
CS 57105 WEB DESIGN & PROGRAMMING I
CS 57106 WEB DESIGN & PROGRAMMING II
CS 57107 WEB DESIGN & PROGRAM STUDIO
CS 57205 INFORMATION SECURITY
CS 59995 ST: GAME ENGINES
CS 59995 ST: GENERIC PROG/LIB DESIGN

CS 59995 ST: HUMAN COMPUTER INTERFACE
CS 59995 ST: INTRO TO BIOINFORMATICS
CS 59995 ST: INTRO TO CRYPTOGRAPHY
CS 59995 ST: MULTICORE COMPUTING
CS 59995 ST: REQUIREMENTS ENGINEERING
CS 59995 ST: RESEARCH PAPER WRITING
CS 59995 ST: SECURE CODING
CS 61001 STRUCT OF COMPUTER SCIENCE
CS 62201 NUMERICAL COMPUTING I
CS 62202 NUMERICAL COMPUTING II
CS 63005 ADV DATABASE SYSTEM DESIGN
CS 63015 DATA MINING TECHNIQUES
CS 63111 ADVANCED COMPILER DESIGN
CS 63201 ADV OPERATING SYSTEMS
CS 63301 PARALLEL & DISTR COMPUTING
CS 63304 CLUSTER COMPUTING
CS 63901 SOFTWARE ENGINEERING METHOD
CS 63903 SOFTWARE VISUALIZATION
CS 63995 ST: COMPUTATIONAL GEOMETRY
CS 63995 ST: DATA MINING
CS 63995 ST: SECUR IN DIS SYS & C NETW
CS 63995 ST: SOFTWARE EVOLUTION
CS 64301 PATTERN RECOGNITION PRINC
CS 64401 IMAGE PROCESSING
CS 65101 ADV COMPUTER ARCHITECTURE
CS 65201 INTERCONNECTION NETWORKS
CS 65202 ADV COMMUNICATION NETWORKS
CS 65203 WIRELESS AND MOBILE COMMUNICATION NETWORKS
CS 65301 SYSTEM MODELING AND PERFORMANCE EVALUATION
CS 66101 ADVANCED TOPICS IN ALGORITHMS
CS 66105 PARALLEL & DISTR ALGORITHMS
CS 66110 COMPUTATIONAL GEOMETRY
CS 66120 EVOLUTIONARY COMPUTATION
CS 67101 ADVANCED COMPUTER GRAPHICS
CS 69098 RESEARCH
CS 69995 ST: ADV ALG FOR GRAPHS AND NETS
CS 69995 ST: ADV INFO SEC AND CRYPTOLOGY
CS 69995 ST: ALGORITHMS GRAPH & NET
CS 69995 ST: ALGORITHM-WIRELESS AD HOC
CS 69995 ST: COMPLEX NETWORKS
CS 69995 ST: DIST MULTIMEDIA LANG
CS 69995 ST: FOUND PEER-TO-PEER SYS
CS 69995 ST: GPU COMPUTING
CS 69995 ST: GRAPH MINING
CS 69995 ST: GRAPH MINING AND MGMT
CS 69995 ST: GRID COMPUTING

CS 69995 ST: INTERNET SYS & APPLS
CS 69995 ST: INTERNET SYSTEMS
CS 69995 ST: MED IMAGE PROCESSING
CS 69995 ST: ML FOR BIOINFORMATICS
CS 69995 ST: MOBILE AD HOC NETWORKS
CS 69995 ST: PARAL-DIST REAL TIME SYS
CS 69995 ST: PARALLEL REAL-TIME SYSTEMS
CS 69995 ST: PROGRAM COMPREHENSION
CS 69995 ST: SCIENTIFIC VISUALIZATION
CS 69995 ST: SORTING NETWORKS
CS 69995 ST: SYSTEM SIMULATION
CS 70094 COLLEGE TEACHING OF CS
CS 72201 NUMERICAL COMPUTING I
CS 72202 NUMERICAL COMPUTING II
CS 73005 ADV DATABASE SYSTEM DESIGN
CS 73015 DATA MINING TECHNIQUES
CS 73201 ADV OPERATING SYSTEMS
CS 73301 PARALLEL & DISTR COMPUTING
CS 73304 CLUSTER COMPUTING
CS 73901 SOFTWARE ENGINEERING METHOD
CS 73903 SOFTWARE VISUALIZATION
CS 73995 ST: COMPUTATIONAL GEOMETRY
CS 73995 ST: DATA MINING
CS 73995 ST: SOFTWARE EVOLUTION
CS 74301 PATTERN RECOGNITION PRINC
CS 74401 IMAGE PROCESSING
CS 75101 ADVANCED COMPUTER ARCHITECTURE
CS 75201 INTERCONNECTION NETWORKS
CS 75202 ADV COMMUNICATION NETWORKS
CS 75203 WIRELESS AND MOBILE COMMUNICATION NETWORKS
CS 75301 SYSTEM MODELING AND PERFORMANCE EVALUATION
CS 76101 ADVANCED TOPICS IN ALGORITHMS
CS 76105 PARALLEL & DISTR ALGORITHMS
CS 76110 COMPUTATIONAL GEOMETRY
CS 76120 EVOLUTIONARY COMPUTATION
CS 77101 ADVANCED COMPUTER GRAPHICS
CS 79995 ST: ADV ALG FOR GRAPHS AND NETS
CS 79995 ST: ADV INFO SEC AND CRYPTOLOGY
CS 79995 ST: ALGORITHMS GRAPH & NET
CS 79995 ST: ALGORITHM-WIRELESS AD HOC
CS 79995 ST: COMPLEX NETWORKS
CS 79995 ST: DIST MULTIMEDIA LANG
CS 79995 ST: FOUND PEER-TO-PEER SYS
CS 79995 ST: GPU COMPUTING
CS 79995 ST: GRAPH MINING AND MGMT
CS 79995 ST: GRID COMPUTING

CS	79995	ST: INTERNET SYS & APPLS
CS	79995	ST: INTERNET SYSTEMS
CS	79995	ST: MED IMAGE PROCESSING
CS	79995	ST: ML FOR BIOINFORMATICS
CS	79995	ST: MOBILE AD HOC NETWORKS
CS	79995	ST: PARAL-DIST REAL TIME SYS
CS	79995	ST: PARALLEL REAL-TIME SYSTEMS
CS	79995	ST: PROGRAM COMPREHENSION
CS	79995	ST: SCIENTIFIC VISUALIZATION
CS	79995	ST: SENSOR NETWORKS
CS	79995	ST: SORTING NETWORKS
CS	79995	ST: SYSTEM SIMULATION
CS	83991	SEM IN SOFTWARE DESIGN
CS	89098	RESEARCH

Section B2:

Course Syllabus
Parallel and Distributed Computing
Fall 2008

Instructor:

- Professor Johnnie W. Baker
- Also Professor Robert Walker will give 2-3 lectures while I attend a conference.
  - Examples of typical parallel architectures
  - His research group's work using FPGAs to create specific parallel architectures

Prerequisites:

- Designed to be accessible to all graduate students in computer science
- Students who are not CS graduate students may also be qualified to take course

Description: Includes a broad coverage of the fundamental concepts of parallel computation rather than focusing primarily on the latest trends, which are often quickly outdated due to the rapid changes in technology in this area. The principal types of parallel computation are covered by investigating three key features for each: typical architectural features, typical programming languages, and algorithm design techniques. Also the popular MPI language used with a wide range of parallel computers will be introduced. Also introduces SIMD programming using the C^n language on the ClearSpeed Accelerator. Programs will be written in both MPI and C^n.

Specific Topics Included in Course:

- Covers the fundamental (i.e., classic) concepts applicable to all parallel computation, rather than focusing only on latest trends, which are often quickly outdated due to rapid technological changes
- Also covers message-passing processors (e.g., clusters), shared memory processors, and the MPI language.
  - This is the focus of the Quinn Textbook
- Covers the common types of parallel computation by looking at three key features for each type:
  - Typical architectural features
  - Typical programming languages used
  - Typical algorithm design techniques used.
- Asynchronous (MIMD) distributed memory computation
  - Message passing communications
  - Programming using the MPI Language
  - Architectural features
  - Examples of typical algorithms

- Asynchronous (MIMD) shared memory computation
  - Symmetric Multiprocessors or SMPs
  - Perhaps a brief OpenMP language overview
- Synchronous Computation
  - SIMD, vector, pipeline computing
  - Associative and Multi-Associative Computing
  - Introduction to associative programming using ASC language
  - Programming on ClearSpeed system using the C^n language.
  - Algorithm examples
- Interconnection Networks
  - Specific computer networks including 2D mesh, hypercube, etc.
  - Synchronous and asynchronous network considerations
- Compare advantages and disadvantages of each MIMD and SIMD computation
  - Includes the use of a real-time ATC (Air Traffic Control) application

Some Expected Benefits of Course:

- While primary focus is primarily on parallel computation, most information is applicable to distributed computing.
- There is a wide choice of thesis and dissertation topics in this area
- Several professors in department work in this area or make major use of parallel computation in their research
- Students working on a thesis or dissertation in another area may benefit from being able to use parallel computation in their work.
- Fundamental knowledge of parallel is of increasing importance.
  - Most large computational problems require a parallel or distributed system to satisfy the speed and memory requirements
  - The use of parallel programming is expected to become widespread due to fact that multiple processors are currently being created on single chips.
- Parallel computation currently has major advantages over both distributed computation and grid computation for computational intensive problems.
  - Programs are normally much simpler
  - Architectures are much cheaper
  - Efficient use of grid computing as a “massive compute engine” is fairly futuristic.

Textbook and References:

- Slides will contain a lot of information and will be a primary source of information for this course.
- Textbook
  - Parallel Programming in C with MPI and OpenMP
  - Michael Quinn, author

  - Published by McGraw Hill in 2004
- References for Supplementary Reading
  - Classroom Slides will also include additional information from a wide range of sources
  - Any additional required reference material will be handed out or posted on course website
- Also, an online reference textbook (by Selim Akl) and a pointer to a second online reference textbook (by Ian Foster) will be available at this site.

Grading Policy

- Homework assignments
  - Problems, activities, or programs assigned for most chapters
  - Probably 5-6 different assignments
  - Some assignments will involve programming
- Course Grade
  - Based on homework, midterm, and final
  - Approximate weights
    - Homework 40%
    - Midterm Exam 30%
    - Final Exam 30%

Course Website

- Website will be established at http://www.cs.kent.edu/~jbaker/PDC-F08/
- Class slides, assignments, and numerous references will be posted on this website.

Attendance & Makeup Policies:

- Regular attendance is important in this class. There is a strong correlation between class attendance and grade performance. On the rare occasions that you cannot avoid being absent, you are responsible for getting class notes and assignments. While class slides posted on the class website will provide information about the material covered in class, they will not include some important information such as discussion of points on slides, class discussions, and information written on the board.
- Students are expected to take all examinations at the scheduled time. If a missed exam is not excused, your grade for that exam will be zero. To receive an excused absence, either you must contact me in advance and receive permission to be absent or else present documented evidence of illness or of an individual/family emergency situation.

Accessibility Statement:

- University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. **Please note, you must first verify your eligibility for these through Student Accessibility Services** (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

A&S Cheating & Plagiarism Statement:

- The College of Arts & Sciences "Cheating & Plagiarism" policy applies to this class. A statement of this policy is available at http://www.kent.edu/policyreg/chap3/3-01-8.cfm
- A PDF version formatted for printing is at the site http://www.kent.edu/policyreg/chap3/upload/3342.3.01.8.pdf
- A link to this statement will be posted on the website for this course

Registration Requirement:

- The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking their class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

Parallel Real-Time Systems
An Online Computer Science Graduate Course
KSU Course Number: CS 6/79995
Spring 2009

Instructors: Dr. Johnnie Baker at CS Dept at Kent State University and Dr. Frank Drews at EECS Dept at Ohio University

Prerequisites: Admission to the computer science graduate program or permission.

Credit: This course is designed to be a 3 hour semester system course for the spring semester or a 4 hour quarter system course for the winter quarter in 2009.

Description: Real-time system professionals build our modern space, avionics, defense, telecommunication, transportation, e-commerce, automated manufacturing, and process control and robotics systems. However, many real-time systems require far more computational power and memory than can be provided by conventional CPU architectures and typically utilize parallel architectures. This course will focus on providing the background knowledge and skills needed for the design and construction of large real-time systems and applications. This will be a project-oriented course, and will involve a series of problems for the students to solve. In the course of their projects, students will obtain hands-on experience with the design, implementation and analysis of real-world applications. The air traffic control (ATC) system is a classic large scale real-time application and will be used to provide examples and some projects for this class.

Class Presentation Mode and Schedule: This class will meet in Room 243 (Graphics/Media Room) in MSB. It will be an online course, with students at both KSU and OU. The two classrooms at KSU and OU will be connected using the polycom equipment in Rm 243 so that participants at both KSU and OU will be able to listen to lectures, ask questions, and in general, to communicate with each other. Since OU is on a quarter system and KSU is on a semester system, the entire class will be taught during the time that both OU is in their 2009 Winter quarter and KSU is in the Spring 2009 semester. In particular, classes will start on January 20, 2009 (the start of the KSU 2009 semester) and will stop on March 20 (the end of the OU 2009 winter quarter). The final exam period will occur the following week, probably during the standard class meeting time. Classes will meet on MWF from 10:25-12:15, with a 15 minute break around the middle of the class period.

Major Units and Approximate Descriptions:

1) Brief coverage of basic parallel computation knowledge needed for this course, such as
   - Basic types of parallel architectures
   - Basic metrics for evaluating efficiency of parallel programs (e.g., cost, speedup)
   - General principles for designing efficient parallel programs

2) Basics of parallel programming
   - Introduction to programming on the ClearSpeed system using the parallel C^n extension of the C language.
   - Short overview of MPI programming to provide background information.
3) Introduction to real-time processing, including
   - Dealing with a mix of periodic, aperiodic, and sporadic tasks
   - Dealing with a mix of non-real-time, soft real-time, and hard real-time activities
4) Overview of conventional real-time design principles
5) Introduction to the ATC problem. Information about this problem will be introduced as needed during the course to provide needed examples and projects. Activities include
   - Discussion aspects of providing a multi-computer solution to the ATC
   - Discussion and projects involving providing a solution to ATC using the ClearSpeed system.

Textbook and References: There is currently no appropriate text or reference book for this course. While there are a number of textbooks on (sequential) real-time systems, these generally contain little or no material concerning parallel real-time systems. Our PowerPoint Slides will contain considerable information and will be a primary reference for this course. Additionally, students will be provided with some online papers and other reference material.

Grading Policy: Since this is the first time this course has been offered, some flexibility will be needed so that some fine-tuning can occur during the course. As this course will be project-oriented, a major portion (i.e., 60%-70%) of the grade will be based on work related to your performance on these projects. There will be some homework and/or graded in-class exercises, and this will constitute at most 20% of your grade. One or two exams will be given during the course and these will constitute 20%-40% of your grade.

Attendance & Makeup Policies: Since this is a project-oriented course, students are expected to attend and participate in all classes. Class sessions may frequently involve activities in which students learn the course material. Some in-class activities are expected to be evaluated. Makeup activities and project deadline extensions will not be granted, except for legitimate reasons (based on standard policies at the respective universities).

Syllabus

Spring 2009

CS 10051 - Introduction to Computer Science Syllabus

instructor	Dr Johnnie Baker
class	CS 10051-005/006, call number 11783/11784 2:15 - 3:05 MWF, room 228 MSB, Section 005 Lab: 2:15 pm - 4:10 pm T room 139 MSB Mike Yuan, Instructor Section 006 Lab: 2:15 pm - 4:10 pm H room 139 MSB Mike Yuan, Instructor
office	260 MSB
office hours	1:00 - 2:00 MWF and by appointment
my website	www.cs.kent.edu/~jbaker
class website	www.cs.kent.edu/~jbaker/CS10051-Sp09
email	**jbaker@cs.kent.edu (Please use this email address rather than my kent.edu address as I check it more frequently).**
voice mail	(67-) 29061
office phone	(67-) 29061
CS dept.	office: 233 MSB phone: (67-) 29980
lecture textbook	*Invitation to Computer Science*, C++ edition, 4th Edition Judith Gersting, G. Michael Schneider Thompson/Course Technology, Copyright 2007 ISBN-10: 142390141X or ISBN-13: 9781423901419 **---768 Pages**, Paperbound, http://www.course.com/catalog/product.cfm?isbn=978-1-4239-0141-9&CFID=7189466&CFTOKEN=93312120 or http://academic.cengage.com/cengage/catalog.do?courseid=CKS02&disciplinenumber=200&codeid=Z391&codeFlag=true
lab textbook	*Invitation to Computer Science Laboratory Manual: C++ and Java*, Kenneth Lambert, Thomas Whaley ISBN 13: 978-1-4188-3754-9 ? 2007 ISBN 10: 1-4188-3754-7 http://course.cengage.com/catalog/product.cfm?isbn=978-1-4188-3754-9&CFID=9363009&CFTOKEN=14196898
tests during term	Two tests (i.e., an early midterm and a late midterm) will be given during the semester. Each exam will be 20% of your final grade. The tests will be announced at least one week in advance. Normally some questions similar to questions on earlier tests may are included

	on all but the first test. As a result, you should review your earlier tests when preparing for your late term test and your final exam. Students are expected to take all examinations at the scheduled time. If a missed exam is not excused, your grade for that exam will be zero. To receive an excused absence, you must either contact me in advance and receive permission to be absent or else present documented evidence of illness or of an individual/family emergency situation. Routine doctor or dental appointments are not an excuse for missing a scheduled examination. A death in the immediate family requires a confirmation of the death and your very close connection to that individual.
homework & class participation	10% of the final grade. Includes class participation, homework presentations in class, homework, and any pop quizzes. You will receive a grade of zero on any pop text missed, unless your absence is excused. Either excessive class absences (i.e,., more than 3 unexcused absences) or failure to turn in your homework on time (unless excused) can severely damage your homework & class participating grade. A poor grade in this category can lower your grade by a full letter.
lab	30% of the final grade.
final exam	20% of the final grade. The final exam will be comprehensive and will cover material from the entire course. Students are encouraged to also study their earlier tests, as some similar questions to the early term and late term exams are likely to be included on the final exam. The final exam will be given at the time indicated in the schedule of classes exam schedule.
final exam time	12:45 - 3pm on Wednesday May 13. The time of the final is fixed by the University and can not be changed.
grading scale	The University +/- grading scheme is used, which (ignoring rounding of decimals) is as follows: 93-100 is A, 90-92 is A-, 87-89 is B+, 83-86 is B, 80-82 is B-, 77-79 is C+, 73-76 is C, 70-72 is C-, 67-69 is D+, 60-66 is D, and below 59 is F. Numeric scores are not converted to letter grades until the end of the semester.
prerequisite	Prerequisites for this class is two years of high school algebra. For more details, see the course description in the catalog. This class is a prerequisite for CS 23021, *Introduction to Object Oriented Programming*.
overview	CS 10051 is broad introduction to computer science. Many important concepts underlying computer science are covered. This includes the algorithmic foundations of computer science and the expression of algorithms as pseudocode. A number of algorithms are examined including sequential search, find greatest, selection sort, and binary search. The time efficiency of algorithms and Big-O classification are discussed. Next, computer hardware and organization are discussed. In particular, the basic building blocks of a computer, e.g., binary numbers, Boolean logic, gates, and

circuits to add and compared, are studied. Using these elementary pieces, the construction of a CPU and a Von Neumann style computer is studied. We study a typical instruction set for this type of machine and its corresponding assembly language. Basic pseudocode control stuctures are implemented in assembly language.

We next see how layers of software can be used to hide the complexity of these machines and make them easier to use. We study the virtual machine environment provided by the operating system and others system software. The virtual machine environment allow us to program and run algorithms using tools like text editors, assemblers, and loaders, rather than dealing with detail machine specifics. Computer network basics are studied next.

The final step in this hierachy is to see how a programming language provides another layer of software that makes these machines even easier to use, and allow us to program in a more human-oriented language. The language C++ is introduced as an example of a high-level programming language. However, we stop with Section 8.6 and leave the OOP (object oriented language) and discussion of functions in C++ to courses CS1 and CS2. Finally, as time permits, various topics from chapters 9-15 are covered.

This course includes a 2 hour weekly lab that provides experience with the concepts covered in the lectures.

reading assignments

The assigned reading assignments will be made in class. They may also be posted on the course website. These should be read **in advance** of the coverage of this material in class.. This will allow you to ask questions in class and clarify any concepts that are unclear. You should read the assigned material both **before and following** the class period when it is covered. The class will generally cover material in the same order as the text book, but there may be exceptions. These exceptions will be discussed in class and will be the order material is covered in my PowerPoint slides, which will be posted on the class website. It is the student's responsibility be aware of what material in the text that is currently being covered. Ask me if you are unsure of the text material currently being covered. A pop quiz may cover topics from the reading assignment for the current class.

attendance

Regular attendance is important in this class. There is a strong correlation between class attendance and grade performance in this class. On the rare occasions that you cannot avoid being absent, you are responsible for getting class notes and assignments. While class slides posted on the class website will provide information about the material covered in class, these will not include some important information such as discussion of points on slides, class discussions, and information written on the board. To receive an excused absence, you must either contact me in advance and receive permission to be absent or else present documented evidence of illness or of an individual/family emergency situation. Routine doctor or dental appointments are not an excuse

	for missing a scheduled examination. A death in the immediate family requires a confirmation of the death and your very close connection to that individual. Either excessive class absences (i.e,., more than 3 unexcused absences) or failure to turn in your homework on time (unless excused) can severely damage your homework & class participating grade. A poor grade in this category can lower your grade by a full letter.
class registration	The official registration deadline for this course is February 1, 2009. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.
cheating & plagiarism	Cheating and plagiarism of any type will not be tolerated and will be dealt with in accordance to the University's Administrative Policy and Procedures Regarding Student Cheating and Plagiarism (condensed version). Unattributed copying from another webpage is also considered plagiarism. The Computer Science Department's Academic Policy involving Programming provides information about what is considered to be plagiarism or cheating regarding writing programs.
sanctions	See academic sanctions for a list of the approved sanctions for student cheating or plagiarism.
disability and assessibility statement	University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).
miscellaneous	Please turn off any wireless phones, beepers, or other noise making devices before class begins. Please be considerate, obviously it is a distraction to the rest of the class when one of these devices begins making noise or conversations occur. If you need to leave class early please inform the instructor before class begins and, if possible, sit near the door. If you have any problems, including understanding the material that we cover in class or using the computer, please talk to me.

Structure of Programming Languages
Fall 2008 and Spring 2009

Instructor: Dr. Arvind Bansal
Address: Room 214, Department of Computer Science
Phone: (216)672- 9035
E-mail: arvind@cs.kent.edu

Motivation

The motivation behind teaching this course is to prepare you for better programming and writing compilers, and to prepare you for tomorrow since languages keep changing while the basic principle and design philosophy does not alter. This course will provide basic understanding of general design issues and behavior of different class of programming languages. No specific programming language syntax will be discussed. Although, examples will use syntax of popular languages in a specific class of languages. In addition, the student will be taught latest concepts in multimedia languages such as synchronization issues, multimedia archival and retrieval issues, and multimedia formats.

Prerequisites

Intermediate programming, Data structures, Symbolic programming, Knowledge of at least couple of programming languages

Contents

Introduction and properties of a good programming language (75 minutes), control flow diagrams (20 minutes), syntax and Backus Naur form (120 minutes), introduction to semantics (45 minutes), parameter passing, side-effect, aliasing (200 minutes)

Pre-First Mid Term tutorial (45 minutes)

First Mid Term (1 lecture)

Implementation of Heap Based Languages and Garbage Collection (135 minutes lectures) Types as sets (100 minutes), use of types and abstract implementation (45 minutes), polymorphism (45 minutes), Data dependency, concurrency and synchronization (120 minutes),

Pre-Second Mid Term tutorial (45 minutes)

Second Mid Term (1 lecture)

Deterministic and non-deterministic programming and languages (80 minutes). Fundamentals of functional programming paradigm (90 minutes), Fundamentals of logic programming paradigm (135 minutes), Fundamentals of object oriented programming

paradigm (45 minutes), Implementation Models of object oriented languages <45 minutes>

Pre-Final tutorial (45 minutes)

Last three lectures (not covered in the examination) Agent Based Languages (45 minutes) , Introduction to Multimedia Languages such as XML and SMIL (45 minutes) Introduction to theory of multimedia systems such as synchronization, multimedia archival and retrieval.

Assignments: Last assignment will be a bonus assignment, and will be counted to promote border line students to a higher grade

1. Control flow diagrams, Syntax, and semantics
2. Behavior of languages, parameter passing, and side effects
3. Type theory , concurrency, and non-deterministic programming
4. Heap and Garbage collection
5. Functional and logic programming paradigm, Object Oriented programming paradigm - 2 assignments

For each assignment you will be given exactly one week. Departmental policy will be followed regarding copying. Two very similar text (to be decided by the instructor) will be treated as case of copying. However, group discussion to understand the problem is encouraged. A student will loose 10% of grade for every working day of late submission. Without proper explanation, late assignment submission is discouraged.

Text Book and Reference Material

1. Concepts of Programming Languages by Robert Sebesta, Eighth Edition, Publisher: Addision Wesley,
2. Arvind Bansal, transparencies from the class

Other Reference Books

1. Material for garbage collection and functional programming will be augmented during the semester.
2. Programming Language Design Concepts, by David Watt, Wiley Publishers, 2004
3. Programming Languages Principles and Paradigms, By Noonan and Tucker, Second Edition, McGraw Hill, 2007

Grading Policy

There will be three examinations: first midterm (20 %), second midterm (20 %), and third midterm (30%), and six assignments. First Five assignments will be counted. Last assignment will be a bonus assignment to help students who are at the border of two grades. Each assignment carries 6% of the grade. A student must attend 85% of the classes.

A > 85%, B > 75%, C > 65%, D > 50%

University Plagiarism Policy

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied. The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students. "Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation.

Cheating includes, but is not limited to:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;
2. Using notes, textbooks or other information in examinations, tests and quizzes except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;
6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;
9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements
of the present course or program without notifying the instructor to whom the work is presented; and
10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes or other academic work.

"Plagiarize" means to take and present as one's own a material portion of the ideas or words of another person or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;
b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and
c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

STUDENT CHEATING AND PLAGIARISM

Academic Sanctions

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. **The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:**

1. Refuse to accept the work for credit; or
2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism takes place; or
3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;
4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

For information regarding the academic appeals procedure, please refer to page 107 of the 2008-2009 FlashGuide

CS 44201 - Artificial Intelligence
Fall 2009

Instructor: Dr. Arvind Bansal
Address: Room 214, Department of Computer Science
Phone: (216)672- 9035
E-mail: arvind@cs.kent.edu

What is AI?

Artificial Intelligence (AI), also referred as computational intelligence, is mimicking human intelligence and reasoning using computational techniques. In early 1970s, the scope of artificial intelligence was limited to mimicking the knowledge based reasoning of a specialist and intelligent guessing (heuristics) for smart game playing. However, as the area of computational intelligence has matured, it has diverged to many areas such as expert systems, decision support systems, shape and object recognition, pattern recognition and data mining, knowledge based systems, content based retrieval, case based reasoning, symbolic systems, neural networks, intelligent agent based system, common sense reasoning, decision support systems, constrained based reasoning, speech recognition, robotics, motion planning, collaborative reasoning, inductive learning, deductive reasoning, and their integration.

Application

AI has been extensively used in diagnostics systems, simulating what-if scenarios, game playing, decision support systems with human in loop to assist humans in decision making, automated control of machines, automobiles, aircrafts, spacecrafts, intelligent industrial machines for specific functions, learning new trends from stock market and traffic and sales understanding systems for multimedia objects, more recently in phone based automated voice interactions, intelligent routing of the messages in the computer networks, human like human-computer interaction, and development of humanoids to assist aging population as in Japan.

Motivation

The motivation of this course is to help you learn various AI techniques and concepts as explained above. The course will teach you above concepts and programming in popular AI languages such as Lisp, Scheme, Prolog, and the implementation of the above concepts using these languages.

Course Outline

Introduction (1 lecture), Lisp programming (2 lectures), Prolog programming (2 lectures), Programming practice (3 lectures)

** First Mid term – Take Home Exam**

Heuristics and State space problems and Searching (3 lectures), Constraint satisfaction and propagation (2 lectures), Forward Chaining and Backward
Chaining Systems (1 lectures), Bayesian network and pattern matching (1 lecture), Tutorial (1 lecture)

** Second Mid Term **

Neural Networks (1 lectures), Semantic Networks, Knowledge Bases and Ontology (2 lectures), Belief and Plans (1 lecture), Hidden Markov Model (1 lecture) Learning (1 lectures), Perception (1 lecture), Tutorial (1 lecture)
** Final **

Text
Artificial Intelligence, A Modern Approach, Third Edition, Year: 2008, Authors: Russell and Norvig, Publisher: Prentice Hall, ISBN: 0-13-790395-2

Grading
There will be five mini-projects and a team project. There will be three midterms. The total of mini-projects/assignments would be 20%, the total of team project would be 20%, First midterm will be 20%, second midterm will be 20%, and the finals would be 20%. The project would be an implementation of a small group project using Prolog or Lisp to write a game or a reasoning system using blackboard architecture or an expert system or simulating a neural network or developing a Hidden Markov Model or developing semantic network and pulling out information using semantic network or using semantic network to identify an object. The projects would involve whole lot of self study and discussion with the professor.
A > 85%, B > 75%, C > 65%, D > 50%

Copying Policy
Professor will treat the students as mature adults who are seriously interested in learning the course material. Copying is not conducive to learning. If you feel that you are unable to meet your grade obligations, talk to the professor to help you out instead of copying. Please read the university policy and the department policy regarding plagiarism given along with the syllabus very carefully. Taking any material from the web site is also plagiarism, and should be avoided.

You are allowed to discuss the projects and assignments in study groups to understand the involved concepts. However, the homework problems must be attempted and programmed only by you and not by others.

Kent State University
CS 4/59995: ST: Introduction to Data Mining
Fall 2009

Professor: Yuri Breitbart, MSB 251

Mondays and Wednesdays 9:15-10:30 am; Rm. MSB 228
Office Hours: Mondays and Wednesdays 11:00-12:00pm and by appointment

---

Prerequisites

CS 33001 Data Structures
Or Consent of the Instructor

Course Overview

The course presents the concepts and techniques of data mining. Data Mining is a process of discovering information from a set of large data sets. Many commercial and government organizations have huge databases and files with a lot of information in them. Data Mining has developed a set of techniques to unlock information from these data. Among data mining successes are: discovering patterns of traveler behaviour, discovery of market associations of the "beer and diaper" type, and comparisons of the genotype of people with and without certain medical problems related to presence of specific genes in their genetic structure. Data Mining is an interdisciplinary field that combines methods from statistic, databases, machine learning and neural networks. All necessary information from these fields will be given in class. The major difference between data mining and previous artificial intelligence and statistical methods is in designing scalable methods that applicable to large data that cannot be stored entirely in computer memory. Such methods led to very impotant applications in bioinformatics, medical informatics, market analysis, financial engineering, web searching, and e-commerce e-science among others. In this course we first focus on issues of data extraction and data preparation for data mining. We then analyse basic data mining techniques: association rules, classification, clustering, and mining complex data types. Finally, we apply the learned techniques to specific applications in medicine and market analysis.

Learning Outcome

It is expected that at the end of the course students will learn basic data mining techniques and examples of application of these techniques to specific application data. The students will also learn the ways to collect data from data warehouses and design data models amenable to scalable data mining techniques.

Textbook

Margaret Duhnam

Data Mining: Introductory and Advanced Topics

Additional Text

Data Mining: Concepts and Techniques by Jiawei Han and Michele Kamber

Weekly Course Outline

Weeks	Topics	Reading Material
1	Introduction to Data Mining	Ch 1
2	Topics Related to Data Mining	Ch 2
3	Data warehousing and Data Preprocessing	Ch 2 of additional textbook
4	Data Preparation and Discretization	Ch 3 of additional textbook
5	Introduction to Data Mining Techniques	Ch 3
6	Characterization and Comparison	Ch 5 of additional textbook
7-9	Classification Algorithms	Ch4
10-11	Association Rules	Ch 6
12-13	Clustering	Ch 5

Exams

Students will be asked to do some homeworks. Each homework contains examples of application of data mining algorithms as well as questions about the alternative algorithms for some data mining algorithms. There will be a midterm on October 14, 2009. Project due date is December 7, 2009. This a firm date and no extension will be granted. There will be final exam on December 15th, 2009 at 10:15. Each project will be either implementation of comparisons between two data mining algorithms discussed in class. or analysis of additional literature assigned by instructor.

Requirements & Grading Policy

University Policies

A student's grade is determined as a weighted average of homeworks (20%), project (20%), midterm (25%), and final exam (35%).

Check This place for New Class Materials

Homeworks

Projects

Viewgraphs

The official registration deadline for this course is 09/13/2009. University policy requires all students to be officially registered for each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for

the course. Each student must confirm enrollment by checking her/his class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline. The last day to withdraw is 11/08/2009.

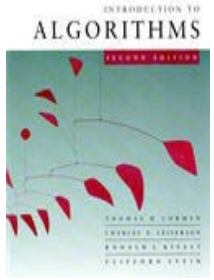
Advanced Algorithms - CS 6/76101

Textbook: Introduction to Algorithms, McGraw Hill Publishing Company and MIT Press, 2001 (2nd Edition).

Topics	Slides per Page
Dynamic Programming	**1 (ps) (pdf), 2 (ps) (pdf)**
Greedy Algorithms	**1 (ps) (pdf), 2 (ps) (pdf)**
Amortized Analysis	**1 (ps) (pdf), 2 (ps) (pdf)**
Parallel Algorithms	**1 (ps) (pdf), 2 (ps) (pdf)**
Computational Geometry	**1 (ps) (pdf), 2 (ps) (pdf)**
NP-Completeness:1	**1 (ps) (pdf), 2 (ps) (pdf)**
NP-Completeness:2	**1 (ps) (pdf), 2 (ps) (pdf)**
Approximation Algorithms	**1 (ps) (pdf), 2 (ps) (pdf)**
Exact Solutions to NP-Complete Problems	**1 (ps) (pdf), 2 (ps) (pdf)**
Network Flow and Matching	**1 (ps) (pdf), 2 (ps) (pdf)**

Reading

by Thomas Cormen, Charles Leiserson, Ronald Rivest, and Cliff Stein



The MIT Press web site for the textbook is http://mitpress.mit.edu/algorithms, and it contains a link to the list of known bugs and errata.

EXAMS

1. **Thursday, October 16, 2008, 3:45-05:00 pm** {Dynamic Programming, Greedy Algorithms, Amortized Analysis, Parallel Algorithms} info results
2. **Thursday, December 11, 2008, 7:45-10:00 a.m.**{Computational Geometry, NP-Completeness, Approximation Algorithms, Max-Flow and Maximum Matching} info results

FINAL GRADES

Student Presentations

December 2: Du, Xiaoxi; Manna, Soumyajit; Nor, Rizal M.;

1. The P versus NP Problem .
2. PRIMES is in P

Demos

1. A Stable Marriage Applet
2. An Applet of SkipList
3. An Applet of Max Flow
4. An Applet of Line Sweeping Algorithm
5. An Applet of Graham's scan (Convex Hull Algorithm)

Venkataramana, Shilpa; Abu-Ata, Muad M., Hoblos, Jalaa.
December 4: Khasawneh, Samer F., Lee, Victor E., Liu, Lin, Ruan, Ning, Zaber, Moinul I.

Survey Paper Due Date: December 2 (Survey Topics)

F. Dragan
dragan at cs.kent.edu
Fall 2008

Advanced Algorithms - CS 6/76101
Fall 2008

TR 3:45 - 5:00 PM
MSB 276

Instructor	Dr. Feodor Dragan	**Teaching Assistant**	??????????
Office Hours	Room 254 MSB TR 2:30 - 3:30 PM and by appointment	**Office Hours**	Room ??? MSB ?? ?????? and by appointment
Email	dragan at cs.kent.edu	**Email**	????? (at) cs.kent.edu
Telephone	(330) 672-9058	**Telephone**	(330) 672-9???

- **WWW** http://www.cs.kent.edu/~dragan/CS6-76101-AdvAlg.html
- **Prerequisites**
  A firm prerequisite is a senior level algorithms course at approximately the same level as CS 4/56101 (Intro. to Design & Analysis of Algorithms). A combined Data Structures and Algorithms course like CS 33001 is not an adequate preparation for this course. This course assumes both CS 33001 and CS 4/56101 as background.
- **Goals:** This course has several goals.
  - Study important algorithm areas and techniques not normally covered in the first course.
  - Develop ability to design efficient algorithms.
  - Develop ability to prove correctness and evaluate efficiency of algorithms.
  - To cover a number of additional topics not covered in CS 4/56101 (first algorithms course).
- **Text (CLRS)**
  Thomas **C**ormen, Charles **L**eiserson, Ronald **R**ivest, and Cliff **S**tein, *Introduction to Algorithms*, McGraw Hill Publishing Company and MIT Press, 2001 (2nd Edition).
  (The MIT Press web site for the textbook is http://mitpress.mit.edu/algorithms, and it contains a link to the list of known bugs and errata.)
- **Additional Book & Web References**
  - Ellis **H**orowitz, Sartaj **S**ahni, and Sangurthevar **R**ajasekaran, *Computer Algorithms*, Computer Science Press, 1998. (**HSR**)
  - Gilles **B**rassard and Paul **B**ratley, *Fundamental of Algorithmics*, Prentice Hall, 1996. (**BB**) (used in Prel.Exams)
  - Rajeev Motwani and Prabhakar Raghavan, *Randomized Algorithms*, Cambridge University Press, 1995.
  - J. O'Rourke, *Computational Geometry in C*, Cambridge University Press, 1998 (Second Edition).
  - Michel Goemans, *Advanced Algorithms Course Notes* are available from this website.
- **Some assumed topics from first algorithms course**
  - Reasonable knowledge of basic topics in **CLRS** textbook.
  - Some maturity in designing, evaluating running time, and proving correctness of algorithms.
  - Asymptotic notation and complexity (Ch. 2-3 of **CLRS**).

- Recurrences and summations (Ch. 4 and Appendix A in **CLRS**).
- Major Sorts (including quicksort and heapsort), linear time sorts, lower bound for sorting (Ch. 6-9 of **CLRS**).
- Hash tables (Ch. 11 of **CLRS**).
- Binary search trees, red-black trees (Ch. 12-13 of **CLRS**).
- Graph algorithms including traversals, minimum spanning trees, shortest path (Ch. 22-25 of **CLRS**).
- Basic numeric routines, including matrix operations (Ch. 29,31 of **CLRS**).

- **Topics will include:** We will cover the following topics (the topics and order listed are tentative and subject to change; some topics may only be quickly surveyed to add breadth, while others will be covered in reasonable depth).
  - Dynamic Programming
  - Optimal greedy algorithms
  - Amortized analysis
  - Parallel and circuit algorithms
  - Network flow algorithms
  - Randomized algorithms
  - Number theoretic & cryptographic algorithms
  - String matching algorithms
  - Computational geometry algorithms
  - Algorithms for NP-hard and NP-complete problems
  - Approximation algorithms
  - Online algorithms
  - Linear programming algorithms
- **Course Requirements**

Homework	-	-	-	40%
Midterm Exam	???????	October ??, 2008	3:45 - 5:00 pm	30%
Final Exam	Thursday	December 11, 2008	7:45 - 10:00 a.m.	30%

- **Milestone for successful completion of the course**
  - Attend the classes regularly,
  - Perform the homework thoroughly and independently,
  - Read the book carefully and several times.
- **Make-up and Late policy:** Attendance at times exams are given is a course requirement. Missed tests are only excused if absence was essential and can be fully documented. Unexcused late homework is normally not accepted after 11:59:99 pm of due date. Class extensions on homework will be announced in class. They may also be announced by email and at the course website.
- **Homework and Collaboration:** You will need to devote a considerable amount of time to homework. You may discuss the homework with other students, but you must write your solutions independently. Study groups should limit their size to 2-3 so that each collaborator can participate in solution. If you obtain a solution to a homework problem through research (e.g., from books or journals), you are expected to acknowledge your sources in your writeup and also to write up your solution

independently.

- **Plagiarism:** Copying the solution from another student or jointly writing up the solution of a problem constitutes plagiarism. You are not permitted to use solutions to assigned problems from earlier terms. Such activities and other unapproved or anti-intellectual behavior violates the University's plagiarism rules and can result in severe penalties. Behavior of this type is unfair to both yourself (in missed learning opportunities) and to the other students. University rules on plagiarism are given here.
- **Student Accessibility:** University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. **Please note, you must first verify your eligibility for these through Student Accessibility Services** (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).
- **Registration Requirement:** The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

F. Dragan
dragan at cs.kent.edu
Fall 2008

Design & Analysis of Algorithms - CS 4/56101

T R 05:30 pm -06:45 pm
MSB 115

Presentations

Chapters	Topics	Reading
Instead of Introduction	By Jeff Edmonds	
Chapter 1: Algorithm Analysis	Analysis of Algorithms	Sections: 1.1,1.2,1.3,1.4
Chapter 2: Basic Data Structures	· Elementary Data Structures · Vectors · Stacks · Queues · Sequences · Trees · Heaps · Priority Queues · Hash Tables · Dictionaries	· Sections: 2.1,2.2,2.3 · Sections: 2.2.1 · Sections: 1.5,2.1.1 · Sections: 2.1.2 · Sections: 2.2.2,2.2.3 · Sections: 2.3.1 - 2.3.4 · Sections: 2.4.1,2.4.3 · Sections: 2.4.1 - 2.4.4 · Sections: 2.5.1 - 2.5.6 · Sections: 3.1.1 - 3.1.6
Chapter 3: Search Trees and Skip Lists	· Binary Search Trees · Red-Black Trees	· Sections: 3.1.1 - 3.1.6 · Sections: 3.3.3
Chapter 4: Sorting, Sets, and Selection	· Merge Sort · Quick Sort · Sorting Lower Bound	· Sections: 4.1.1 · Sections: 4.3.1, 4.6 · Sections: 4.4

Textbook: Algorithm Design: Foundations, Analysis, and Internet Examples, Wiley, 2002, ISBN 0-471-38365-1

by
Michael T. Goodrich, Univ. of California, Irvine
and
Roberto Tamassia, Brown Univ.

Book's web site

HOMEWORKS

- Homework #1: distributed on 9/9/2008, due 9/16/2008 results
- Homework #2: distributed on 9/16/2008, due 9/23/2008 results

	· Sets · Radix Sort · Selection	· Sections: 4.2.1 · Sections: 4.5.1, 4.5.2 · Sections: 4.7, 4.7.1 - 4.7.3
Chapter 5: Fundamental Techniques	· Greedy Method · Divide-and-Conquer · Dynamic Programming	· Sections: 5.1,5.1.1, 5.1.2 · Sections: 5.2, 5.2.1 - 5.2.3 · Sections: 5.3, 5.3.1 - 5.3.3
Chapter 6: Graphs	· Graphs · Depth-First Search · Breadth-First Search · Biconnectivity · Directed Graphs	· Sections: 6.1- 6.2 · Sections: 6.3.1 · Sections: 6.3.3 · Sections: 6.3.2 · Sections: 6.4.1, 6.4.2, 6.4.4
Chapter 7: Weighted Graphs	· Shortest Paths · Minimum Spanning Tree	· Sections: 7.1, 7.1.1-7.1.3, 7.2, 7.2.1 · Sections: 7.3, 7.3.1-7.3.3
Chapter 8: Network Flow and Matching	· Network Flow	· Sections: 8.1.1, 8.1.2, 8.2.1-8.2.3
Chapter 9: Text Processing	· Pattern Matching	· Sections: 9.1.1 - 9.1.4

Demos

· Minimum Spanning Tree
· Shortest Path by Dijkstra's Algorithm
· A Stable Marriage Applet
· An Applet of SkipList
· An Applet of Max Flow
· An Applet of Line Sweeping Algorithm
· An Applet of Graham's scan (Convex Hull Algorithm)

- Homework #3: distributed on 9/23/2008, due 9/30/2008 results
- Homework #4: distributed on 9/30/2008, due 10/7/2008 results
- Homework #5: distributed on 10/21/2008, due 10/28/2008 results
- Homework #6: distributed on 10/28/2008, due 11/13/2008 results
- Homework #7: distributed on 11/6/2008, due 11/??/2008 results
- **EXTRA CREDIT PROBLEM DUE 11/20/08**

EXAMS

- **October, 16, 2008, 5:30 pm - 6:45 pm** (All material covered before Chapter 5) Info Overview Results
- Tuesday, Dec. 9, 2008, 5:45 - 8:00 p.m. Info Overview Results

Review of the material for

	the Final Exam is on Thursday, Dec. 4 ***FINAL GRADES***

F. F. Dragan
dragan at cs dot kent dot edu
Fall 2008

Design & Analysis of Algorithms - CS 4/56101
FALL 2008

T R 05:30 pm -06:45 pm

MSB 115

Instructor **Office Hours**	Dr. Feodor Dragan Room 254 MSB TR 2:30 – 3:30 PM and by appointment	**Teaching Assistant** **Office Hours**	Jamal Alsakran Room 140 MSB TR 2:30 – 3:30 PM and by appointment
Email **Telephone**	dragan at cs dot kent dot edu (330) 672-9058	**Email** **Telephone**	**jalsakra@cs.kent.edu** (330) 672-7059

- **WWW:** http://www.cs.kent.edu/~dragan/CS4-56101-D&AofAlg.html
- **Prerequisites:** Data Structures - CS 33001
- **Course Description:** This course is an introductory undergraduate/graduate course on the design and analysis of algorithms. The course builds on the study of the analysis and implementation of data structures and algorithms from CS 33001. The goal is to introduce a number of important algorithm design techniques as well as basic algorithms that are interesting both from a theoretical and also practical point of view. We will cover basic algorithm design techniques such as divide-and-conquer, dynamic programming, and greedy techniques for optimization. We will cover techniques for proof of the correctness of algorithms and also asymptotic analysis of algorithm time bounds by the solution of recurrence equations. We will apply these design and analysis techniques to derive algorithms for a variety of tasks such as sorting, searching, and graph problems. Some specific algorithm topics include: deterministic and randomized sorting and searching algorithms, depth and breadth first search graph algorithms for finding paths and matchings, and algebraic algorithms for fast multiplication and linear system solving.
- **Goals:** This course has several goals.
  - Study important data structures and algorithmic techniques not normally covered in CS 33001.
  - Develop ability to design efficient algorithms.
  - Develop ability to prove correctness and evaluate efficiency of algorithms.

  The *main goal* of the course is to learn to think algorithmically like a ``real'' computer scientist.

- **Textbook**

  Algorithm Design: Foundations, Analysis, and Internet Examples, Wiley, 2002, ISBN 0-471-38365-1
  (by Michael T. Goodrich and Roberto Tamassia) Book's web site

- **Topics will include:** We will cover the following topics (the topics and order listed are tentative and subject to change; some topics may only be quickly surveyed to add breadth, while others will be covered in reasonable depth).
  - *Algorithm Analysis*
  - *Data Structures* (Elementary Data Structures, Vectors, Stacks, Queues, Sequences, Trees, Heaps, Priority Queues, Hash Tables, Dictionaries)

- *Searching* (Binary Search Trees, AVL Trees, Red-Black Trees, Skip Lists, Locators)
- *Sorting* (Merge Sort, Quick Sort, Sorting Lower Bound, Sets, Radix Sort, Selection)
- *Fundamental Techniques* (Greedy Method, Divide-and-Conquer, Dynamic Programming)
- *Graphs Algorithms* (Graphs, Depth-First Search, Breadth-First Search, Biconnectivity, Directed Graphs)
- *Weighted Graphs* (Shortest Paths, Minimum Spanning Tree)
- *Network Flow and Matching*
- *Text Processing* (Pattern Matching)

- **Course Requirements**

Homework	-	-	-	40%
Midterm Exam	???????	????????, 2008	5:30 - 6:45 pm	30%
Final Exam	Tuesday	Dec. 9, 2008	5:45 - 8:00 p.m.	30%

Homework is very important. It is expected that most of your learning will come from the process of solving the homework problems. Exams will in large part be based on the homework.

- **Milestone for successful completion of the course**
  - Attend the classes regularly,
  - Perform the homework thoroughly and independently,
  - Read the book carefully and several times.
- **Make-up and Late policy:** Attendance at times exams are given is a course requirement. Missed tests are only excused if absence was essential and can be fully documented. Unexcused late homework is normally not accepted after 11:59:99 pm of due date. Class extensions on homework will be announced in class. They may also be announced by email and at the course website.
- **Homework and Collaboration:** You will need to devote a considerable amount of time to homework. You may discuss the homework with other students, but you must write your solutions independently. Study groups should limit their size to 2-3 so that each collaborator can participate in solution. If you obtain a solution to a homework problem through research (e.g., from books or journals), you are expected to acknowledge your sources in your writeup and also to write up your solution independently.
- **Plagiarism:** Copying the solution from another student or jointly writing up the solution of a problem constitutes plagiarism. You are not permitted to use solutions to assigned problems from earlier terms. Such activities and other unapproved or anti-intellectual behavior violate the University's plagiarism rules and can result in severe penalties. Behavior of this type is unfair to both yourself (in missed learning opportunities) and to the other students. University rules on plagiarism are given here.
- **Student Accessibility:** University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. **Please note, you must first verify your eligibility for these through Student Accessibility Services** (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).
- **Registration Requirement:** The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

F. F. Dragan
dragan at cs dot kent dot edu
Fall 2008

Design & Analysis of Algorithms - CS 4/56101

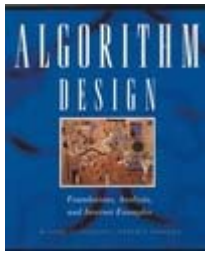
MW 12:30 pm - 01:45 pm
MSB 115

Presentations

Chapters	Topics	Reading
Instead of Introduction	By Jeff Edmonds	
Chapter 1: Algorithm Analysis	Analysis of Algorithms	Sections: 1.1,1.2,1.3,1.4
Chapter 2: Basic Data Structures	· Elementary Data Structures · Vectors · Stacks · Queues · Sequences · Trees · Heaps · Priority Queues · Hash Tables · Dictionaries	· Sections: 2.1,2.2,2.3 · Sections: 2.2.1 · Sections: 1.5,2.1.1 · Sections: 2.1.2 · Sections: 2.2.2,2.2.3 · Sections: 2.3.1 - 2.3.4 · Sections: 2.4.1,2.4.3 · Sections: 2.4.1 - 2.4.4 · Sections: 2.5.1 - 2.5.6 · Sections: 3.1.1 - 3.1.6
Chapter 3: Search Trees and Skip Lists	· Binary Search Trees · Red-Black Trees	· Sections: 3.1.1 - 3.1.6 · Sections: 3.3.3
Chapter 4: Sorting, Sets, and Selection	· Merge Sort · Quick Sort · Sorting Lower Bound	· Sections: 4.1.1 · Sections: 4.3.1, 4.6 · Sections: 4.4

Textbook: Algorithm Design: Foundations, Analysis, and Internet Examples, Wiley, 2002, ISBN 0-471-38365-1

by
Michael T. Goodrich, Univ. of California, Irvine
and
Roberto Tamassia, Brown Univ.



Book's web site

HOMEWORKS

- Homework #1: distributed on 2/2/2009, due 2/9/2009 results
- Homework #2: distributed on 2/9/2009, due 2/16/2009 results
- Homework #3:

	· Sets · Radix Sort · Selection	· Sections: 4.2.1 · Sections: 4.5.1, 4.5.2 · Sections: 4.7, 4.7.1 - 4.7.3
Chapter 5: Fundamental Techniques	· Greedy Method · Divide-and-Conquer · Dynamic Programming	· Sections: 5.1,5.1.1, 5.1.2 · Sections: 5.2, 5.2.1 - 5.2.3 · Sections: 5.3, 5.3.1 - 5.3.3
Chapter 6: Graphs	· Graphs · Depth-First Search · Breadth-First Search · Biconnectivity · Directed Graphs	· Sections: 6.1- 6.2 · Sections: 6.3.1 · Sections: 6.3.3 · Sections: 6.3.2 · Sections: 6.4.1, 6.4.2, 6.4.4
Chapter 7: Weighted Graphs	· Shortest Paths · Minimum Spanning Tree	· Sections: 7.1, 7.1.1-7.1.3, 7.2, 7.2.1 · Sections: 7.3, 7.3.1-7.3.3
Chapter 8: Network Flow and Matching	· Network Flow	· Sections: 8.1.1, 8.1.2, 8.2.1-8.2.3
Chapter 9: Text Processing	· Pattern Matching	· Sections: 9.1.1 - 9.1.4

Demos

· Minimum Spanning Tree
· Shortest Path by Dijkstra's Algorithm
· A Stable Marriage Applet
· An Applet of SkipList
· An Applet of Max Flow
· An Applet of Line Sweeping Algorithm
· An Applet of Graham's scan (Convex Hull Algorithm)

distributed on 2/16/2009, due 3/2/2009 results
- Homework #4: distributed on 2/25/2009, due 3/9/2009 results
- Homework #5: distributed on 4/6/2009, due 4/13/2009 results
- Homework #6: distributed on 4/13/2009, due 4/20/2009 results
- Homework #7: distributed on 4/20/2009, due 4/29/2009 results

Extra credit problem: distributed on 4/15/2009, due 4/27/2009. results

EXAMS

- **Wednesday, March 18, 2009, 12:30 pm - 01:45 pm** (All material covered before Chapter 5) Info Overview Results MidTerm Review is on Monday, March 16, 2009.
- **Wednesday, May 13, 2009, 10:15 am - 12:30**

	pm Info Overview Results Final Exam Review is on Wednesday, May 6, 2009.

F. F. Dragan
dragan at cs dot kent dot edu
Spring 2009

Design & Analysis of Algorithms - CS 4/56101
SPRING 2009

MW 12:30 pm - 01:45 pm

MSB 115

Instructor **Office Hours**	Dr. Feodor Dragan Room 254 MSB MW 1:45 – 2:15 pm, M 3:30 – 5:00 pm and by appointment	**Teaching Assistant** **Office Hours**	Yang Xiang Room 352 MSB T 11:00 am-12:30pm, W 2:00-3:30pm and by appointment
Email **Telephone**	dragan at cs dot kent dot edu (330) 672-9058	**Email** **Telephone**	yxiang@cs.kent.edu (330) 672-9106

- **WWW:** http://www.cs.kent.edu/~dragan/CS4-56101-D&AofAlg.html
- **Prerequisites:** Data Structures - CS 33001
- **Course Description:** This course is an introductory undergraduate/graduate course on the design and analysis of algorithms. The course builds on the study of the analysis and implementation of data structures and algorithms from CS 33001. The goal is to introduce a number of important algorithm design techniques as well as basic algorithms that are interesting both from a theoretical and also practical point of view. We will cover basic algorithm design techniques such as divide-and-conquer, dynamic programming, and greedy techniques for optimization. We will cover techniques for proof of the correctness of algorithms and also asymptotic analysis of algorithm time bounds by the solution of recurrence equations. We will apply these design and analysis techniques to derive algorithms for a variety of tasks such as sorting, searching, and graph problems. Some specific algorithm topics include: deterministic and randomized sorting and searching algorithms, depth and breadth first search graph algorithms for finding paths and matchings, and algebraic algorithms for fast multiplication and linear system solving.
- **Goals:** This course has several goals.
  - Study important data structures and algorithmic techniques not normally covered in CS 33001.
  - Develop ability to design efficient algorithms.
  - Develop ability to prove correctness and evaluate efficiency of algorithms.

  The *main goal* of the course is to learn to think algorithmically like a ``real'' computer scientist.

- **Textbook**

  Algorithm Design: Foundations, Analysis, and Internet Examples, Wiley, 2002, ISBN 0-471-38365-1
  (by Michael T. Goodrich and Roberto Tamassia) Book's web site

- **Topics will include:** We will cover the following topics (the topics and order listed are tentative and subject to change; some topics may only be quickly surveyed to add breadth, while others will be covered in reasonable depth).
  - *Algorithm Analysis*
  - *Data Structures* (Elementary Data Structures, Vectors, Stacks, Queues, Sequences, Trees, Heaps, Priority Queues, Hash Tables, Dictionaries)
  - *Searching* (Binary Search Trees, AVL Trees, Red-Black Trees, Skip Lists, Locators)
  - *Sorting* (Merge Sort, Quick Sort, Sorting Lower Bound, Sets, Radix Sort, Selection)
  - *Fundamental Techniques* (Greedy Method, Divide-and-Conquer, Dynamic Programming)
  - *Graphs Algorithms* (Graphs, Depth-First Search, Breadth-First Search, Biconnectivity, Directed Graphs)
  - *Weighted Graphs* (Shortest Paths, Minimum Spanning Tree)
  - *Network Flow and Matching*
  - *Text Processing* (Pattern Matching)
- **Course Requirements**

Homework	-	-	-	40%
Midterm Exam	**???????**	**March ??, 2009**	**12:30 pm - 01:45 pm**	30%
Final Exam	**Wednesday**	**May 13, 2009**	**10:15 am - 12:30 pm.**	30%

Homework is very important. It is expected that most of your learning will come from the process of solving the homework problems. Exams will in large part be based on the homework.

- **Milestone for successful completion of the course**

  - Attend the classes regularly,
  - Perform the homework thoroughly and independently,
  - Read the book carefully and several times.
- **Make-up and Late policy:** Attendance at times exams are given is a course requirement. Missed tests are only excused if absence was essential and can be fully documented. Unexcused late homework is normally not accepted after 11:59:99 pm of due date. Class extensions on homework will be announced in class. They may also be announced by email and at the course website.
- **Homework and Collaboration:** You will need to devote a considerable amount of time to homework. You may discuss the homework with other students, but you must write your solutions independently. Study groups should limit their size to 2-3 so that each collaborator can participate in solution. If you obtain a solution to a homework problem through research (e.g., from books or journals), you are expected to acknowledge your sources in your writeup and also to write up your solution independently.
- **Cheating and Plagiarism:** Copying the solution from another student or jointly writing up the solution of a problem constitutes plagiarism. You are not permitted to use solutions to assigned problems from earlier terms. Such activities and other unapproved or anti-intellectual behavior violate the University's plagiarism rules and can result in severe penalties. Behavior of this type is unfair to both yourself (in missed learning opportunities) and to the other students. University rules on plagiarism are given here and on sanctions are here.
- **Student Accessibility:** University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. **Please note, you must first verify your eligibility for these through Student Accessibility Services** (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).
- **Registration Requirement:** The official registration deadline for this course is Feb. 1, 2009. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

F. F. Dragan
dragan at cs dot kent dot edu
Spring 2009

ST: Advanced Algorithms for Graphs and Networks - CS 6/79995

(Research oriented Course)

Spring 2009

MW 02:15 pm - 03:30 pm, MSB 276

Instructor	Dr. Feodor Dragan
Office	Room 254 MSB
Hours	MW 1:45 – 2:15 PM, M 3:30 – 5:00 pm and by appointment
Email	dragan at cs kent edu
Telephone	(330) 672-9058

Final grades: http://www.cs.kent.edu/~dragan/ST/ST-Grades.pdf

Student Presentations Schedule http://www.cs.kent.edu/~dragan/Stud-Pres-ST.pdf

Topics and Literature

Centers, medians, centroids, diameters of trees and other graphs

1. Goldman, A.J., Optimal center location in simple networks, Transportation Science, 5, 212-221 (1971)
2. Tancel, B.C., Francis, R.L., Lowe, T.J.: Location on Networks, Parts 1,2. Management Science, 29, 482-511 (1983)
3. Chepoi V.D. and Dragan F.F., Linear-time algorithm for finding a central vertex of a chordal graph, "Algorithms - ESA'94 " Second Annual European Symposium, Utrecht, The Netherlands, September 1994, Springer, LNCS 855 (Jan van Leeuwen, ed.), 159-170, 1994.
4. V.D. Chepoi, F.F. Dragan, B. Estellon, M. Habib and Y. Vaxes, Diameters, centers, and approximating trees of δ-hyperbolic geodesic spaces and graphs, Proceedings of the 24th Annual ACM Symposium on Computational Geometry (SoCG 2008), June 9–11, 2008, College Park, Maryland, USA, pp. 59-68.

p-Centers, r-domination, p-medians of trees and other graphs

5. O. Kariv and S. L. Hakimi: An Algorithmic Approach to Network Location Problems. I: The p-Centers, SIAM J. Appl. Math., Volume 37, Issue 3, pp. 513-538 (1979)
6. O. Kariv and S. L. Hakimi: An Algorithmic Approach to Network Location Problems. II: The p-Medians, SIAM J. Appl. Math., Volume 37, Issue 3, pp. 539-560 (1979)
7. J. Plesnik: A heuristic for the p-center problem in graphs, Discrete Applied Mathematics, Volume 17, Issue 3, Pages: 263 - 268 (1987)
8. Brandstädt, Chepoi V.D. and Dragan F.F.: The algorithmic use of hypertree structure and maximum neighbourhood orderings, Discrete Appl. Math., 82 (1998), 43-77.
9. Book: D. Hochbaum (ed.) “Approximation Algorithms for NP-hard Problems” PWS Publ.Co. 1995

Self-stabilizing algorithms for centers and medians of trees

10. G. Antonoiu and P. K. Srimani, A Self-Stabilizing Distributed Algorithm to Find the Median of a Tree Graph, Journal of Computer and System Sciences, Volume 58, Issue 1, February 1999, Pages 215-221.

11. S. C. Bruell, S. Ghosh, M. H. Karaata, and S. V. Pemmaraju, Self-Stabilizing Algorithms for Finding Centers and Medians of Trees, SIAM J. Comput., Volume 29, Issue 2, pp. 600-614 (1999)

12. S. M. Hedetniemi, S. T. Hedetniemi, D. P. Jacobs and P. K. Srimani, Self-stabilizing algorithms for minimal dominating sets and maximal independent sets. Computers & Mathematics with Applications, Volume 46, Issues 5-6, September 2003, Pages 805-811.

Labeling schemes: NCA-queries, adjacency, ancestry

13. S. Kannan, M. Naor, S. Rudich, Implicit representation of graphs, Proceedings of the twentieth annual ACM symposium on Theory of Computing, Pages: 334 - 343 (1988).

14. Book: J. Spinrad, Efficient Graph Representations, Fields Institute Monographs, 19, 2003.

15. N. Santoro and R. Khatib, Labelling and Implicit Routing in Networks, The Computer Journal, 28(1):5-8 (1985)

16. Serge Abiteboul, Haim Kaplan, Tova Milo, Compact labeling schemes for ancestor queries, Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms, Pages: 547 - 556, 2001

17. Haim Kaplan, Tova Milo, Ronen Shabo, A comparison of labeling schemes for ancestor queries, Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms, Pages: 954 - 963, 2002

18. Stephen Alstrup, Cyril Gavoille, Haim Kaplan, Theis Rauhe: Nearest common ancestors: a survey and a new distributed algorithm, Proceedings of the fourteenth annual ACM symposium on Parallel algorithms and architectures, Pages: 258 - 264, 2002; also: http://www.it-c.dk/people/stephen/Papers/ITU-TR-2001-6.ps

Labeling schemes: connectivity, reachability, distance

19. Book: D. Peleg, Distributed Computing: A Locality-Sensitive Approach, SIAM Monographs on Discrete Mathematics and Applications, 2000.

20. David Peleg: Proximity-Preserving Labeling Schemes and Their Applications. WG 1999: 30-41

21. David Peleg: Informative labeling schemes for graphs. Theor. Comput. Sci. 340(3): 577-593 (2005)

22. V.D. Chepoi, F.F. Dragan, Y. Vaxès, Addressing, distances and routing in triangular systems with applications in cellular networks, Wireless Networks 12 (2006), 671-679.

23. Cyril Gavoille, David Peleg, Stéphane Pérennes, Ran Raz: Distance labeling in graphs. J. Algorithms 53(1): 85-112 (2004)

24. Cyril Gavoille, Michal Katz, Nir A. Katz, Christophe Paul, David Peleg: Approximate Distance Labeling Schemes. ESA 2001: 476-487.

25. Slivkins, Distance estimation and object location via rings of neighbors, PODC 2005, pp. 41--50.

26. Mikkel Thorup, Uri Zwick: Approximate distance oracles. STOC 2001: 183-192

27. Michal Katz, Nir A. Katz, Amos Korman, David Peleg: Labeling Schemes for Flow and Connectivity. SIAM J. Comput. 34(1): 23-40 (2004)

28. Mikkel Thorup: Compact Oracles for Reachability and Approximate Distances in Planar Digraphs. FOCS 2001: 242-251

29. Amos Korman, David Peleg: Compact separator decompositions in dynamic trees and applications to labeling schemes. Distributed Computing 21(2): 141-161 (2008)

Network design: minimum spanning trees, diameter-bounded minimum spanning trees, light approximate shortest path trees, Steiner trees, optimal communication spanning trees

30. Book: B.Y. Wu and K.-M. Chao "Spanning Trees and Optimization Problems", CRC Press, 2003

31. Book: J. Cheriyan and R. Ravi, Approximation Algorithms for Networks Problems, http://www.gsia.cmu.edu/afs/andrew/gsia/ravi/WWW/new-lecnotes.html

32. David Peleg, Eilon Reshef: Deterministic Polylog Approximation for Minimum Communication Spanning Trees. ICALP 1998: 670-681.

33. David Peleg: Low Stretch Spanning Trees. MFCS 2002: 68-80.

34. Noga Alon, Richard M. Karp, David Peleg, Douglas B. West: A Graph-Theoretic Game and Its Application to the k-Server Problem. SIAM J. Comput. 24(1): 78-100 (1995)

Graph Spanners

35. Book: D. Peleg, Distributed Computing: A Locality-Sensitive Approach, SIAM Monographs on Discrete Mathematics and Applications, 2000. (the same as 19)
36. D. Peleg, A.A. Schaffer, Graph spanners, Journal of Graph Theory 13 (1) (1989) 99–116.
37. V.D. Chepoi, Dragan F.F., and Chenyu Yan, Additive Sparse Spanners for Graphs with Bounded Length of Largest Induced Cycle Theoretical Computer Science 347 (2005), 54-75.
38. Y. Dourisboure, Dragan F.F., C. Gavoille, and C. Yan, Spanners for bounded tree-length spanners, Theoretical. Computer Science 383 (2007), 34-44.
39. Victor Chepoi, Feodor F. Dragan, Bertrand Estellon, Michel Habib, Yann Vaxes and Yang Xiang, Additive Spanners and Distance and Routing Labeling Schemes for delta-Hyperbolic Graphs, manuscript 2008.
40. M. Elkin, D. Peleg, (1 + epsilon, beta)-spanner constructions for general graphs, 33rd Annual ACM Symposium on Theory of Computing, STOC, Hersonissos, Crete, Greece, 2001, pp. 173–182.
41. S. Baswana, T. Kavitha, K. Mehlhorn, S. Pettie, New constructions of (alpha, beta)-spanners and purely additive spanners, 16th Symposium on Discrete Algorithms, SODA, ACM–SIAM, 2005, pp. 672–681.
42. Althofer, G. Das, D. Dobkin, D. Joseph, J. Soares, On sparse spanners of weighted graphs, Discrete & Computational Geometry 9 (1) (1993) 81–100.
43. M. Thorup, U. Zwick, Approximate distance oracles, 33rd Annual ACM Symposium on Theory of Computing, STOC, Hersonissos, Crete, Greece, 2001, pp. 183–192. (the same as 26)
44. S. Baswana, S. Sen, A simple linear time algorithm for computing a $(2k-1)$-spanner of $O(n^{1+1/k})$ size in weighted graphs, 30th International Colloquium on Automata, Languages and Programming, ICALP, in: Lecture Notes in Computer Science, vol. 2719, Springer, 2003, pp. 384–396.
45. Bilel Derbel, Cyril Gavoille, David Peleg, Laurent Viennot: On the locality of distributed sparse spanner construction. PODC 2008: 273-282.
46. Li, X.-Y., Wang, Y.: Geometrical Spanner for Wireless Ad Hoc Networks. Handbook of Approximation Algorithms and Metaheuristics (Editor: Teofilo F. Gonzalez), Chapman & Hall/Crc (2006)
47. Gao, J., Guibas, L.J., Hershberger, J., Zhang, L., Zhu, A.: Geometric spanner for routing in mobile networks. Proceedings of the 2nd ACM international symposium on mobile ad hoc networking & computing, October 04-05, 2001, Long Beach, CA, USA
48. F.F. Dragan, F. Fomin and P. Golovach, A PTAS for the sparsest spanners problem on apex-minor-free graphs, 33rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2008), Torun, Poland, August 27-31, 2008, Springer, Lecture Notes in Computer Science 5162, pp. 290-298.
49. F.F. Dragan, F. Fomin and P. Golovach, Spanners in sparse graphs, 35th International Colloquium on Automata, Languages and Programming (ICALP 2008), Reykjavik, Iceland, July 6-13, 2008, Springer, Lecture Notes in Computer Science 5125 (Part I), pp. 597-608.

Tree spanners, collective tree spanners, flow spanners

50. L. Cai, D.G. Corneil, Tree spanners, SIAM Journal on Discrete Mathematics 8 (3) (1995) 359–387.
51. Brandstädt, Dragan F.F., H.-O. Le, and V.B. Le, Tree Spanners on Chordal Graphs: Complexity and Algorithms, Theoretical Computer Science 310 (2004), 329-354.
52. Brandstädt, Dragan F.F., H.-O. Le, V.B. Le and R. Uehara, Tree spanners for bipartite graphs and probe interval graphs, Algorithmica 47 (2007), 27–51.
53. Stefan Eckhardt and Sebastian Wernicke, On the Algorithmic Intractability of Computing Distance-Approximating Spanning Trees, Preprint: http://wwwbib.informatik.tu-muenchen.de/infberichte/2004/TUM-I0409.pdf.gz
54. Yuval Emek, David Peleg: Approximating Minimum Max-Stretch spanning Trees on unweighted graphs. SODA 2004: 261-270.
55. Christian Liebchena and Gregor Wünsch, The zoo of tree spanner problems, Discrete Applied Mathematics, Volume 156, Issue 5, 1 March 2008, Pages 569-587.
56. F.F. Dragan, C. Yan and I. Lomonosov, Collective tree spanners of graphs, SIAM J. Discrete Math. 20 (2006), 241-260.
57. F.F. Dragan, C. Yan and D.G. Corneil, Collective Tree Spanners and Routing in AT-free Related Graphs, Journal of Graph Algorithms and Applications, Vol. 10, no. 2, 2006, 97-122.
58. F.F. Dragan and C. Yan, Network Flow Spanners, Proceedings of the 7th Latin American Symposium "LATIN 2006: Theoretical Informatics", Valdivia, Chile, March 20-24, Springer, Lecture Notes in Computer Science 3887, pp. 410-422.

Navigating in a graph: Message routing, hierarchical cluster-based routing, interval routing, labeling routing schemes, routing in mobile ad-hoc networks, pseudo coordinates, navigating with aid.

59. C. Gavoille, A survey on interval routing schemes, Theoretical Computer Science, 245 (1999), 217--253.
60. Thorup, M., Zwick, U.: Compact routing schemes. In: Proceedings of the 13th Ann. ACM Symp. on Par. Alg. and Arch. (SPAA 2001), pp. 1–10.
61. Y. Dourisboure, Compact Routing Schemes for Bounded Tree-Length Graphs and for k-Chordal Graphs, DISC 2004, 365--378.
62. Victor Chepoi, Feodor F. Dragan, Bertrand Estellon, Michel Habib, Yann Vaxes and Yang Xiang, Additive Spanners and Distance and Routing Labeling Schemes for delta-Hyperbolic Graphs, manuscript 2008. (the same as 36)
63. Abraham, C. Gavoille, A.V. Goldberg, D. Malkhi, Routing in Networks with Low Doubling Dimension, ICDCS 2006, p. 75 (see also http://dept-info.labri.fr/~gavoille/article/AGGM06).
64. H. T.-H. Chan, A. Gupta, B. M. Maggs, and S. Zhou, On hierarchical routing in doubling metrics, Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2005), Vancouver, British Columbia, Canada, January 23-25, 2005. SIAM, pp. 762--771.
65. Slivkins, Distributed approaches to triangulation and embedding, Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2005), Vancouver, British Columbia, Canada, January 23-25, 2005, SIAM, pp. 640-649.
66. Slivkins, Distance estimation and object location via rings of neighbors, PODC 2005, pp. 41--50. (the same as 25)
67. K. Talwar, Bypassing the embedding: Algorithms for low dimensional metrics, STOC 2004, pp. 281--290.
68. S. Giordano and I. Stojmenovic, Position based routing algorithms for ad hoc networks: A taxonomy, In X. Cheng, X. Huang, and D. Du, editors, Ad Hoc Wireless Networking, pages 103-136. Kluwer, 2004.
69. Karp and H.T. Kung, GPSR: greedy perimeter stateless routing for wireless networks, Proceedings of the 6th ACM/IEEE MobiCom., 2000, ACM Press, pp. 243-254.
70. J.M. Kleinberg, The small-world phenomenon: an algorithm perspective, Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing (STOC 2000), May 21-23, 2000, Portland, OR, USA. ACM, pp. 163-170.
71. Rao, C. Papadimitriou, S. Shenker, and I. Stoica, Geographical routing without location information, Proceedings of MobiCom 2003, 2003, pp. 96--108.
72. R. Kleinberg, Geographic routing using hyperbolic space, In INFOCOM 2007, pp. 1902-1909.
73. F.F. Dragan and M. Matamala, Navigating in a graph by aid of its spanning tree, 19th International Symposium on Algorithms and Computation (ISAAC 2008), Surfers Paradise, Gold Coast, Australia, December 15-17, 2008, Springer, Lecture Notes in Computer Science 5369, pp. 789–800, 2008.

Balanced separators of graphs and their algorithmic use

74. Nathan Linial, Eran London, Yuri Rabinovich: The Geometry of Graphs and Some of its Algorithmic Applications. Combinatorica 15(2): 215-245 (1995)
75. F.F. Dragan and C. Yan, Collective Tree Spanners in Graphs with Bounded Genus, Chordality, Tree-width, or Clique-width, In Proceedings of the 16th Annual International Symposium on Algorithms and Computation (ISAAC 2005), December 19-21, 2005, Hainan, China, Springer, Lecture Notes in Computer Science 3827, pp. 583–592.
76. Gupta, A., Kumar, A., Rastogi, R.: Traveling with a Pez Dispenser (Or, Routing Issues in MPLS). SIAM J. Comput., 34 (2005), pp. 453–474.
77. V.D. Chepoi, F.F. Dragan, Y. Vaxès, Distance and routing labeling schemes for non-positively curved plane graphs, Journal of Algorithms 61 (2006), 60-88.
78. F.F. Dragan, D.G. Corneil, E. Köhler and Y. Xiang, Additive Spanners for Circle Graphs and Polygonal Graphs, 34th International Workshop on Graph-Theoretic Concepts in Computer Science (WG 2008), Durham University, U.K., June 30- July 2, 2008, Springer, Lecture Notes in Computer Science 5344, pp. 110–121, 2008.
79. Abraham, C. Gavoille, Object location using path separators, Proceedings of the Twenty-Fifth Annual ACM Symposium on Principles of Distributed Computing (PODC 2006)}, Denver, Colorado, USA, July 23-26, 2006, ACM, pp 188-197.
80. Chenyu Yan, Yang Xiang and Feodor F. Dragan, Compact and Low Delay Routing Labeling Scheme for Unit Disk Graphs, manuscript 2008.
81. See also 20, 21, 23, 28.

F. F. Dragan
dragan at cs kent edu
Spring 2009

ST: Advanced Algorithms for Graphs and Networks - CS 6/79995

(Research oriented Course)

Spring 2009

MW 02:15 pm - 03:30 pm

MSB 276

Instructor	Dr. Feodor Dragan
Office	Room 254 MSB
Hours	MW 1:45 – 2:15 PM, M 3:30 – 5:00 pm and by appointment
Email	dragan at cs kent edu
Telephone	(330) 672-9058

- **Outline:** The course will focus on advanced algorithms for practical problems arising in Communication Networks and Analysis of Data. This is a graduate level and *research oriented* course.
- **Course Content:** We will have discussions on some of the following topics and more.
  - Location problems: *centers, medians, centroids, diameter, p-centers, r-domination, p-medians*
  - Labeling schemes: *NCA-queries, adjacency, connectivity, reachability, distance, routing*
  - Network design: *minimum spanning trees, diameter-bounded minimum spanning trees, light approximate shortest path trees, Steiner trees, optimal communication spanning trees, tree covers and spanners, tree spanners, collective tree spanners, flow spanners*
  - Navigating in a graph: *Message routing, hierarchical cluster-based routing, interval routing, labeling routing schemes, routing in mobile ad-hoc networks, pseudo coordinates, navigating with aid.*
  - Balanced separators *of graphs and their use.*
- **Prerequisites**
  CS 4/56101 (Design and Analysis of Algorithms).
- **Supplementary Texts**
  - Lecture notes
  - D. Peleg, Distributed Computing: A Locality-Sensitive Approach, SIAM Monographs on Discrete Mathematics and Applications, 2000.
  - J. Spinrad, Efficient Graph Representations, Fields Institute Monographs, 19, 2003.
  - B.Y. Wu and K.-M. Chao "Spanning Trees and Optimization Problems", CRC Press, 2003
  - J. Cheriyan and R. Ravi, Approximation Algorithms for Networks Problems, http://www.gsia.cmu.edu/afs/andrew/gsia/ravi/WWW/new-lecnotes.html
  - D. Hochbaum (ed.) "Approximation Algorithms for NP-hard Problems" PWS Publ.Co. 1995
  - T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein "Introduction to Algorithms", MIT Press, 2001 (2nd edition)
  - A number of recent journal and conference papers.
- **Course Requirements:** Research project, presentations and final exam-quiz (12:45 - 3:00 p.m. Wed. May 13).
- **Cheating and Plagiarism:** Copying the solution from another student or jointly writing up the solution of a problem constitutes plagiarism. You are not permitted to use solutions to assigned problems from earlier terms. Such activities and other unapproved or anti-intellectual behavior violate the University's plagiarism rules and can result in severe penalties. Behavior of this type is unfair to both yourself (in missed learning opportunities) and to the other students. University rules on plagiarism are given here and on sanctions are here.
- **Student Accessibility:** University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. **Please note, you must first verify your eligibility for these through Student Accessibility Services** (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).
- **Registration Requirement:** The official registration deadline for this course is Feb. 1, 2009. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm

enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline

---

F. Dragan
dragan at cs kent edu
Spring 2009

CS 6/73304 Cluster Computing

Fall 2008

Course: 6/73304	Fall 2008
Call Number: 14487/14488	
Time: 5:30 pm - 6:45pm Tu Th	Location: 276 MSB
Paul A. Farrell	Office : 258
Phone : 672-9060	Mail address : farrell@cs.kent.edu
Office Hours : TBA and by appointment	

Description: This course will investigate *clusters of computers* as a computing platform for distributed computing. By the end of the course, each student should understand specific hardware and software tradeoffs for cluster and application performance, and the issues involved in deployment and programming of a cluster of computers. Each student will complete a project based on cluster computing and/or cluster deployment and management. Course material will come from texts and assigned readings.

Prerequisites: CS 43201 Operating Systems, and, either CS45101 Computer Organization & Architecture or CS45201 Computer Communication Networks, or equivalent; or permission required

CS 6/73995 Parallel and Distributed Computing is complementary to this course and would provide a useful background.

Recommended Text: Beowulf Cluster Computing with Linux, 2nd edition, edited by William Gropp, Ewing Lusk, Thomas Sterling, MIT Press, 2003

Optional Texts:
MPI The Complete Reference - 2nd Ed by Marc Snir, et. al., The MIT Press, 1998.
Parallel Programming with MPI by Peter Pacheco, Morgan Kaufmann, 1998.
Using MPI-2, Advanced Features of the Message Passing Interface, William Gropp, Ewing Lusk, Rajeev Thakur, The MIT Press, 1999.
In Search of Clusters: The ongoing battle in lowly parallel computing, Second Edition, by Gregory F. Pfister, Prentice Hall Publishing Company, 1998.
How to Build a Beowulf – A Guide to the Implementation and Application of PC Clusters, by Thomas Sterling, John Salmon, Donald J. Becker and Daniel F. Savarese, MIT Press, 1999

Course Outline: Topics to be covered include:

- A general introduction to the concept of cluster based distributed computing.
- Hardware technologies for cluster computing, including a survey of the possible node hardware and high-speed networking hardware and software.
- Software for cluster computing
- Software and software architectures for cluster computing, including both shared memory (OpenMP) and message-passing (MPI/PVM) models
- Programming, features and performance of standard MPI variants (LAM/MPICH/vendor specific MPI versions), Derived data types, communicators.
- MPI-2 extension, dynamic process creation, one-sided communication, parallel I/O.
- Variants based on new low level protocols (MVAPICH), evaluation and tuning of system and software performance
- Linux for clusters, cluster monitoring.

- Performance evaluation tools, HINT, netperf, netpipe, ttcp, Iperf.
- Configuring and Tuning Clusters: This will involve evaluation of the performance of various nodes and networking hardware such as Gigabit Ethernet, Myrinet, Infiniband, Quadrics etc.

and possibly some of the following topics:

- Security
- Troubleshooting
- Setting up Clusters: OSCAR, NPCAI Rocks, Score etc.
- Managing cluster resources: single system images, system level middleware, distributed task scheduling, monitoring and administering system resources
- Parallel I/O and Parallel Virtual File System
- Scheduling: Condor, Maui Scheduler, Portable Batch System (PBS)
- Application steering and visualization: Cumulvs, GUIs for visualization and debugging
- Brief overview of meta-clustering: the Computational Grid, Globus, Grid Portals, Java RMI, Jini

Notes:

- Assignments will be a mix of cluster installation and administration tasks, performance evaluation, programming, and written questions.
- Some of the assigments will consist in programming in C using MPI (and possibly OpenMP or PVM).
- There will be a penalty for late assignments which may amount to up to 10% per day.
- Assignments are to be completed by the student without assistance from or collaboration with other persons, unless otherwise specifiied.
- The final examination is nominally scheduled for 5:45 - 8:00 p.m. Tuesday December 9

Additional Requirements:

- This syllabus and all subsequent information on the course will be available using the WWW. The home page for the course is: http://www.cs.kent.edu/~farrell/cc08/index.html
- All programs should conform to the submission standards given in URL http://www.cs.kent.edu/~farrell/cc08/submission.html

Cheating and Plagiarism Students are reminded of the University policy on Cheating and Plagiarism - a condensed version of which is available in this document.

Students with Disabilities: University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

Fall 08
CS 10051 – 600 * Introduction to Computer Science

Department of Computer Science
Kent State University Stark

TR 11:00 AM - 12:15 PM - Room MH 304 (Instructor: Dr. Angela Guercio)

Laboratory: F 11:00 AM - 1:00 PM - Room MH 306 (Instructor: Prof. Judith Edwards)

Class Instructor	Lab Instructor
Dr. Angela Guercio	Dr. Judith Edwards
Office: 424, Main Hall	**Office:** 310G desk 12, Main Hall
Phone: 330 244-3424 (KSU ext. 53424)	**Phone**: 330 244-3424 (KSU ext. 53319)
Best way to contact me: e-mail to aguercio@kent.edu	**Best way to contact me:** e-mail to jedwar9@kent.edu
Office Hours: **TR** 10:00am - 10:55am 12:25pm - 1:55pm 4:55pm - 5:25pm other times are available by appointment	**Office Hours:** **TR** 4:00pm – 5:15pm and F 1:15pm – 2:30pm Other times are available by appointment

Course Information

Class Webpage: http://www.personal.kent.edu/~aguercio/Fall08/CS10051-600Fa08.html

- all important class information will be posted on the class webpage, readings, assignments, notes, deadlines, cancellations, ect..
- You must **CHECK THE CLASS WEBSITE REGULARLY!!!**

Prerequisites: No prerequisites

Credit: 4

Required Text:

G. M. Schneider, J. Gersting – *An Invitation to Computer Science, C++ Version* – Thompson, 4th Edition, 2006.

Emergency: In case of an emergency please contact the security on campus.

Security phone on campus: #53123

Security cell phone (330) 705-0430 or, of course, 911.

I recommend that you program into your cell phone the previous numbers.

Course Outline and Objectives

This course will introduce you to the computer science discipline. The course covers

- the algorithmic foundations of computer science by introducing the concept of algorithm, algorithm design, the efficiency of algorithms;
- the hardware world by introducing binary numbers, Boolean logic, gates and circuits, and computer organization;
- virtual machines and computer networks;
- the software world by introducing high level language programming and the use of compilers.

The objectives of the course are:

- to introduce you to the basic terminology of the Computer Science discipline;
- to expose you to the foundation of this discipline and to show you the ideas and principles that helped its formation;
- to show what can be done and what cannot be done in computing;
- to introduce the most important elements of computing;
- to expose you to the basic elements of programming and to provide an experimental approach to the computer science discipline;
- to deepen your writing ability on scientific issues in computing;
- to improve your ability to read and understand computing material;
- to develop in you a familiarity with computing elements and to enable to use them for future courses;
- to show alternative solutions to computer science problems and discuss the complexity of the solutions;
- to provide you with hands-on experience in computing;
- to develop in you an appreciation for the interesting features of this discipline.

Class Requirements and Expectations

1. **Regular class attendance is REQUIRED.**

There tends to be a strong correlation between class attendance and grade performance. If you will miss a class or a lab**, let me know ahead of time**. In any case, you are responsible for bringing yourself up to date on class material and assignments.

- Since class participation and regular attendance are part of the final grade**, <u>if you miss more than 5 classes without a documented reason or without making prior arrangements with me, your final grade will be dropped one grade (A to B, B+ to C+ and so on).</u>**

2. Laboratory attendance is MANDATORY.

- Lab activity **MUST** be started in class and can be completed at home only with instructor permission. Laboratory worksheets will not be accepted if you are absent during the laboratory. <u>Labs completed at home without instructor permission or health professional's excuse will not be accepted</u>.

3. COMPLETE the laboratory activity.

- Laboratory activity is issued weekly and must be completed in class.
- For each lab activity you will be asked to perform a Lab Experiment and to complete a Lab Worksheet.

4. **Reading ahead is REQUIRED**.

- The readings are posted online on the class webpage. You must read the material **<u>before</u>** class **<u>and again after</u>** the class.
- Regular study of the material is REQUIRED. We will roughly cover ½ to 1 chapter per week.

5. COMPLETE the assigned homework.

- Regular homework assignments will be given and they will be posted online on the class webpage.
- The class webpage will list the assignments for each week at the beginning of that week so that you can better schedule your work.

6. REVIEW the graded Homework/Labs.

- Homework and Labs will be graded and some difficulties will be discussed in class. Review the mistakes.
- Late Homework/Lab Reports will not be accepted if returned after the solution is given or discussed.

7. Return work ON TIME

- All the printed copies of the Homework are due ***before or at the beginning of class***. All assignments, either printed or submitted via e-mail, turned in one day late will get **3 points per day penalty** including those returned after the beginning of class.

- For all Homework that are e-mailed, the instructor will acknowledge the receipt within 24 hours via e-mail. The time of your e-mail will be compared against the work deadline. The reply is your receipt that the work has been turned in (not that it is correct!). If you do not receive a receipt, it is YOUR responsibility to contact me to see if the assignment has been lost in transmission. ***Important***: once you submit your files **DO NOT OPEN THEM AGAIN!** If your e-mail didn't reach me or something happened to your files, I may need to ask you to resubmit your files by logging on in my presence to check the modification dates on your files and make sure that they haven't been modified after the due date.

What to expect to find in your computer science class

- The class should be interactive. In-class questions and exercises are designed to encourage participation. There will be in class cooperation, open discussions about problems and possible solutions.

- You will be exposed to traditional lecture methods on the blackboard as well as computer presentations and hands-on activities. Handouts will be given when necessary, but in general PowerPoint slides of the lectures will be available. In any case, you are responsible for taking good notes.

- You will participate in group activities. Collaborative learning will be used to discuss possible solutions to problems as well as to provide critical observations to problem solutions. Formal and/or informal groups will be formed in class to stimulate team work. In some cases, you will be required to work on your own. In those cases, I expect appropriate academic behavior from you. Exchange of information, when forbidden, is not appropriate.

- You will work both **with** and **without** a computer. The laboratory activity provides hands-on application of the concepts learned in class and complements the theoretical studies of the computer science discipline. Exercises and problems solved without the computer will help you in developing the ability to discuss and identify the most appropriate techniques for the solution of a problem, and to stress the importance of the development of an optimal design of the solution contrary to a "brute-force" design driven by the specific computer requirements (i.e. the first solution that comes to your mind and that you design directly on the computer!) which is typical of an untrained person.

- ↗ Expect to commit some time each day to study the theory of computer science and to observe, analyze, solve and report the solution of the assigned lab problems.

Some Useful Hints:

- ✓ ***Do not procrastinate! Homework and Labs should be started immediately. You will find out that it requires more time than you have planned!*** Lab experiments and reports will need considerable extra time for completion when errors occur. ***Any error discovered at the last minute might be the cause of an undesired delay, so plan accordingly!***
- ✓ If you have difficulties doing your homework or your Labs, get help from the Instructor, prepare questions for class, or visit my office.
- ✓ If my office hours do not work for you, ask for an appointment.
- ✓ If your difficulties are in writing, get help from the Writing Center.

The Secret Key (not so secret after all!) of how to succeed in this CS class is to:

1. work conscientiously and do all the homework that has been assigned;
2. extrapolate, from the examples provided to you, techniques and answers to problems;
3. spend several hours at the computer to solve problems as well as reading material;
4. be alert and participate in class discussions;
5. learn from other peoples' mistakes;
6. be critical of your own work. Question every step you are making; ask yourself "Is this step correct?" "Are there other easier or more efficient alternative steps?"
7. attend the class and the laboratory regularly;
8. spend time studying the theoretical concepts. Memory helps, but it is practice that reinforces the theory;
9. do all the above consistently through the whole semester, be confident about what you are doing and don't be afraid to ask for help;
10. Think and enjoy!

I am very confident that you can make the above commitment and that you can maintain it during the semester. I am sure that you have all the ability to be successful!

Exams

- There will be 3 100-points Mid-Term Exams which will cover the topics of the previous 4 weeks.

- The 100-points Final Exam is comprehensive and will cover with greater stress the topics of the last 3 weeks of the course.
- All exams are closed books, closed notes.
- Retaking exams are not available.
- Make-up exams will only be given in case of serious need (written verification for your inability to take an exam is required) and only when I have been notified ***prior*** to the exam being issued, otherwise you are considered absent for that exam and the grade of your exam is automatically 0.

Grading

Your grade will be based on

1. Your homework completion
2. Your Lab Activity completion
3. Your participation in discussions concerning the homework, class topics and reading material
4. Your attendance in class and in the lab
5. …and, of course, your exams!

The COURSE is formed of two independent parts.

TO PASS THE COURSE, YOU MUST PASS EACH PART GIVEN BELOW INDEPENDENTLY!

---- i.e. an A in PART II and an F in PART I, is *NOT* a passing grade.

Part I

Labs Attendance and Worksheets 25%

Penalty for late lab report: 3 points a day

Part II

Homework and Class Participation 10%

Penalty for homework: 3 points a day

Exam 1 15%
Exam 2 15%
Exam 3 15%
Final Exam 20%

Points	Grade
92.5-100	A
89.5-92.4	A-
87-89.5	B+
82.5-86.9	B
80-82.4	B-
77.5-79.9	C+
72-77.4	C

70-72	C-
67.5-69.9	D+
60-67.5	D
00-59.9	F

Course Withdrawal

If you are considering withdrawing from this course, please inform your instructor and consult a staff member in the Student Services Office, 134 Main Hall. Withdrawal from a course can affect financial aid, student status, or progress within your major. For withdrawal deadlines, please refer to http://www.registrars.kent.edu/home/TermUpdate/sche_adj.htm.

Academic Honesty Policy

When assignments must be individually and independently done, if some students turn in substantially the same solution or program of another student, in my judgment, the solution will be considered a group effort. All involved in the group effort homework will receive a zero grade for that assignment. Policy on academic dishonesty involving programming can be found at http://www.cs.kent.edu/programs/grad/DishonestyPolicy.pdf.

Use of the intellectual property of others without attributing it to them is considered a serious academic offense. Cheating or plagiarism will result in a failing grade for the work or for the entire course. Repeat offenses result in dismissal from the University. University guidelines require that all infractions be reported to the Student Conduct Officer on our campus.

Students with Disabilities

Kent State University recognizes its responsibility for creating an institution atmosphere in which students with disabilities can succeed. In accordance with University Policy Subpart E…104.44, if you have a documented disability, you may request accommodations to obtain equal access in this class. Please contact the disability coordinator on campus, Kelly Kulick in Student Accessibility Services, located in the Student Success Center, lower level of the Campus Center, phone (330) 244-5047, or kkulick @kent.edu. After your eligibility for accommodations is determined, you will be given a letter which, when presented to instructors, will help us know best how to assist you.

Classes Canceled – Campus Closings

Announcements of class cancellations and/or campus closings will be made on the campus home page. In the case of an emergency, weather-related or otherwise, please check the web page at http://www.stark.kent.edu for information on the buildings and times of the closing. While information may be broadcast by radio and television, this should be confirmed by the web page, which is the official announcement of the campus and which will be the information used to determine issues related to student attendance, rescheduling of tests, and other concerns.

Conduct

Students and faculty behavior at the Stark Campus is governed by the guidelines set forth in *The Digest of Rules and Regulations*. That document can be found in the University telephone directory. Information can be found at the Office of Judicial Affairs at http://www.kent.edu/administration/emsa/judicial.cfm.

Recycling

KSU Stark Campus recycles. Recycling saves energy, which is currently generated by expensive and vanishing fossil fuels. Recycling one aluminum can saves enough energy to run a TV for three hours! Please take a few seconds to separate your trash. Aluminum cans and plastic and glass bottles may be placed in the blue recycling bins, and all types of paper may be placed in the blue recycling trash cans. All other waste may be placed in the black, brown or gray trash cans.

Important Dates to Remember

- Last day to withdraw ***before grade W*** is assigned, is Sept. 7, 2008
- Last day to drop the class is Nov. 2, 2008
- Exam 1 is Tuesday, Sept. 23
- Exam 2 is Tuesday, Oct. 21
- Exam 3 is Thursday, Nov. 13
- Final Exam is Thursday, Dec. 11 (6:00 pm – 8:00pm)

Thanksgiving Recess: Nov. 26 – Nov. 30
Classes End: Dec. 7, 2008

The Course Outline.

- Labs days are highlighted in orange
- Review days are highlighted in green
- Exams days are highlighted in yellow
- More Assignments may be issued according to the needs of the class

	Month/Day	**Topic**	**Reading and Assignments**
1	Aug 26	Introduction to Computer Science: Definition of Algorithm	**Read Chapter 1 and 2.** **First Assignment** **Deadline:** e-mail your assignment by 9:59 a.m. by Aug 28
2	Aug 28	Introduction to Computer Science: Definition of Computer Science. Introduction to Algorithm Design: Pseudocode	There are a few symbols missing in Ch.2 of the new edition. Here is an Errata Corrige (latin for "correct the errors") for you. Please add the missing symbols in your Chapter.
1L	Aug 29	Lab 1	
3	Sept 2	Algorithm Design: More algorithms in Pseudocode with Sequential Operations, Conditional and Iterative Operations, Sequential Search, Find the maximum	**Read ahead Chapter 3.**
4	Sept 4	Algorithm Design: Pattern Matching Algorithm. The efficiency of the algorithms: Sorting	

2L	Sept 5	Lab 2	
5	Sept 9	The efficiency of the algorithms: Data Cleanup	
6	Sept 11	More Data Cleanup Algorithm	
3L	Sept 12	Lab 3	
7	Sept 16	Binary search, Pattern Matching, When things get out of hands	**Read ahead Chapter 4.**
8	Sept 18	Review and Practice - Algorithm Design	
4L	Sept 19	Lab 4	
9	Sept 23	Exam 1 (Ch 1-3)	
10	Sept 25	Binary Numbers, Algorithms for Base Conversion	Assignment 2 due before class on October 7.
5L	Sept 26	Lab 5	
11	Sept 30	Signed Magnitude, Two's complement, Text and Image representation	
12	Oct 2	Review and Practice: Base Conversion and Data Representation	
6L	Oct 3	Lab 6	
13	Oct 7	Boolean Logic - Gates - Truth Tables	
14	Oct 9	CE, Adder, Control Circuits	
7L	Oct 10	Lab 7	
15	Oct 14	Review and Practice – Boolean Logic and Circuits	**Read ahead Chapter 5** Assignment 3 due before class on October 16.
16	Oct 16	Computer Systems Organization: Memory	
8L	Oct 17	Lab 8	
17	Oct 21	Exam 2 (Ch. 4)	
18	Oct 23	Computer Systems Organization: Control Unit	**Read ahead Chapter 6.**
9L	Oct 24	Lab 9	
19	Oct 28	Computer Systems Organization: Computer Architecture	
20	Oct 30	Review and Practice – Memory, CU and Architecture	

10L	Oct 31	Lab 10	
21	Nov 4	System Software and Virtual Machine: Machine Language	
22	Nov 6	Review and Practice – Assembly	
11L	Nov 7	Lab 11	
	Nov 11	**Veterans Day - No Class**	
23	Nov 13	Exam 3	**Read ahead Chapter 8.**
12L	Nov 14	Lab 12	
24	Nov 18	High Level Language Programming	
25	Nov 20	High Level Language Programming	
13L	Nov 21	Lab 13	
26	Nov 25	High Level Language Programming	
	Nov 26-30	**Thanksgiving Recess**	
26	Dec 2	Review and Practice	
27	Dec 4	Review and Practice	
14L	Dec 5	Lab 14	
28	Dec 9	10:30 am - 12:30pm - room MH 304 Final Exam (comprehensive)	

Spring 2009

CS 23021 Section 600
Computer Science I - Programming And Problem Solving

Department of Computer Science
Kent State University Stark

TR 11:00 AM - 12:15 PM - Room MH 306 (Instructor: Dr. Angela Guercio)

Laboratory: F 11:00 AM - 1:00 PM - Room MH 306 (Instructor: Ms. Shannon Steinfadt)

Class Instructor	Lab Instructor
Dr. Angela Guercio	Ms. Shannon Steinfadt
Office: 424, Main Hall	**Office:** 310G desk 4, Main Hall
Phone: 330 244-3424 (KSU ext. 53424)	**Phone**: 330 244-3311 (KSU ext. 53311)
Best way to contact me: e-mail to aguercio@kent.edu	**Best way to contact me:** e-mail to ssteinfa@cs.kent.edu
Office Hours: **TR** 11:30am - 1:30pm 4:35pm -5:15pm **F** 10:15am - 10:55am other times are available by appointment	**Office Hours:** **TR** 11:30am -12:30pm Other times are available by appointment

- **Class Webpage**:
  - http://www.personal.kent.edu/~aguercio/Spring09/CS23021Sp09.html
- **Lab Webpage:**
  - http://www.personal.kent.edu/~aguercio/lab23021/index.html

Course Information

Class Webpage: http://www.personal.kent.edu/~aguercio/Spring09/CS23021Sp09.html

- all important class information will be posted on the class webpage, readings, assignments, notes, deadlines, cancellations, ect..
- You must **CHECK THE CLASS WEBSITE REGULARLY!!!**

Prerequisites: CS10051 with ***a grade of C or better.***
This means that a C- in CS10051 is not sufficient to meet the prerequisite.
For more details, please visit ***http://www.cs.kent.edu/programs/ugrad/planner.html***

Credit: 4 Credit Hours
Required Text: Walter Savitch – *Problem Solving with C++- The Object Oriented Programming*– Pearson Addison Wesley, 7th Edition, 2009, ISBN-10: 0-321-53134-5; ISBN-13: 978-0-321-53134-6. The Online Book Resources can be found at **http://www.aw-bc.com/savitch/** Any other texts or papers that I might require you to read will be given in class.
Emergency: In case of an emergency please contact the security on campus. **Security phone on campus**: #53123 **Security cell phone** (330) 705-0430 or, of course, 911. I recommend that you program into your cell phone the previous numbers.

Course Outline and Objectives

This course will introduce you to the Object Oriented paradigm. This course will teach you how to write programs using the object-oriented paradigm language C++, and will cover the syntax of the language. Particular attention will be paid to program design and the problem-solving methodologies, which should be used to produce a program of good quality.

The course outline covers

- The C++ basic features;
- Procedural Programming and Object Oriented Programming;
- Variables, Data Types and Expressions
- Functions
- Classes and Objects
- Class Properties
- Inheritance
- Arrays and Vectors
- Pointers
- Testing and Debugging

The objectives of the course are:

- To introduce you to the object-oriented paradigm of the C++ language
- To teach you how to write a C++ program and how use the C++ compiler
- To introduce you to the most important elements of computing
- To deepen your program design abilities before proceeding in study of more complex problems and language features.
- To show that there are several ways to solve problems but some solutions are more efficient, better readable and easier to maintain than others. Being a program designer is different from being a 'brute force' programmer: problem-solving methodologies are essential for the scope and the language is the media through which we express those techniques.

- To teach you good programming habits.
- To empower you with the use of data structures.
- To learn how to select methodologies to apply to a series of sample problems. Examples of several classes of problems will be discussed in class.
- To give you hands-on experience in designing and testing C++ programs on different environments.
- To show you the interesting features of C++ such as pointers.
- To satisfy requirements the computer systems major and minor.

ATTENTION!!!

CS23021 is a prerequisite for CS33001. A grade of C or better is required to take CS33001. This means that a C- is not sufficient to meet the requirement.

Class Requirements and Expectations

- **Regular class attendance is REQUIRED.**

  - There tends to be a strong correlation between class attendance and grade performance. If you will miss a class or a lab**, let me know ahead of time**. In any case, you are responsible for bringing yourself up to date on class material and assignments.

  - Since class participation and regular attendance are part of the final grade**, if you miss more than 5 classes without a documented reason or without making prior arrangements with me, your final grade will be dropped one grade (A to B, B+ to C+ and so on).**

- **Laboratory attendance is MANDATORY.**

  - Labs attendance is MANDATORY. Labs **will** be completed partly in class and partly at home. Labs completed ONLY at home (without having completed the required part in class) will not be accepted. Exception will be made only in case you have received a specific instructor permission or you have a health professional's excuse**.**

- **Reading ahead is REQUIRED**.

  - The readings are posted online on the class webpage. You must read the material **before** class **and again after** the class. Roughly we will cover 1 to 2 chapters per week.

  - Regular study of the material is REQUIRED. We will roughly cover 1 chapter per week.

- **COMPLETE the assigned homework (i.e. projects and exercises).**

  - Assignments will be issued on a regular basis and they will be posted online on the class webpage.

  - The class webpage will list the assignments for each week at the beginning of that week so that you can better schedule your work.

  - The projects will require heavy use of the computer and will be time consuming. Please, plan accordingly.

  - Since the course assumes that you have mastered some ability to program, most of the programming activities will be part of your homework. However programming activity will be performed in class whenever possible and compatible with the lecture schedule.

- **REVIEW the graded Homework/Projects/Lab Reports.**

  - Homework, Projects, and Labs will be graded and some difficulties will be discussed in class. Review the mistakes.

  - Late Homework/Projects/Labs will not be accepted if returned after the solution is given or discussed.

  - If you have difficulties doing your homework or your project or your Lab please contact me or come to see me or your Lab instructor. ***Do not procrastinate! Homework, Projects, and Lab should be started immediately.*** You will find out that they will often require more time than you have planned, due to unexpected and unfortunate computer events (which often occur and therefore should be part of your planning)**!**

- **Return work ON TIME**

  - All the homework and project should be zipped and e-mailed as an attachment to aguercio@kent.edu **AND** a printed copy should be returned to the instructor as well.

  - All the printed copies of the Homework or the Projects are due ***before or at the beginning of class***. All assignments, either printed or submitted via e-mail, turned in one day late will get **3 points per day penalty** including those returned after the beginning of class.

  - For all Homework or Projects that are e-mailed, the instructor will acknowledge the receipt within 24 hours via e-mail. The time of your e-mail will be compared against the work deadline. The reply is your receipt that the work has been turned in (not that it is correct!). If you do not receive a receipt, it is YOUR responsibility to contact me to see if the assignment has been lost

in transmission. ***Important***: once you submit your files **DO NOT OPEN THEM AGAIN!** If your e-mail didn't reach me or something happened to your files, I may need to ask you to resubmit your files by logging on in my presence to check the modification dates on your files and make sure that they haven't been modified after the due date.

What to expect to find in your Computer Science II class

- The class should be interactive. In-class exercises are designed to encourage participation. There will be cooperation between you and I, open discussions about problems and possible solutions. You are responsible for taking good notes. Handouts will be given only when necessary.

- You will be exposed to traditional lecture methods on the blackboard as well as PowerPoint presentations. Your will participate in group activities and collaborative learning will be used to discuss possible solutions to problems as well as to provide critical observation to problem solutions. Formal and informal groups will be formed in class to work together. In some cases, you will be required to work on your own. In those cases, I expect appropriate academic behavior from you. Exchange of information, when forbidden, will not be tolerated.

- You will work both with and without a computer. When working with a computer (your homework activity) you will experiment hands-on with the concepts that have been covered in class. The projects are designed to complement the theoretical studies. Exercises of problem analysis and design, without the use of the computer, will reinforce the ability to strive for the optimal design of a problem's solution.

- Expect to commit some time each day to practice the syntax of C++, to study the language, to program and to observe, analyze solve and report the solution of the assigned lab problems.

The Secret Key (not so secret after all!) of how to succeed in this CS class is to:

1. work conscientiously and do all the homework that has been assigned;
2. extrapolate, from the examples provided to you, techniques and answers to problems;
3. spend several hours at the computer to solve problems as well as reading material;
4. be alert and participate in class discussions;
5. learn from other people mistakes;
6. be critical of your own work. Question every step you are making; ask yourself "Is this step correct?" "Are there other easier or alternative and more efficient steps? Did I use the data structure in the appropriate way?"
7. attend class regularly;
8. spend time studying the theoretical concepts. Memory helps, but it is practice that reinforces the theory;
9. do all the above consistently through the whole semester, be confident about what you are doing and don't be afraid to ask for help;
10. Think and enjoy!

I am very confident that you can make the above commitment and that you can maintain it during the semester. I am sure that you have all the ability to be successful!

Exams

- There will be 2 100-points Mid-Term Exams which will cover the topics of the previous 5 weeks.
- The 100-points Final Exam will cover the topics of the last 5 weeks of the course.
- All exams are closed books, closed notes.
- Retake exams are not available.
- Make-up exams will only be given in case of serious need (written verification for your inability to take an exam is required) and only when I have been notified ***prior*** to the exam being issued, otherwise you are considered absent for that exam and the grade of your exam is automatically 0.

Grading

Your grade will be based on

1. Your homework and group projects
2. Your participation in discussions concerning the homework, class topics, and material
3. Your laboratory attendance and activity.
4. Your exams

The COURSE is formed of two independent parts.

TO PASS THE COURSE, YOU MUST PASS EACH PART GIVEN BELOW INDEPENDENTLY!

---- i.e. an A in PART II and an F in PART I, is *NOT* a passing grade.

Part I

Laboratory Attendance and Reports	30%

Penalty for late lab report: 3 points × day

Part II

Homework and Class Participation	10%

Penalty for homework: 3 points × day

Exam 1	20%
Exam 2	20%
Final Exam	20%

Points	Grade
92.5-100	A
89.5-92.4	A-
87-89.5	B+
82.5-86.9	B
79.8-82.4	B-
77-79.8	C+
72-76.9	C
70-72	C-
60-69.9	D
00-59.9	F

Even though I have never found it necessary to change the grading distribution and the number of assignments and exams specified in the syllabus during the semester, I reserve the right to change the method of assigning grades, including changing the number of assignments or exams if I consider it necessary.

Course Withdrawal

If you are considering withdrawing from this course, please inform your instructor and consult a staff member in the Student Services Office, 134 Main Hall. Withdrawal from a course can affect financial aid, student status, or progress within your major. For withdrawal deadlines, please refer to http://www.registrars.kent.edu/home/TermUpdate/sche_adj.htm.

Academic Honesty Policy

When assignments must be individually and independently done, if some students turn in substantially the same solution or program of another student, in my judgment, the solution will be considered a group effort. All involved in the group effort homework will receive a zero grade for that assignment. Policy on academic dishonesty involving programming can be found at http://www.cs.kent.edu/programs/grad/DishonestyPolicy.pdf. A condensed version of the Administrative Policy And Procedures Regarding Student Cheating And Plagiarism has been added to the last page of this syllabus.

Use of the intellectual property of others without attributing it to them is considered a serious academic offense. Cheating or plagiarism will result in a failing grade for the work or for the entire course. Repeat offenses result in dismissal from the University. University guidelines require that all infractions be reported to the Student Conduct Officer on our campus (see Academic Sanctions below).

Students with Disabilities

Kent State University recognizes its responsibility for creating an institution atmosphere in which students with disabilities can succeed. In accordance with University Policy Subpart E…104.44, if you have a documented disability, you may request accommodations to obtain equal access in this class. Please contact the disability coordinator on campus, Kelly Kulick in Student Accessibility Services, located in the Student Success Center, lower level of the Campus Center, phone (330) 244-5047, or kkulick@kent.edu. After your eligibility for accommodations

is determined, you will be given a letter which, when presented to instructors, will help us know best how to assist you.

Classes Canceled – Campus Closings

Announcements of class cancellations and/or campus closings will be made on the campus home page. In the case of an emergency, weather-related or otherwise, please check the web page at http://www.stark.kent.edu for information on the buildings and times of the closing. While information may be broadcast by radio and television, this should be confirmed by the web page, which is the official announcement of the campus and which will be the information used to determine issues related to student attendance, rescheduling of tests, and other concerns.

Conduct

Students and faculty behavior at the Stark Campus is governed by the guidelines set forth in *The Digest of Rules and Regulations*. That document can be found in the University telephone directory. Information can be found at the Office of Judicial Affairs at http://www.kent.edu/administration/emsa/judicial.cfm.

Recycling

KSU Stark Campus recycles. Recycling saves energy, which is currently generated by expensive and vanishing fossil fuels. Recycling one aluminum can saves enough energy to run a TV for three hours! Please take a few seconds to separate your trash. Aluminum cans and plastic and glass bottles may be placed in the blue recycling bins, and all types of paper may be placed in the blue recycling trash cans. All other waste may be placed in the black, brown or gray trash cans.

Important Dates to Remember:

- Last day to withdraw ***before grade W*** is assigned, is Feb 1, 2009
- Last day to drop the class is April 5, 2009
- Exam 1 is Thursday, Feb 19
- Exam 2 is Thursday, March 19
- Final Exam is Tuesday, May 12 (1:00pm – 3:00pm)

Spring Recess: March 23 – March 29
Classes End: May 10, 2009

Tentative Outline of the Course

Jan 20	Introduction to C++
Jan 22	Introduction to C++
Jan 27	Basic C++: Variables and I/O
Jan 29	Basic C++: Control Flow
Feb 3	Applications of C++ Statements
Feb 5	Procedural Abstraction and Functions that return a value
Feb 10	Procedural Abstraction and Functions that return a value
Feb 12	Functions for all Subtasks
Feb 17	Review and practice

Feb 19	**Exam 1**
Feb 24	I/O Streams
Feb 26	I/O Streams
Mar 3	I/O Streams. Classes
Mar 5	Classes
Mar 4	Classes
Mar 10	Abstract Data types: Structures definition, Classes
Mar 12	More Flow of Control
Mar 17	Review and practice
Mar 19	**Exam 2**
Mar 23-29	***Spring Recess***
Mar 31	Loop control. Testing
Apr 2	Graphics
Apr 7	Arrays
Apr 9	Arrays
Apr 14	Review and Practice
Apr 16	Two Dimensional Arrays and C_Strings
Apr 21	Strings and Intro to Vectors
Apr 23	Vectors and Intro to Pointers
Apr 28	Pointers and Dynamic Array
Apr 30	Project Presentations
May 5	Review and Practice
Tuesday, May 12	1:00pm-3:00pm Final Exam (comprehensive)

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING

STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. **Cheating includes, but is not limited to:**

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;
2. Using notes, textbooks or other information in examinations, tests and quizzes except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;
6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;

9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and
10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes or other academic work.

"Plagiarize" means to take and present as one's own a material portion of the ideas or words of another person or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;
b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and
c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

STUDENT CHEATING AND PLAGIARISM: ACADEMIC SANCTIONS

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. **The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:**

1. Refuse to accept the work for credit; or
2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism takes place; or
3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;
4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

For information regarding the academic appeals procedure, please refer to page 107 of the 2008-2009 FlashGuide

Spring 2009

CS 23022 Section 600
Discrete Structures for Computer Science

Department of Computer Science
Kent State University Stark

TR 5:30 PM - 6:45 PM Room MH 219

Professor: Dr. Angela Guercio

Office: 424, Main Hall

Phone: 330 244-3424 (KSU ext. 53424)

Office Hours: **TR** 11:30am - 1:30pm 4:35pm -5:15pm **F** 10:15am - 10:55am

other times are available by appointment

Best way to contact me: e-mail to aguercio@kent.edu

Course Information

Class Webpage: http://www.personal.kent.edu/~aguercio/Spring09/CS23022Sp09.html

- all important class information will be posted on the class webpage, readings, assignments, notes, deadlines, cancellations, etc..
- You must **CHECK THE CLASS WEBSITE REGULARLY!!!**

Prerequisites:

CS10051, MATH 12001 or MATH 11022 or appropriate placement test score into MATH 12002**.** For more details on course dependences, visit *http://www.cs.kent.edu/programs/ugrad/planner.html*

Credit: 3 Credit Hours

Required Text:

Gary Haggard, John Schlipf, Sue Whitesides, ***Discrete Mathematics for Computer Science***, Thomson Brooks/Cole, 2006. ISBN 0-534-49501-X

Emergency: In case of an emergency please contact the security on campus.

Security phone on campus: #53123

Security cell phone (330) 705-0430 or, of course, 911.

I recommend that you program into your cell phone the previous numbers.

Course Description and Objectives

This course will introduce you to the discrete mathematics from the perspective of a computer science scientist. The course focuses on Sets, Relations, Formal Logic, Functions, Analysis of Algorithms and Graph Theory. The objectives of the course is to develop the ability to follow and write mathematical proofs and to strengthen the algorithmic thinking.

Class Requirements and Expectations

- **Regular class attendance is REQUIRED.**

  - There tends to be a strong correlation between class attendance and grade performance. If you will miss a class**, let me know ahead of time**. In any case, you are responsible for bringing yourself up to date on class material and assignments.

  - Since class participation and regular attendance are part of the final grade**, <u>if you miss more than 3 classes without a documented reason or without making prior arrangements with me, your final grade will be dropped one grade (A to B, B+ to C+ and so on).</u>**

- **Reading ahead is REQUIRED**.

  - The readings are posted online on the class webpage. You must read the material **<u>before</u>** class **<u>and again after</u>** the class.

  - Regular study of the material is REQUIRED.

- **COMPLETE the assigned homework (i.e. projects and exercises).**

  - There will be 10 assignments in the course. I will retain the right to change the number of assignments if necessary.

  - The class webpage will list the assignments for each week at the beginning of that week so that you can better schedule your work.

- **REVIEW the graded Homework/Projects.**

  - Homework and Projects will be graded and difficulties will be discussed in class. Review the mistakes after discussion and learn from them.

  - Late Homework/Projects will not be accepted if returned after the solution is given or discussed.

  - If you have difficulties doing your homework or your project please contact me or come to see me. ***Do not procrastinate! Homework and Projects should be started immediately.***

- **Return work ON TIME**

  - All the homework should be printed any time it is possible and e-mailed as an attachment to aguercio@kent.edu. In any case a printed copy or a hand written copy MUST be returned to the instructor as well unless stated otherwise.

  - **All handwritten homework should be readable.** Unreadable homework will NOT be graded! If you realize that the homework you are going to return is not neat, clearly organized and easily readable, please copy it on a new page.

  - All homework, printed or handwritten are due ***before or at the beginning of class***. Late homework will get **3 points per day penalty** including those returned after the beginning of class.

  - For all e-mailed homework, the instructor will acknowledge the receipt within 24 hours via e-mail. The time of your e-mail will be compared against the deadline of your homework. The reply is your receipt that the work has been turned in (not that the homework is correct!). If you do not receive a receipt, it is YOUR responsibility to contact me to see if the assignment has been lost in transmission. ***Important***: once you submit your files **DO NOT OPEN THEM AGAIN!** If your e-mail didn't reach me or something happened to your files, I may need to ask you to resubmit your files by logging on in my presence to check the modification dates on your files and make sure that they haven't been modified after the due date.

What to expect to find in your Computer Architecture class

- The class should be interactive. In-class exercises are designed to encourage participation. There will be cooperation between you and I, open discussions about problems and possible solutions. You are responsible for taking good notes. Handouts will be given only when necessary.

- You will present proofs to other students, your will participate in group activities and collaborative learning will be used to discuss possible solutions to problems as well as to

provide critical observation to problem solutions. Formal and informal groups will be formed in class to work together. In some cases, you will be required to work on your own. In those cases, I expect an appropriate academic behavior from you. Exchange of information, when forbidden, will not be tolerated.

- Expect to commit some times to practice problems. You will find that the reading will go very slow and that you need to reread the material several times before you master it.

The <u>Secret Key</u> (not so secret after all!) of how to succeed in this CS class is to:

1. work conscientiously and do all the homework that has been assigned;
2. extrapolate from the examples provided to you, techniques and answers to problems;
3. spend several hours practicing;
4. be alert and participate in class discussions;
5. learn from other people mistakes;
6. Think! Use your logic and be critical of your own work.
7. attend the class and read the material ahead of time;
8. spend time studying the theoretical concepts. Memory helps, but it is practice that reinforces the understanding of the theory;
9. do all the above consistently through the whole semester, be confident about what you are doing and don't be afraid to ask for help;
10. Think, think, think and enjoy!

I am very confident that you can make the above commitment and that you can maintain it during the semester. I am sure that you have all the ability to be successful!

Exams

- There will be 3 100-points Term Exams which will cover the topics of the previous weeks.
- The 100-points Final Exam will cover the topics of the last 5 weeks of the course.
- All exams are closed books, closed notes.
- Retake exams are not available.
- Make-up exams will only be given in case of serious need (written verification for your inability to take an exam is required) and only when I have been notified ***prior*** to the exam being issued, otherwise you are considered absent for that exam and the grade of your exam is automatically 0.

Pop Quizzes

- Pop Quizzes will be issued. The date of the quiz will NOT be announced. A pop quiz is held during the first 10 minutes of the class. Late students will not be given extra time to complete the quiz. No late quizzes will be accepted; no make-up quizzes.

Grading

Your grade will be based on

1. Your homework
2. Your participation in discussions concerning the homework, class topics, and reading material
3. Your attendance in class
4. Your Pop Quiz results
5. Your exams

Homework	25%
Pop Quizzes&Class Participation	10%
Exam 1	15%
Exam 2	15%
Exam 3	15%
Final Exam	20%

Points	Grade
92.5-100	A
89.5-92.4	A-
87-89.5	B+
82.5-86.9	B
79.8-82.4	B-
77-79.8	C+
72-76.9	C
70-72	C-
60-69.9	D
00-59.9	F

Even though I have never found it necessary to change the grading distribution and the number of assignments and exams specified in the syllabus during the semester, I reserve the right to change the method of assigning grades, including changing the number of assignments or exams if I consider it necessary.

Course Withdrawal

If you are considering withdrawing from this course, please inform your instructor and consult a staff member in the Student Services Office, 134 Main Hall. Withdrawal from a course can affect financial aid, student status, or progress within your major. For withdrawal deadlines, please refer to http://www.registrars.kent.edu/home/TermUpdate/sche_adj.htm.

Academic Honesty Policy

When assignments must be individually and independently done, if some students turn in substantially the same solution or program of another student, in my judgment, the solution will be considered a group effort. All involved in the group effort homework will receive a zero grade for that assignment. Policy on academic dishonesty involving programming can be found at http://www.cs.kent.edu/programs/grad/DishonestyPolicy.pdf. A condensed version of the Administrative Policy And Procedures Regarding Student Cheating And Plagiarism has been added to the last page of this syllabus.

Use of the intellectual property of others without attributing it to them is considered a serious academic offense. Cheating or plagiarism will result in a failing grade for the work or for the entire course. Repeat offenses result in dismissal from the University. University guidelines require that all infractions be reported to the Student Conduct Officer on our campus (see Academic Sanctions below).

Students with Disabilities

Kent State University recognizes its responsibility for creating an institution atmosphere in which students with disabilities can succeed. In accordance with University Policy Subpart E…104.44, if you have a documented disability, you may request accommodations to obtain equal access in this class. Please contact the disability coordinator on campus, Kelly Kulick in Student Accessibility Services, located in the Student Success Center, lower level of the Campus Center, phone (330) 244-5047, or kkulick@kent.edu. After your eligibility for accommodations is determined, you will be given a letter which, when presented to instructors, will help us know best how to assist you.

Classes Canceled – Campus Closings

Announcements of class cancellations and/or campus closings will be made on the campus home page. In the case of an emergency, weather-related or otherwise, please check the web page at http://www.stark.kent.edu for information on the buildings and times of the closing. While information may be broadcast by radio and television, this should be confirmed by the web page, which is the official announcement of the campus and which will be the information used to determine issues related to student attendance, rescheduling of tests, and other concerns.

Conduct

Students and faculty behavior at the Stark Campus is governed by the guidelines set forth in *The Digest of Rules and Regulations*. That document can be found in the University telephone directory. Information can be found at the Office of Judicial Affairs at http://www.kent.edu/administration/emsa/judicial.cfm.

Recycling

KSU Stark Campus recycles. Recycling saves energy, which is currently generated by expensive and vanishing fossil fuels. Recycling one aluminum can saves enough energy to run a TV for three hours! Please take a few seconds to separate your trash. Aluminum cans and plastic and glass bottles may be placed in the blue recycling bins, and all types of paper may be placed in the blue recycling trash cans. All other waste may be placed in the black, brown or gray trash cans.

Important Dates to Remember:

- Last day to withdraw ***before grade W*** is assigned, is Feb 1, 2009
- Last day to drop the class is April 5, 2009

- Exam 1 is Thursday, Feb 19
- Exam 2 is Tuesday, March 10
- Exam 3 is Thursday, April 9
- Final Exam is Thursday, May 14 (6:00pm – 8:00pm)

Spring Recess: March 23 – March 29
Classes End: May 10, 2009

Tentative Outline of the Course

Jan 20	Introduction to the Course - Sets
Jan 22	Sets and Operation on Sets
Jan 27	Proof Templates
Jan 29	The Principle of Inclusion-Exclusion
Feb 3	Mathematical Induction
Feb 5	Application of Induction - Recursion
Feb 10	Strong Form of Mathematical Induction and Application
Feb 12	Program Correctness
Feb 17	Review and Practice
Feb 19	**Exam 1**
Feb 24	Introduction to Propositional Logic
Feb 26	Truth and Logical Truth
Mar 3	Normal Forms
Mar 5	Predicates and Quantification
Mar 4	Review and Practice
Mar 10	**Exam 2**
Mar 12	Relations – Definitions and Operations
Mar 17	Special Types of Relations
Mar 19	Equivalence Relations
Mar 23-29	***Spring Recess***
Mar 31	Partitions and Equivalence Classes
Apr 2	Partial Ordering
Apr 7	Review and Practice
Apr 9	**Exam 3**
Apr 14	Functions
Apr 16	Operations on functions
Apr 21	Sequences and Subsequences Countable and Uncountable Sets – Power Set
Apr 23	Analysis of Algorithms - Computability and Uncomputability
Apr 28	Graph Theory –Path and Cycles
Apr 30	Connected Graphs – Graphs visit
May 5	Trees – Spanning Trees
May 7	Review and Practice
May 14	6:00pm–8:00pm - Final Exam

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING

STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. **Cheating includes, but is not limited to:**

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;
2. Using notes, textbooks or other information in examinations, tests and quizzes except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;
6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;
9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and
10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes or other academic work.

"Plagiarize" means to take and present as one's own a material portion of the ideas or words of another person or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;
b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and
c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

STUDENT CHEATING AND PLAGIARISM: ACADEMIC SANCTIONS

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. **The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:**

1. Refuse to accept the work for credit; or
2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism takes place; or
3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;
4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

For information regarding the academic appeals procedure, please refer to page 107 of the 2008-2009 FlashGuide

Spring 2009

CS 43901 Section 600
Software Engineering

Department of Computer Science
Kent State University Stark

F 11:00 AM – 1:30 PM Room MH 303

Professor: Dr. Angela Guercio

Office: 424, Main Hall

Phone: 330 244-3424 (KSU ext. 53424)

Office Hours: **TR** 11:30am - 1:30pm 4:35pm -5:15pm **F** 10:15am - 10:55am

other times are available by appointment

Best way to contact me: e-mail to aguercio@kent.edu

Course Information

Class Webpage: http://www.personal.kent.edu/~aguercio/Spring09/CS43901Sp09.html

- all important class information will be posted on the class webpage, readings, assignments, notes, deadlines, cancellations, etc..
- You must **CHECK THE CLASS WEBSITE REGULARLY!!!**

Prerequisites: CS 33001 with ***a grade of C or better.*** This means that a C- in CS10051 is not sufficient to meet the prerequisite.
For more details on course dependences, visit
http://www.cs.kent.edu/programs/ugrad/planner.html

Credit: 3 Credit Hours

Required Text:

G. Booch, R.A. Maksimchuk, M.W. Engle, B.J. Young, J. Conallen, K.A. Houston, ***Object-Oriented Analysis and Design with Application***, Addison-Wesley, 3rd ed., 2007. ISBN13: 978-0-201-89551-3 ISBN10: 0-201-89551-X

The e-book is available on Safari that you can access via the Kent Library/Ohio Link

Emergency: In case of an emergency please contact the security on campus.

Security phone on campus: #53123

Security cell phone (330) 705-0430 or, of course, 911.

I recommend that you program into your cell phone the previous numbers.

Course Description and Objectives

To this course in an introduction to software engineering concepts: life cycle models; modeling languages; requirements analysis; specification; design; testing; validation; project management; and maintenance. The objectives of the course are to develop the ability to master the notation and the process of the object-oriented analysis and design and to experiment the concepts through some realistic applications.

Class Requirements and Expectations

- **Regular class attendance is REQUIRED.**

  - There tends to be a strong correlation between class attendance and grade performance. If you will miss a class**, let me know ahead of time**. In any case, you are responsible for bringing yourself up to date on class material and assignments.

  - Since class participation and regular attendance are part of the final grade**, <u>if you miss more than 3 classes without a documented reason or without making prior arrangements with me, your final grade will be dropped one grade (A to B, B+ to C+ and so on).</u>**

- **Reading ahead is REQUIRED**.

  - The readings are posted online on the class webpage. You must read the material **<u>before</u>** class **<u>and again after</u>** the class.

  - Regular study of the material is REQUIRED.

- **COMPLETE the assigned homework (i.e. Assignments and exercises).**

  - There will be 4 assignments in the course. I will retain the right to change the number of assignments if necessary.

- The class webpage will list the assignments for each week at the beginning of that week so that you can better schedule your work.

- **REVIEW the graded Homework/Assignments.**
  - Homework will be short exercises completed in class or due at the next class period. Review the mistakes after discussion and learn from them.
  - Assignments are larger exercises and will be assigned periodically through the semester. Review the mistakes after grading and discussion and learn from them.
  - Late Homework/Assignments will not be accepted if returned after the solution is given or discussed.
  - If you have difficulties doing your homework or your project please contact me or come to see me. ***Do not procrastinate! Homework and Assignments should be started immediately.***

- **Return work ON TIME**
  - All the homework **must be printed** and e-mailed as an attachment to aguercio@kent.edu. In any case a printed copy MUST be returned to the instructor as well unless stated otherwise.
  - All homework, printed or handwritten are due ***before or at the beginning of class***. Late homework will get **3 points per day penalty** including those returned after the beginning of class.
  - For all e-mailed homework, the instructor will acknowledge the receipt within 24 hours via e-mail. The time of your e-mail will be compared against the deadline of your homework. The reply is your receipt that the work has been turned in (not that the homework is correct!). If you do not receive a receipt, it is YOUR responsibility to contact me to see if the assignment has been lost in transmission. ***Important***: once you submit your files **DO NOT OPEN THEM AGAIN!** If your e-mail didn't reach me or something happened to your files, I may need to ask you to resubmit your files by logging on in my presence to check the modification dates on your files and make sure that they haven't been modified after the due date.

What to expect to find in your Computer Architecture class

- The class should be interactive. In-class exercises are designed to encourage participation. There will be cooperation between you and I, open discussions about problems and possible solutions. You are responsible for taking good notes. Handouts will be given only when necessary.

- ↗ You will present proofs to other students, your will participate in group activities and collaborative learning will be used to discuss possible solutions to problems as well as to provide critical observation to problem solutions. Formal and informal groups will be formed in class to work together. In some cases, you will be required to work on your own. In those cases, I expect an appropriate academic behavior from you. Exchange of information, when forbidden, will not be tolerated.

- ↗ Expect to commit some times to practice problems. You will find that the reading will go very slow and that you need to reread the material several times before you master it.

The <u>Secret Key</u> (not so secret after all!) of how to succeed in a CS class is to:

1. work conscientiously and do all the homework that has been assigned;
2. extrapolate from the examples provided to you, techniques and answers to problems;
3. spend several hours practicing;
4. be alert and participate in class discussions;
5. learn from other people mistakes;
6. Think! Use your logic and be critical of your own work.
7. attend the class and read the material ahead of time;
8. spend time studying the theoretical concepts. Memory helps, but it is practice that reinforces the understanding of the theory;
9. do all the above consistently through the whole semester, be confident about what you are doing and don't be afraid to ask for help;
10. Think, think, think and enjoy!

I am very confident that you can make the above commitment and that you can maintain it during the semester. I am sure that you have all the ability to be successful!

Exams

- There will be 2 Mid-Term Exams which will cover the topics of the previous weeks.

- The 100-points Final Exam is comprehensive.

- All exams are closed books, closed notes.

- Retake exams are not available.

- Make-up exams will only be given in case of serious need (written verification for your inability to take an exam is required) and only when I have been notified ***prior*** to the exam being issued, otherwise you are considered absent for that exam and the grade of your exam is automatically 0.

Grading

Your grade will be based on

1. Your homework
2. Your participation in discussions concerning the homework, class topics, and reading material
3. Your class attendance
4. Your assignments
5. Your exams

Homework	10%
Assignments	35%
Class Participation	5%
Exam 1	15%
Exam 2	15%
Final Exam	20%

Points	Grade
92.5-100	A
89.5-92.4	A-
87-89.5	B+
82.5-86.9	B
79.8-82.4	B-
77-79.8	C+
72-76.9	C
70-72	C-
60-69.9	D
00-59.9	F

Even though I have never found it necessary to change the grading distribution and the number of assignments and exams specified in the syllabus during the semester, I reserve the right to change the method of assigning grades, including changing the number of assignments or exams if I consider it necessary.

Course Withdrawal

If you are considering withdrawing from this course, please inform your instructor and consult a staff member in the Student Services Office, 134 Main Hall. Withdrawal from a course can affect financial aid, student status, or progress within your major. For withdrawal deadlines, please refer to http://www.registrars.kent.edu/home/TermUpdate/sche_adj.htm.

Academic Honesty Policy

When assignments must be individually and independently done, if some students turn in substantially the same solution or program of another student, in my judgment, the solution will be considered a group effort. All involved in the group effort homework will receive a zero grade for that assignment. Policy on academic dishonesty involving programming can be found at http://www.cs.kent.edu/programs/grad/DishonestyPolicy.pdf. A condensed version of the

Administrative Policy And Procedures Regarding Student Cheating And Plagiarism has been added to the last page of this syllabus.

Use of the intellectual property of others without attributing it to them is considered a serious academic offense. Cheating or plagiarism will result in a failing grade for the work or for the entire course. Repeat offenses result in dismissal from the University. University guidelines require that all infractions be reported to the Student Conduct Officer on our campus (see Academic Sanctions below).

Students with Disabilities

Kent State University recognizes its responsibility for creating an institution atmosphere in which students with disabilities can succeed. In accordance with University Policy Subpart E…104.44, if you have a documented disability, you may request accommodations to obtain equal access in this class. Please contact the disability coordinator on campus, Kelly Kulick in Student Accessibility Services, located in the Student Success Center, lower level of the Campus Center, phone (330) 244-5047, or kkulick@kent.edu. After your eligibility for accommodations is determined, you will be given a letter which, when presented to instructors, will help us know best how to assist you.

Classes Canceled – Campus Closings

Announcements of class cancellations and/or campus closings will be made on the campus home page. In the case of an emergency, weather-related or otherwise, please check the web page at http://www.stark.kent.edu for information on the buildings and times of the closing. While information may be broadcast by radio and television, this should be confirmed by the web page, which is the official announcement of the campus and which will be the information used to determine issues related to student attendance, rescheduling of tests, and other concerns.

Conduct

Students and faculty behavior at the Stark Campus is governed by the guidelines set forth in *The Digest of Rules and Regulations*. That document can be found in the University telephone directory. Information can be found at the Office of Judicial Affairs at http://www.kent.edu/administration/emsa/judicial.cfm.

Recycling

KSU Stark Campus recycles. Recycling saves energy, which is currently generated by expensive and vanishing fossil fuels. Recycling one aluminum can saves enough energy to run a TV for three hours! Please take a few seconds to separate your trash. Aluminum cans and plastic and glass bottles may be placed in the blue recycling bins, and all types of paper may be placed in the blue recycling trash cans. All other waste may be placed in the black, brown or gray trash cans.

Important Dates to Remember:

- Last day to withdraw ***before grade W*** is assigned, is Feb 1, 2009
- Last day to drop the class is April 5, 2009
- Exam 1 is Friday, Feb 27
- Exam 2 is Friday, April 10
- Final Exam (comprehensive) is Friday, May 15 (10:30am – 12:30pm)

Spring Recess: March 23 – March 29
Classes End: May 10, 2009

Tentative Outline of the Course

We will cover one chapter per week from the book. Other material and reading sources will be d in class at the appropriate time

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING

STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. **Cheating includes, but is not limited to:**

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;
2. Using notes, textbooks or other information in examinations, tests and quizzes except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;
6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;
9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and
10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes or other academic work.

"Plagiarize" means to take and present as one's own a material portion of the ideas or words of another person or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;
b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and
c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

STUDENT CHEATING AND PLAGIARISM: ACADEMIC SANCTIONS

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. **The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:**

1. Refuse to accept the work for credit; or
2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism takes place; or
3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;

4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

For information regarding the academic appeals procedure, please refer to page 107 of the 2008-2009 FlashGuide

Kent State University
CS 63015/73015: Data Mining Techniques
Fall 2008

Instructor: Ruoming Jin

Mondays 5:30PM-8:15PM; Rm. MSB 276
Office Hours (MSB 264): Mondays 4:00PM-5:00PM

Prerequisites

CS 4/56101 Algorithms
CS 4/53005 Introduction to Database Systems
CS 33001 Data Structures
or Consent of the Instructor

Course Overview

This course teaches the fundamental concepts and techniques of data mining. We will cover a set of interesting topics, including pattern discovery/association rule mining, clustering, classification, information theory, decision theory/Bayesian inference, graphical models, kernel methods/support vector machine, etc.

Each student will be expected to *present two papers* and *do a project* on selected topics. There will be neither homework nor exam. There will be two or three in-class exercise-sessions.

Textbook

P.-N. Tan, M. Steinbach, and V. Kumar, **Introduction to Data Mining,** Addison Wesley, 2005.
Christopher M. Bishop, **Pattern Recognition and Machine Learning**, Springer, 2006

Other references:

[1] Data Mining --- Concepts and techniques, by Han and Kamber, Morgan Kaufmann, 2001. (ISBN:1-55860-489-8)
[2] Principles of Data Mining, by Hand, Mannila, and Smyth, MIT Press, 2001. (ISBN:0-262-08290-X)
[3] The Elements of Statistical Learning --- Data Mining, Inference, and Prediction, by Hastie, Tibshirani, and Friedman, Springer, 2001. (ISBN:0-387-95284-5)
[4] Mining the Web --- Discovering Knowledge from Hypertext Data, by Chakrabarti, Morgan Kaufmann, 2003. (ISBN:1-55860-754-4)

[5] I. H. Witten and E. Frank, Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann, 2nd ed. 2005.

Additional materials will include papers supplied by the instructor

Requirements & Grading Policy

A student's grade will be determined as a weighted average of project (40%), class participation (20%), and presentation (40%).

Accessibility:

University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. **Please note, you must first verify your eligibility for these through Student Accessibility Services** (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

Registration Requirement:

The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such mis-representation. Cheating includes, but is not limited to:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;

2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;

3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;

4. Securing, giving or exchanging information during examinations;

5. Presenting data or other material gathered by another person or group as one's own;

6. Falsifying experimental data or information;

7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;

8. Cooperating with another to do one or more of the above;

9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and

10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work.

“Plagiarize” means to take and present as one’s own a material portion of the ideas or words of another or to present as one’s own an idea or work derived from an existing

source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;

b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and

c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

Academic Sanctions

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:

1. Refuse to accept the work for credit; or

2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took
place; or

3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;

4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The
department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the
vice president for the extended university a recommendation for further sanction under this rule.

Academic Appeals

The general principle that applies to the following procedures is that an appeal is directed to the administrative level immediately above the unit from which the appeal emanates.

Appeals are limited to the following reasons:

a. The decision is arbitrary or unreasonable,
b. The decision resulted from a procedural error,
c. The decision is not in accordance with the facts presented,
d. New information is available which may suggest modification of the decision.

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.

Kent State University
CS 43005/53005: Introduction To Database Systems Design
Spring 2009

Tuesdays and Thursdays 9:15 - 10:30am Rm. MSB 115
Office Hours Tuesdays and Thursdays: 11:00am - 12:15pm

Prerequisites

CS 33001 Data Structures
CS31011 - Discrete Structures
Structured Programming (C++)
Software Engineering topics related to project documentation
Or Consent of the Instructor

Course Overview

This course is an introduction to the design, use, and internal workings of database management systems We consider here systems that are based on relational model - that is, users data is represented as a set of two dimensional tables. During the class we learn the ways to organize the data into relations so that the user applications may concurrently manipulate the data from database quickly and reliably. We briefly discuss the relational model and then concentrate on relational query language SQL. We continue with the study of relational database design. Finally, we study database internal storage organization and concurrency control issues. **Attendance is required for this class**

Textbook

Silberschatz, Korth, Sudarshan
Database System Concepts, 5th Edition, McGrow Hill 2005

Additional Materials

Garcia-Molina, Ullman, Widom
Database Systems The Complete Book, Prentice Hall, 2002

Projects and Exams

There will be three exams during the semester. In addition, there will be a final exam. Final exam is all inclusive. Each student is expected to do a project. List of projects will be available on or before March 1st, 2009. Students must select the project from the list and send an email notification of their selection to jin@cs.kent.edu by March 7th, 2009. Once project is selected, no project change is allowed without the instructor's permission. The project deadline is May 7,

2008 and no late project will be accepted or given any credit.

Requirements & Grading Policy

A student's grade will be determined as a weighted average of exam 1 (15%), exam 2 (20%), exam 3 (15%), final exam (25%), project (20%), and attendance (5%)

University policy 3342-3-18 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Disability Services (contact 330-672-3391 or visit Student Disability Services for more information on registration procedures)

Kent State University
CS 6/79995 St:Graph Mining
Spring 2009

Instructor: Ruoming Jin

Tuesdays and Thursdays 11:00AM-12:15PM; Rm. MSB 276
Office Hours: Mondays and Wednesdays 10:00-11:00AM

Prerequisites

CS 4/56101 Algorithms
CS 6/73105 Data Mining Techniques
Or Consent of the Instructor

Course Overview

The existing database and data mining mainly deal with relational and/or semi-structured data. The recent advance in science and technology has led to a new type of data – graph data – being collected with an unprecedented rate in many fields of human endeavor. Driven by the economic and scientific need, storing, querying, and mining such data are emerging as a new research area.

In the graph model, the vertices represent different entities and the edges capture their pair-wise relationships. For instance, the entire WWW can be treated as a massive direct graph where the nodes represent the web pages. In system biology, various kinds of massive networks, that may easily contain thousands and thousands of genes or gene products, are being constructed for many different species. The graph models provide a unified and powerful representation for these biological networks, including protein interactions, metabolic networks, gene regulatory networks, and gene co-expression networks, among others.

Graph mining and management, referred to as GMM for simplicity, is a highly inter-disciplinary field representing the confluence of several disciplines, including database systems, data warehousing, machine learning, statistics, graph algorithms, and information visualization. This course will provide an in depth discussion of the main topics (including and not limited to scale-free graphs, random graphs, clustering, pattern discovery, graph queries, and etc.) in GMM as well as a wide spectrum of applications such as web mining, social network analysis, sensor/P2P network monitoring, software engineering, and bioinformatics.

This course teaches the fundamental techniques for mining and managing massive graphs (networks). Some basic mathematical tools, such as random graph, spectral methods, and graph

theoretical approaches will also be covered. Each student will be expected to *present a paper* and *lead the discussion* following his/her presentation and *do a project* on selected topics.

There will be nor exam.

Suggested References:

[1] Data Mining --- Concepts and techniques, by Han and Kamber, Morgan Kaufmann, 2001. (ISBN:1-55860-489-8)
[2] Mining the Web --- Discovering Knowledge from Hypertext Data, by Chakrabarti, Morgan Kaufmann, 2003. (ISBN:1-55860-754-4)

[3] Random Graph Dynamics, By Rick Durrett, Cornell U, Cambridge U. Press.

Additional materials will include papers supplied by the instructor.

Requirements & Grading Policy

A student's grade will be determined as a weighted average of homework (30%), presentation (35%), and project (35%).

Acknowledgement

Thanks to Dr. Panayiotis Tsaparas for slides material.

Syllabi for Networks:

Network Architecture & OSI Model
Shannon's Theory & Nyquist Criterion
Encoding, Framing
Error Detection Codes
Error Correction Codes
Flow Control-1
Flow Control-2
Aloha Protocol
Slotted Aloha
CSMA/CD
Ethernet
Wireless Lan 802.11
MIDTERM
Switching & Forwarding
Shortest Path Routing
Distance Vector Protocol
Link State Protocol
Repeaters, Bridges & LAN
IP Addressing
IP Forwarding & Subnetting
IP Fragmentation & Reassebmly
ARP, ICMP
TCP-1
TCP-2
Congestion Control-1
Congestion Control-2

Syllabi for Complex Networks:

Complex Network Theory & Tools
Introduction to Complex Networks; Properties of Complex Networks: Diameter, Degree Distribution, Transitivity, Clustering, Degree Correlation, Phase Transition; Random Graph Models: Uniform Random Graph, Small World Graph, Price's Model; Generating Functions: Configuration Model, one hop neighbors, second neighbors, analysis of various properties with generating functions; Evolution of Networks;

Advanced Characteristics of Complex Networks:
Growing Clustered Scale Free Networks, Evolutionary Games & Complex Networks , Self-Similarity in Networks

Case Analysis: Web & Internet-based Systems
Topology of Usenet
Dynamic Properties of the Internet
Graph Structures of Web
Congestion in Complex Network

Compressing the Massive Web

Cases Analysis: Social Networks & Virtual Communities
Dynamic Centrality in a Social Network
Discovering Local Communities
Evolution in Web Communities
Clustering of Communities
Structural Evolution
Cohesion in Social Groups

Case Analysis: New Applications
Characteristics of Concept Networks
Complex Network for Economic Development Analysis

From: Subject: CS 35101 Computer Architecture Date: Thu, 3 Sep 2009 11:58:28 -0400 MIME-Version: 1.0 Content-Type: text/html; charset="Windows-1252" Content-Transfer-Encoding: quoted-printable Content-Location: http://www.cs.kent.edu/~lucc/35101/sylex.html X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2900.3350

CS 35101 Computer Architecture

Section

Mon and Wed, 11 - 12:15, 121 MSB

Text

Patterson and Hennessy, *Computer organization and design, the=20 hardware/software interface*, 4th edition, Morgan Kaufmann = Publishers.=20

Course Contents

This course is intended primarily as a computer science course in = computer=20 architecture at the advanced undergraduate level. It focuses on = performance and=20 cost analysis, computer architecture, memory systems, input/output = systems,=20 functional units, CPU, pipelining, memory hierarchy, and parallel = processing.=20 This course helps students to learn:=20

- The role of performance measurement and evaluation.=20
- The basics of an instruction set and assembly language programming = using=20 RISC architecture.=20
- The important principles of computer organization and key ideas = such as=20 representation of integers and floating-points numbers, and arithmetic = algorithms,=20
- The key ideas in control mechanisms including pipelining.=20
- The organization and control.=20
- Principles of memory hierarchies and unifying the ideas of caching = and=20 virtual

memory.=20

- The key principles of multiprocessors and multicomputer systems. =

Credits

3

Course Information

- Grader =
- Grade=20
- General=20 Policies=20
- Students=20 with Disabilities

Grader/T.A.

Grade

Homework & Attendance	-	-	15%
1st Exam	9/23	-	20%
2nd Exam	10/19	-	20%
3rd Exam	11/16	-	20%
Final Exam	12/17	10:15 - 12:30	25%

General Policies

In order that work can be graded and returned promptly late = assignments will=20 not be

accepted without an excused absence.=20

Any work that you do for this class is to be your own. Any violation = means=20 that the work will not be accepted and further action will be taken.=20

It is up to the student to make up any missed material. Make-ups will = only be=20 given in the case of an excused absence or a documented, valid = emergency. This=20 includes tests and homework. I encourage you to contact me if an = emergency=20 arises.=20

Students with Disabilities

In accordance with University policy, if you have a documented = disability and=20 require accommodations, please contact the instructor at the beginning = of the=20 semester. Students with disabilities must verify their eligibility = through=20 Accessibility Services (contact 330-6723391 or visit www.kent.edu/sas for more = information on=20 registration procedure).=20

C. C. Lu / lucc@cs.kent.edu =

From: Subject: CS 6/74401 Image Processing Date: Thu, 3 Sep 2009 11:55:42 -0400 MIME-Version: 1.0 Content-Type: multipart/related; type="text/html"; boundary="----=_NextPart_000_000C_01CA2C8D.765E8FC0" X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2900.3350 This is a multi-part message in MIME format. ------=_NextPart_000_000C_01CA2C8D.765E8FC0 Content-Type: text/html; charset="Windows-1252" Content-Transfer-Encoding: quoted-printable Content-Location: http://www.cs.kent.edu/~lucc/64401/sylex.html

CS 6/74401 Image Processing

Section

Mon and Wed, 11 - 12:15, 276 MSB

Text

Gonzalez and Woods, *Digital Image=20 Processing*, 3rd Edition, Prentice Hall, 2008.

Course Contents

- Digital Image Processing Fundamentals=20
- Image Enhancement=20
- Image Transforms=20
- Image Coding=20
- Image Segmentation=20
- Standards for Image Processing

Credits

3

Course Information

- Tests=20
- Programmi= ng=20 assignments=20
- Term=20 Project=20
- General=20 Policies=20
- Cheating = and Plagiarism=20
- Student=20 Accessibility Services

Tests

There will be 3 tests each worth 20% of your overall grade.=20

Literature Review Presentations

There will be 2 literature review presentations each worth 10% of = your=20 overall grade.=20

Term Project

There will be a term project worth 20% of your overall grade.=20

General Policies

In order that work can be graded and returned promptly late = assignments will=20 not be accepted without an excused absence.=20

Any work that you do for this class is to be your own. Any violation = means=20 that the work will not be accepted and further action will be taken.=20

It is up to the student to make up any missed material. Make-ups will = only be=20 given in the case of an excused absence or a documented, valid = emergency. This=20 includes tests and

homework. I encourage you to contact me if an = emergency=20 arises.=20

Cheating and Plagiarism

Cheating = and=20 Plagiarism=20

Student Accessibility Services

Student Accessibility=20 Service=20

C. C. Lu / *mailto:lucc@mcs.kent.edu=20*

------=_NextPart_000_000C_01CA2C8D.765E8FC0 Content-Type: text/css; charset="iso-8859-1" Content-Transfer-Encoding: 7bit Content-Location: http://www.cs.kent.edu/~lucc/64401/master.css BODY { BACKGROUND: white } ADDRESS { FONT-SIZE: medium } P { MARGIN-LEFT: 0.5in } UL { MARGIN-LEFT: 0.5in } DL { MARGIN-LEFT: 0.5in } OL { MARGIN-LEFT: 0.5in } ------ =_NextPart_000_000C_01CA2C8D.765E8FC0--

Syllabus CS 33001
Fall 2008
Computer Science II: Data Structures & Abstraction

Instructor: Dr. Jonathan I. Maletic
Office Hours: MW 10:00am – 11:00am
Office: 218 MSB
Phone: 330.672.9039
Email: jmaletic@cs.kent.edu
URL: www.cs.kent.edu/~jmaletic

Course Time: MW 11:00am-12:15pm
Course Location: MSB 121
Course URL: www.cs.kent.edu/~jmaletic/CS33001/

Course Description: Advanced computer programming design, and development with a primary focus on data structures and abstraction using an object oriented programming language.

Prerequisites:
- CS 23021 Computer Science I: Programming & Problem Solving - *with C (2.0) or better*
- CS 23022 Discrete Structures for Computer Science

Text Book: *Data Structures with C++ using STL*, Ford & Topp, Prentice Hall 2002
Reference Books (recommended): *The C++ Programming Language*, Stroustrup, B., Addison Wesley, 2000 and *Practical Debugging in C++*, Ford & Teorey, Prentice Hall, 2002

Course Objectives:
- Continue developing a disciplined approach to problem solving methods and algorithm development.
- Provide a clear understanding of the concepts of abstract data types.
- To teach a number of the basic algorithms and data structures used in computer science.
- To teach the concepts of object oriented programming.
- To provide a foundation for further studies in Computer Science.
- On completion of this course, students must have a basic understanding of the concepts of abstract data types and object oriented programming methods. Data structures such as lists, stacks, queues, strings, and trees must be understood. The student will have working knowledge of the concepts of classes and objects, operator overloading, constructors, destructors, and generics. The concepts of dynamic data structures and recursion must be well understood. The student will also have an intermediate understanding of sorting, searching, and tree-based algorithms.

Course Organization and Grading:

30% Programming Assignments (4-6 projects)
10% Homework Assignments (4-8 assignments)
10% In class quizzes and exercises (~20)
20% Midterm (Oct. 13th)
30% Final Exam (Dec 10th, 10:15am-12:30pm)

Scale: 100-90 A, 89-80 B, 79-70 C, 69-60 D, 59-0 F, plus/minus used.

Assignments:

There will be at least four substantial programming projects given over the term. The requirement for these assignments will be posted on the course web page. The general topics of the programs will be:

Program 1 – Abstract Data Types & Classes
Program 2 – Dynamic Memory, Containers, Templates
Program 3 – Stacks and/or Queues
Program 4 –Dynamic Data Structures (Complex Pointers)

There will be four to eight homework assignments given. These will typically relate directly to the programming assignments but may relate to a particular issue not addressed in the assignments.

Additionally, there will be a number of in class exercises and quizzes. These will be completed during the class period and handed in at the end of class.

Other Notes:

- Lecture is the student's responsibility, if class is missed; it is in the students best interests to get the notes from a fellow student. The instructor does not have slides or lecture notes to hand out.
- There will be no make up exams.
- Turn off any wireless phones, beepers, or other noise making devices. Turn off volume on lab tops.
- Any modifications to the syllabus will be made on the URL noted below.

University Requirements and Announcements:

- Registration Requirement: The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.
- University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. **Please note, you must first verify your eligibility for these through Student Accessibility Services** (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).
- Copying or plagiarism of any type will not be tolerated and will be dealt with in accordance to Kent State University's policy on cheating and plagiarism described in the student handbook. See the Department's policy on academic Dishonesty: http://www.cs.kent.edu/programs/grad/studentinfo.html#dishonesty and University's policy no cheating and plagiarism: http://www.kent.edu/policyreg/chap3/3-01-8.cfm.

Course Content:

Abstract Data Types (ADTs) and Object Oriented Concepts:

- Definition of ADTs
- Encapsulation and information hiding
- Classes, methods, constructors, and destructors
- Information hiding: Public, private, (and protected)
- Operator overloading and polymorphism
- Generics (templates)
- Inheritance, polymorphism, and virtual functions (dynamic variable binding)

Dynamic Memory Structures:

- Allocation and de-allocation of memory (new, delete)
- Dynamic Arrays
- Pointers, Linked Lists (insertion, deletion, etc.)

Abstract Data Structures & Algorithms:

- Array, multi-dimensional arrays records, files, strings
- Lists, stacks, and queues, sets, bags, vectors
- Infix, prefix, and postfix notations and conversion algorithms
- Binary trees, binary search trees
- Recursion: Design and implementation of recursive functions
- Sorting, Searching
- Hashing and priority queues
- Brief introduction to graphs & associated algorithms

Additional Topics:

- Multi-file programs, make
- Testing and debugging techniques
- Exception handling

Last updated: 08/18/2008

Math 40055/50055 40056/50056 Actuarial Mathematics I & II

Actuarial Mathematics I & II provide a comprehensive introduction to probability models that represent all aspects of life contingencies. These courses, while theoretical in nature, are intended to prepare candidates for successful completion of Society of Actuaries Examination M-LC. They carry prerequisites of probability theory (Math 40011/50011) and the mathematical theory of interest (Math 30055). The focus is on the development and implementation of probability models for a variety of financial variables that arise in the analysis of life insurance.

Recommended Text:

Bowers, N.L., Gerber, H.U., Hickman, J.C., Jones, D.A., Nesbitt, C.J., *Actuarial Mathematics, 2nd ed*., Actex pub., 1997.

Chapter 3,

Chapter 4, Sections 4.1—4.4,

Chapter 5, Sections 5.1—5.4,

Chapter 6, Sections 6.1—6.4, excluding utility theory approach

Chapter 7, Sections 7.1—7.6, excluding utility theory approach

Chapter 8, Sections 8.1—8.4

Chapter 9, Sections 9.1—9.5, 9.6.1, 9.7, 9.9

Chapter 10, Sections 10.1—10.4, 10.5—10.5.1, 10.5.4, 10.6

Chapter 11, Sections 11.1—11.3

Chapter 15, Sections 15.1—15.2.1, 15.4, 15.6—15.6.1

I. Review of Probability Theory

A. Random Variables, Densities, Distributions and Survival Functions

1. Continuous variables

2. Discrete variables

3. Mixed variables

4. Expected Value and Variance

5. Specific distributions

6. Joint distributions, marginal and conditional densities

B. Sums if Independent Random Variables

1. Convolution

2. Moment generating functions

C. Conditional expectation

D. Covariance and correlation

E. Convergence of Random Variables, Inequalities, and Limit Theorems

II. Basic Insurance Models

A. Life Insurance

B. Casualty Insurance—Emphasis on mixed distributions

III. Survival Models

A. Single life—single and multiple decrement models

B. Two lives—both independent and dependent cases

C. Common shock model

D. Interpolation of discrete survival models

1. uniform distribution of deaths
2. constant force of mortality
3. harmonic (Balducci) interpolation

IV. Markov Chain Models

A. Non-homogeneous and homogeneous discrete-time models

B. Poisson Processes

V. Life Insurances and Life Annuities

A. Present value of benefit random variables

B. Present value of loss random variables as a function of premiums

C. Benefit premiums

1. using the Equivalence Principle
2. using percentiles

D. Contract premiums

1. Expense-loaded premiums
2. Expense-loaded benefit reserves
3. Asset shares

E. General Benefit Reserve Theory

F. Actuarial Present Values associated with Markov Processes

ELEMENTARY TOPOLOGY
MATH 46001/56001
Fall 2008

INSTRUCTOR: Austin Melton
OFFICE: 202 MSB
PHONE: (330) 672-5977
E-MAIL: amelton@kent.edu

OFFICE HOURS: Mondays and Fridays 1:00 – 2:00 and by appointment

TEXT: *Topology: Point-Set and Geometric*, by Paul L. Shick

EXAMS: There will be four exams and a final. The exams will be given on Fridays: 12 September, 3 October; 24 October, and 14 November. The final will be given on Friday, 12 December, from 10:15 to 12:30. The exams will be worth 100 points each, and the final will be worth 150 points. Thus, in total the exams and the final will be worth 550 points. All the exams and the final may be comprehensive.

MAKE-UP EXAMS: Make-up exams are given under extraordinary circumstances with written verification of a university-accepted excuse. Please notify me in advance, if possible, if an exam is to be missed.

CLASSROOM SOLUTIONS: Each student will be asked to present solutions to three problems in class. You may work with others to get the solutions. You may even present the solutions as a group. However, if two students present a solution, each student will only get credit for one-half presentation. The presentations will be graded pass/fail. Each presentation will be worth 25 points.

REGISTRATION REQUIREMENT: The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

STUDENTS WITH DISABILITIES: University policy 3342-3-18 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact me at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Disability Services (contact 330-672-3391or visit www.kent.edu/sds for more information on registration procedures).

PLANNED SCHEDULE:

Chapter 1, sections 1 – 3.

Chapter 2, sections 1 – 4.

Chapter 3, sections 1 – 8.

Chapter 4, sections 1 – 5.

Chapter 5, sections 1 – 3.

Chapter 6, sections 1 and 2.

Chapter 8, sections 1 – 6.

Chapter 9, sections 1 – 6.

Chapter 10, sections 1 and 2.

Computer Science I
CS 23021 Sections 003 and 004
Fall 2008, MSB 121, TU 11:00am-12:15pm

Instructor **Dr. Mikhail Nesterenko**

e-mail: mikhail@cs.kent.edu
office: MSB 356
phone: 672-9109
web: http://www.cs.kent.edu/~mikhail/
office hours: 12:30-2:00pm

Lab Instructor **Ms. Debbie Stoffer**

e-mail dstoffer@cs.kent.edu
web: http://www.cs.kent.edu/~dstoffer/
labs: Section 003 – Monday 11:00am-12:55pm,
Section 004 – Wednesday 11:00am-12:55pm

Course Prerequisites

cs10051 “Introduction to Computer Science”

Course Overview

The goal of the course is to familiarize the students with programming in a high-level object-oriented language (C++) while studying the main constructs of C++. The students will learn to translate algorithms into correct programs as well as to debug, document and maintain the code.

The C++ constructs covered include: conditional and loop statements, functions, arrays, pointers, object classes, dynamic memory allocation. Time permitting we will also study standard template library and inheritance.

Textbook

- Walter Savitch, "Problem Solving with C++", 7th edition, 2008, *Addison-Wesley*, ISBN: 0321531345

Class Web Page, Mailing List, Contacting the Instructor

The web page for the class is http://www.cs.kent.edu/~mikhail/classes/ioop/ (I have a link to this page from my homepage). The web page will contain links to the following course materials:

- course syllabus;
- class schedule;
- lecture notes and programming examples used in class;
- programming project assignments;
- additional helpful links.

There is a mailing list set up for the students taking this course. I am going to send announcements and other class related information to this list. It is very important to be on this list to get the latest news and updates about the class. You are required to subscribe to the list within the first two weeks of classes. The subscription instructions are on the course’s webpage. You have to check your mail at least once a day while the school is in session. The simplest way to contact me is via e-mail. **If you**

need to talk to me in person - see me during my office hours or make an appointment via e-mail.

Lectures

Students are expected to attend each lecture. I will not take roll, yet attendance and active participation during a lecture will help you learn the material and succeed in class.

Class Participation

10 points are given for participation. You are expected to answer questions I ask in class. The questions usually deal with the material we covered in the previous class. If you do not attend the class I consider that you do not answer questions I ask you. Rather than participate in class you may select to do a harder last project (which will earn you the extra 10 points.) If you select this option you have to *inform me by e-mail within the first two weeks of classes*. Once you choose this option, you cannot go back to class participation option. Even though I provide this alternative, I encourage you to select class participation since I believe this is the best way to learn the material.

Quizzes

There will be approximately 5 quizzes held during the class. The date of the quiz is announced about a week in advance (there will be no surprise quizzes.) A quiz is held during the first 10 minutes of the class. Late students will not be given extra time to complete the quiz. A quiz usually contains 10 multiple-choice questions. Each question is worth 1 point. I will not count your worst score towards your final grade (missing quiz is equivalent to scoring 0.)

Exams

There will be one exam (held during class) and a final exam (held during finals week). All exams are closed book, closed notes, and must be individual work. It is expected that you take each exam at the scheduled time, unless you make *prior* arrangements with me, or have a *documented* illness (in which case I expect you to contact me as soon as possible). You will be tested on the material I covered in class. The textbook alone may not be sufficient for adequate preparation for the exams.

Programming Projects

There will be approximately 7 programming projects. The programming projects involve reading, modifying and writing C++ code. You will submit your projects electronically. The projects will also be graded electronically. The details on submission will be given to you together with the project assignment. You will be provided with an account on the departmental undergraduate Unix servers. You are, however, free to do your work on any other Unix machine you have access to.

Labs

The lectures are complemented with a lab sessions. The two sections of this course differ in the time of their lab sessions. The lab session is conducted by a teaching assistant. Lab attendance and participation is required. 80 points will be given for lab participation towards your final grade.

Late Policies

- quizzes no late quizzes accepted, no make-up quizzes;
- exams no late exams, no make-up exams;
- projects late projects accepted. 10% of the grade is subtracted for each day the project is late. For penalty calculation Saturday and Sunday are counted as one day.

Late work will be accepted as stated above. I may waive the late policy conditions only in case of a *documented* illness or some extraordinary circumstance. In either case you have to contact me immediately. With respect to projects, my decision to grant you a waiver is partially influenced by the degree of completion of the work assigned. For example, if the project is assigned for 2 weeks, by 10th day I expect you to complete 65-70% of the work.

In general, you will have adequate time to complete each assignment. However, you should begin working on each assignment early so that you will have plenty of time for debugging which may take significantly longer than the initial code writing. Waiting to start coding until the night before the project is due is a bad idea.

Academic Integrity

Student-teacher relationships are built on trust. Students must trust that teachers have made appropriate decisions about the structure and content of the courses they teach, and teachers must trust that the assignments which students turn in are their own. Acts that violate this trust undermine the educational process. Academic dishonesty in any form will be penalized up to assigning grade F in the course.

Cooperation on Homework Assignments and Programming Projects

For both homework assignments and programming projects, I strongly believe that discussion with your peers is an excellent way to learn. If you don't understand something, discussing it with someone who does can be far more productive than beating your head against the wall.

Having advocated discussion, then, I must be clear what is allowed, and what is not. In general, students are allowed to cooperate as follows: you are allowed to discuss with other students the assignment, and general methods for solving the assignment. However, you are not allowed to work with someone else to actually *solve* the assignment, or to *write code* (even pseudocode) for a program, and you are certainly not allowed to *copy* anyone else's solution; doing any of these things will be considered cheating, and will constitute grounds for failing the course.

Note that there is a fine line between discussion and cheating. If you are unsure what is allowed and what isn't, feel free to discuss the distinction with me, but if something feels uncomfortable, it's probably not allowed.

Finally, you should be careful not to give others access to your code. This means that you shouldn't keep your program in a publicly accessible directory, you shouldn't leave your terminal unattended, and you shouldn't forget to pick up your printouts.

University's cheating and plagiarism statement can be found here

http://www.kent.edu/policyreg/chap3/upload/3342.3.01.8.pdf

Grades

Your final course grade will be calculated as follows:

• quizzes (approximately 5)	10 points each, worst score dropped
• class participation	10 points
• programming projects (approximately 7)	20 points for the first
•	50 points for each following project
• midterm exam	100 points
• final exam	100 points
• lab participation	80 points

The sum of the possible scores on all assignments is considered 100% and your final course grade will be determined as follows – 100-93% A, 92-90 A–, 89-87 B+, 86-83 B, 82-80 B–, 79-77 C+, 76-

73 C, 72-70 C–, 69-67 D+, 66-60 D, 59-0 F. **There will be no curve at the end of the course,** although individual exams may occasionally (although rarely) be curved. **Note that this means that your score will not be rounded up: if you get 66.99% you will get a D not a D+**. Thus, you should always be able to determine how well you are doing in the course.

You will provide me with a pseudonym. Your grades will be posted on the course's webpage under your pseudonym.

Students with Disabilities

University policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

Miscellaneous

Try not to be late for class. Make sure you silence your cellphone. The use of laptops while the class is in session is **not** allowed. If you take notes or otherwise cannot avoid using your laptop, notify me within the first two weeks of classes.

Introduction to Computer Science

CS 10051-007/008
Fall 2008
TU 9:15-10:30am in MSB 228

Instructor

instructor:	Mikhail Nesterenko
email:	mikhail@cs.kent.edu
office:	MSB 356
office hours:	TU 12:30-2:00pm

Course Description

A broad introduction to the discipline of computer science. A high level introduction of various aspects of computer science, including algorithm design and analysis, digital logic, architecture and system organization, low-level language programming, operating system concepts, high-level language programming, networking, and social issues.

Prerequisite: MATH 11011 College Algebra, MATH 12001 Algebra & Trigonometry, or two years of high school algebra.

Text

Invitation to Computer Science: C++ Version, 4th Ed., G. Michael Schneider and Judith Gersting, Thomson/Course Technology, 2007, ISBN 978-1-4239-0141-9.

Lab

This course has a lab that meets on Wednesdays or Fridays 9:55-11:50am in room 139 MSB.

Lab attendance is mandatory and your lab grade will count for 80 points towards your overall course grade. You must pass both the lecture and the lab in order to pass the course. Further details about the lab will be given at first lab meeting.

Grading

Midterm Exam 1	100 points
Midterm Exam 2	100 points
Final Exam	100 points
Homework	20 points
Class Participation	10 points
Lab	80 points

Grading Scale: at the end of the course, on the basis of the points that you got, I'll determine the percentage. Using this, I'll assign the overall grade: 100-93% A, 92-90 A–, 89-87 B+, 86-83 B, 82-80 B–, 79-77 C+, 76-73 C, 72-70 C–, 69-67 D+, 66-60 D, 59-0 F.

This course is prerequisite for the next course, CS 23021 Computer Science I. You must earn a grade of C (2.0) or better in this course in order to take CS 23021. Note that a grade of C– (1.7) will not meet this requirement.

Web Page

The course web page will provide links to the course syllabus, lecture schedule, and homework, as well as links to other online resources related to the course material. Students are encouraged to check the web page regularly.

URL `www.cs.kent.edu/~mikhail/classes/intro/`

Registration Requirement

The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFAST) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

Cheating and Plagiarism

The university's plagiarism policy can be found here

`http://www.kent.edu/policyreg/chap3/upload/3342.3.01.8.pdf`

Students with Disabilities

University policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

Introduction to Computer Science

CS 10051-009/010
Spring 2009
TU 12:30-1:45pm in MSB 228

Instructor

instructor:	Mikhail Nesterenko
email:	mikhail@cs.kent.edu
office:	MSB 356
office hours:	TU 11:00am-12:15pm

Course Description

A broad introduction to the discipline of computer science. A high level introduction of various aspects of computer science, including algorithm design and analysis, digital logic, architecture and system organization, low-level language programming, operating system concepts, high-level language programming, networking, and social issues.

Prerequisite: MATH 11011 College Algebra, MATH 12001 Algebra & Trigonometry, or two years of high school algebra.

Text

Invitation to Computer Science: C++ Version, 4th Ed., G. Michael Schneider and Judith Gersting, Thomson/Course Technology, 2007, ISBN 978-1-4239-0141-9.

Lab

This course has a lab that meets on Wednesdays or Fridays 12:05-2:00pm in room 139 MSB.

Lab attendance is mandatory and your lab grade will count for 100 points towards your overall course grade. You must pass both the lecture and the lab in order to pass the course. Further details about the lab will be given at first lab meeting.

Grading

Midterm Exam 1	100 points
Midterm Exam 2	100 points
Final Exam	100 points
Homework	20 points
Class Participation	10 points
Lab	100 points

Grading Scale: at the end of the course, on the basis of the points that you got, I'll determine the percentage. Using this, I'll assign the overall grade: 100-93% A, 92-90 A–, 89-87 B+, 86-83 B, 82-80 B–, 79-77 C+, 76-73 C, 72-70 C–, 69-67 D+, 66-60 D, 59-0 F.

This course is prerequisite for the next course, CS 23021 Computer Science I. You must earn a grade of C (2.0) or better in this course in order to take CS 23021. Note that a grade of C– (1.7) will not meet this requirement.

Web Page

The course web page will provide links to the course syllabus, lecture schedule, and homework, as well as links to other online resources related to the course material. Students are encouraged to check the web page regularly.

URL `www.cs.kent.edu/~mikhail/classes/intro/`

Registration Requirement

University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFAST) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

Cheating and Plagiarism

The university's plagiarism policy can be found here

`http://www.kent.edu/policyreg/chap3/upload/3342.3.01.8.pdf`

Students with Disabilities

University policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

Operating Systems
CS 33211
Spring 2009, TU 2:15-3:30pm , MSB 115

Instructor Dr. Mikhail Nesterenko

e-mail:	mikhail@cs.kent.edu
office:	MSB 356
phone:	(330) 672-9109
office hours:	TU 11:00am-12:15pm
web:	http://www.cs.kent.edu/~mikhail/

Course Prerequisites

CS 33001(Data Structures) and CS 35101 (Computer Architecture) or their equivalents.

Course Overview

The goal of this course is to provide an introduction to operating systems. We will study process management, threads, deadlocks, memory management, file system organization. If time permits we will also examine the networking and distributed aspects of OS design.

Textbook

Operating Systems Concepts, 8th edition, Silberschatz, Galvin and Gagne, John Wiley and Sons, 2008

Class Web Page, Mailing List, Contacting the Instructor

The web page for the class is http://www.cs.kent.edu/~mikhail/classes/os/ (I have a link to this page from my homepage). The web page will contain links to the following course materials:

- course syllabus, class schedule;
- lecture notes;
- homework and programming projects assignments, homework solutions.

A mailing list is set up to for the students taking this course. I am going to send announcements and other class-related information to this list. The instructions how to subscribe to the mailing list as well as the list archive are linked to the course's mailing list. **You have to subscribe to the mailing list within the first two weeks of classes**. You can subscribe to the list from the account of your choice (it does not have to be Kent-State's university or departmental account.) You have to check your mail at least once a day while the school is in session.

The simplest way to contact me is via e-mail. **If you need to talk to me in person – see me during my office hours or make an appointment via e-mail.**

Lectures

Students are expected to attend each lecture. I will not take roll, yet attendance and active participation during a lecture will help you learn the material and succeed in class.

Class Assignments

Homeworks. There will be approximately 3 homework assignments during the semester. The homework assignments will be pencil-and-paper based and will involve solving problems from the textbook.

Quizzes. There will be approximately 3 quizzes. The date of the quiz is announced about a week in advance (there will be no surprise quizzes.) A quiz is held during the first 10 minutes of the class. Late students will not be given extra time to complete the quiz. A quiz usually contains 10 multiple-choice questions.

Programming projects. There will be approximately 3 programming projects. They will involve reading and writing C++ code. If you need quick refresher on C++, check the course's webpage for links to online tutorials.

The projects will be submitted and graded electronically. The details on project submission will be given to you together with project assignment.

Exams. There will be two midterm exams (held during class) and a final exam (held during finals week.) You will be tested on the material I covered in class. The textbook alone may not be sufficient for adequate preparation for the exams.

Late Policies

- quizzes no late quizzes accepted, no make-up quizzes;
- exams no late exams, no make-up exams;
- homeworks no late homeworks;
- projects late projects accepted. 10% of the grade is subtracted for each day the project is late. For penalty calculation Saturday and Sunday are counted as a single day.

Late work will be accepted as stated above. I may waive the late policy conditions only in case of a *documented* illness or another extraordinary circumstance. In either case you have to contact me immediately. With respect to projects and homeworks my decision to grant you a waiver is partially influenced by the degree of completion of the work assigned. For example, if the project is assigned for 2 weeks, by 10^{th} day I expect you to complete 65-70% of the work.

In general, you will have adequate time to complete each assignment. However, you should begin working on each assignment early so that you will have plenty of time for debugging which may take significantly longer than the initial code writing. Waiting to start coding until the night before the project is due is a bad idea.

Academic Integrity

Student-teacher relationships are built on trust. Students must trust that teachers have made appropriate decisions about the structure and content of the courses they teach, and teachers must trust that the assignments which students turn in are their own. Acts that violate this trust undermine the educational process. Academic dishonesty in any form will be penalized up to assigning grade F.

Cooperation on Homework Assignments and Programming Projects

For both homework assignments and programming projects, I strongly believe that discussion with your peers is an excellent way to learn. If you don't understand something, discussing it with someone who does can be far more productive than beating your head against the wall.

Having advocated discussion, then, I must be about clear what is allowed, and what is not. In general, students are allowed to cooperate as follows: you are allowed to discuss with other students the assignment, and general methods for solving the assignment. However, you are not allowed to work with someone else to actually *solve* the assignment, or to *write code* (even pseudocode) for a program, and you are certainly not allowed to *copy* anyone else's solution; doing any of these things will be considered cheating, and will constitute grounds for failing the course.

Note that there is a fine line between discussion and cheating. If you are unsure what is allowed and what isn't, feel free to discuss the distinction with me, but if something feels uncomfortable, it's probably not allowed.

Finally, you should be careful not to give others access to your code. This means that you shouldn't keep your program in a publicly accessible directory, you shouldn't leave your terminal unattended, and you shouldn't forget to pick up your printouts.

The university's plagiarism policy can be found here
http://www.kent.edu/policyreg/chap3/upload/3342.3.01.8.pdf

Grades

Your final course grade will be calculated as follows:

• quizzes (approximately 3) will be dropped)	10 points each (the lowest quiz score
• programming projects (approximately 3)	50 points each
• homeworks (approximately 3)	20 points each
• midterm exams (2)	100 points
• final exam	100 points

The sum of possible scores on all assignments is considered 100% and your final course grade will be determined as follows: 100-93% A, 92-90 A–, 89-87 B+, 86-83 B, 82-80 B–, 79-77 C+, 76-73 C, 72-70 C–, 69-67 D+, 66-60 D, 59-0 F. **There will be no curve at the end of the course,** although individual exams, homeworks, etc. may occasionally (although rarely) be curved. Note that this means that **your score will not be rounded up: if you get 69.99% you will get a D+ not a C-**. Thus you should always be able to determine how well you are doing in the course.

You will provide me with a pseudonym. Your grades will be posted on the course's webpage under your pseudonym.

Students with Disabilities

University policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

Wireless and Mobile Communication Networks
CS 6/75203
Fall 2008

Department of Computer Science

Time and place: MW 12:30 pm - 1:45 pm in Room 276 MSB.
Instructor: Hassan Peyravi
Office hours: MW 2:00 - 3:00 pm, and by appointment.
Office: Room 262 MSB
Contact: e-mail: peyravi@cs.kent.edu, Phone: (330) 672-9062, Fax: (330) 672-0737.

Course Description:

This course examines how wireless systems work and how mobility is supported by the underlying infrastructure. The architecture and the interactions among different functional units are studied in this course.

Prerequisites: Computer Communication Networks.

Texts : Dharma P. Agrawal and O-A Zeng, Introduction to Wireless and Mobile Systems, 2nd Ed., Thomson Publisher, 2006. On reserve at KSU library. TK5103.2 .A42.

References:

Mischa Schwartz, Mobile Wireless Communications, ISBN: 0-521-84347-2, University of Cambridge Press, 2005. On reserve at KSU library. TK5103.2 .S37.

Andrea Goldsmith, Wireless Communications, Cambridge University Press, 2005. On reserve at KSU library. TK5103.2 .G65.

William Stallings, Wireless Communications and Networks. ISBN: 0-13-040864-6, Prentice Hall, 2002.

Charles Perkins, Ad Hoc Networking ISBN: 0-201-30976-9, Addison Wesley, 2002.

David Tse and Pramod Viswanath, Fundamentals of wireless communication, 2005. On reserve at KSU library. TK5103.2 .T74.

Glisic, Savo G, Advanced wireless communications : 4G technologies, On reserve at KSU library. TK5103.2 .G55.

Course Contents:

Introduction and historical perspective of wireless and mobile communications
Mobile Radio Propagation
Channel Coding
Media Access Control (MAC)
Channel Allocation
WiMax
Mesh (multi-hop WiFi/WiMax)

Research topics (QoS, Fairness, Energy, intrusion detection)
Sensor and Ad hoc Networks
Traffic Theories

Evaluation:

Midterm: Wednesday, Oct. 15, 25%
Final: Friday, December 12, 25%
Project: Due Friday, December 5 , 30% (Technical report , 20%, Presentation, 10%)
Homework, 20%

Research Project:

In addition to the requirements for the course, in consultation with the instructor, a student is required to conduct a research project. The topic should be chosen prior to the midterm exam. The due date of the technical report is last day of the classes. The format of the technical report should follow the guidelines given by IEEE Transactions Latex and Microsoft Word Style Files. In class presentations will be scheduled during the last two weeks of the semester.

Students with Disabilities:

Students seeking disabilities services should visit the Office of Student Disability Services (SDS) , or contact (330)672-3391.

Administrative policy regarding student conducts and academic integrity

---

5/08, H. Peyravi
Home

System Modeling and Performance Evaluation
CS 6/75301
Spring 2009

KENT STATE UNIVERSITY

Department of Computer Science

Time and place: MW 12:30 pm - 1:45 pm in Room 276 MSB.
Instructor: Hassan Peyravi
Office hours: MW 1:45 - 2:45 pm, and by appointment.
Office: Room 262 MSB
Contact: e-mail: peyravi@cs.kent.edu, Phone: (330) 672-9062, Fax: (330) 672-0737.

Course Description:

This course covers methods and concepts in system modeling and performance evaluation. Topics include stochastic processes, measurement techniques, monitoring tools, statistical analysis of performance experiments, simulation models, analytic modeling and queuing theory, and workload characterization.

Prerequisites: Graduate Student

Texts : There is no required textbook. I will pass out course notes and supplementary handouts at the end of each class.

References:

Raj Jain. The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling, John Wiley and Sons, Inc., New York, NY, 1991.
Douglas C. Montgomery. Design and Analysis of Experiments, 5th Edition, Wiley Text Books, June 30, 2000. ISBN 0471316490.

Course Contents:

Statistics.
Selection techniques and metrics.
Workload, type of workloads, workload characterization techniques.
Modeling random behavior.
Estimation and testing.

Control charts.
Statistical data analysis, data presentation and visualization.
Capacity planning and benchmarking
Experimental design
Simulation and simulation techniques
Formulation and implementation of simulation models
Queuing theory.

Evaluation:

Midterm: Wednesday, March 11, 25%
Final: Wednesday, May 13, 25%
Project: Due May 6, 30%
Homework, 20%

Research Project:

In addition to the requirements for the course, in consultation with the instructor, a student is required to conduct a research project. The topic should be chosen prior to the midterm exam. The due date of the technical report is last day of the classes. The format of the technical report should follow the guidelines given by IEEE Transactions Latex and Microsoft Word Style Files. In class presentations will be scheduled during the last two weeks of the semester.

Students with Disabilities:

Students seeking disabilities services should visit the Office of Student Disability Services (SDS) , or contact (330)672-3391.

Administrative policy regarding student conducts and academic integrity.

- *Please read **Academic Sanctions** with read to **Administrative Policy and Procedures Regarding Student Cheating and Plagiarism**.*

1/09, H. Peyravi
Home

Computer Science CS 23022 Discrete Structures
Call Number 14180
SYLLABUS
Fall, 2008

Time and Place: Monday, Wednesday, 2:15 - 3:30 in room 121 MCS aka MSB;

Instructor: Michael Rothstein, 268 MSB, phone 330-672-9065. Email addresses: either mrothste@kent.edu or rothstei@cs.kent.edu (both addresses go to the same place, so if you send a message to both places, I will simply get it twice).

Web address: http://www.cs.kent.edu/ rothstei

Office Hours: MW 9:30 to 10:30 and 12:30 to 1:30 pm. Also, you can always send email with questions and/or to set up an appointment. Usual turnaround will be a few hours during the day. Email use is to be preferred over voicemail, which will not be checked as often.

Computer Science Departmental Office: 241 MSB (after you exit the elevator, turn right, walk to the corridor, turn right again, the CS office entrance is at the end of the corridor.)
Phone 330-672-9980

Course Goals: The primary goal of this course is to learn how to prove statements about algorithms. Along the way, we will learn about sets, logic, numbers, and some algorithm design techniques.

Textbook: Rosen Kenneth, *Discrete Mathematics and its Applications*, 6th Ed., McGraw Hill, 2007, ISBN-10: 0-07-288008-2; ISBN-13 978-0-07-288008-3.

Course Description: This course is very different from most courses you have taken in the past; probably the only course similar to it may be Geometry in High School, with its emphasis on proofs. We will emphasize proofs in this course also; in fact, that is the whole reason for this course.

On *even*-numbered Wednesdays, that is September 3 and 17, October 1, 15 and 29, November 12 and December 3, we will hold "exercise sessions" where exercises will be assigned to be done in class. These exercises are open book, open notes, etc., but ALL sources must be acknowledged; in addition, no verbatim copied is allowed; all text must be your own.

Material to be covered: I will try to cover as much as possible from the following sections in the book:

Chapter 1 Sections 1 - 3

Chapter 2 Sections 1 - 4

Chapter 3 Sections 1 - 6 , 8

Chapter 4 Sections 1 - 5

Chapter 5 Sections 1 - 3

Chapter 6 Section 1

Chapter 8 Sections 1 , 3 - 6

Chapter 9 Sections 1 - 5

Chapter 10 Sections 1

Prerequisite: The only prerequisite for this course is CS 10051, Intro to Computer Science.

Attendance policy By initiative of the Provost of the University, I have been charged with keeping full attendance records, at least for the first ten weeks of the semester. Though I will not compute these records into your final averages, when I assign letter grades, I will give you a slightly better grade if you have a better attendance record. Notwithstanding the above, if you are absent, there may be material created, either spontaneously or in response to questions, and covered in the classroom; often there will not be any written notes of this material, so it might be a good idea to team up with somebody who keeps good notes to make sure you have all the material covered. Some of this material may show up in an exam.

Read the text. Only general reading assignments will be given. The class will mostly cover material in the same order as the text book, there may be exceptions however. It is the student's responsibility to maintain an awareness of the material in the text that is currently being covered. Ask the instructor if you are unsure of the text material currently being covered.

The syllabus may be changed during the semester if necesary: changes will be announced in class; they might also show up on the instructor's website.

Some good hints about studying computer science are available at the website:
http://www.educ.kent.edu/community/docwhiz/rdng.html

I recommend you read it and follow its advice: notice what it says about exercises: DO them; at least try; they illustrate a lot of the concepts from the class.

Class disruptions Disruptions should be kept to a minimum; these include (in increasing order of seriousness):

1. Early departure (if announced and done discreetly: please sit near the door so that as few people as possible notice.)
2. Late arrival

3. Use of electronic devices or other devices which may interfere with your or other student's participation. Laptops are acceptable for taking notes, however, please sit in the last row of the room so that your screen does not distract/block other students.
4. Conversation among students.
5. Aiding and/or abetting these or any other student's disruptive behaviors.

Guidelines pertaining to class disruptions are outlined in the University Rules and Regulations, available through:
http://www.kent.edu/policyreg/chap4/4-02-2.cfm

Grading: Your grade will be based mainly on the average grade you obtain in the "exercise sessions" and the final, scheduled for 12:45-3:00 on Friday, Dec 12th. The final will be comprehensive. I will compute two different averages:

1. All Exercise sessions.
2. Dropping the worst exercise session, and giving the remaining 6 a weight of 13% with the final getting the remaining 22%

I will then use the better of the two averages to assign a letter grade as described below.

Test make-up policy: I will need signed documentation to verify *each* individual absence in order to provide make-ups; only university accepted reasons will be honored.

Grading scale: I will assign number grades during the session and only convert them to letter grades when I turn them in at the end of the session. No decision can be made regarding a converion table until the very last minute due to such imponderables as test difficulty, class attendance and participation, etc. which will influence the grade. However, I guarantee the following, worst case, table:

97-100	will convert into an A
94-96	will convert into at least an A-
91-93	will convert into at least a B+
88-90	will convert into at least a B
85-87	will convert into at least a B-
82-84	will convert into at least a C+
79-81	will convert into at least a C
76-78	will convert into at least a C-
73-75	will convert into at least a D+
66-72	will convert into at least a D

Special accommodations for Students with Disabilities: University policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content.

If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Disability Services (contact 330-672-3391 or visit:
http://www.kent.edu/sas for more information on registration procedures).

Registration Requirement : The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

On cheating, plagiarism and other unethical behavior You are encouraged to discuss class problems with other students but required to work independently of anybody else except the instructors and/or tutor, unless otherwise indicated. Copying other people's work, allowing your work to be copied (even inadvertently!) and plagiarizing work will not be tolerated and will be dealt with according to University regulations, as described in the University Policies and Procedures, a condensed version of which is attached.

Notes:

1. By default, the penalty for cheating in this course is an "F" in the course.
2. University regulations require me to notify Student Conduct in case of violations.
3. Cooperation is just as bad as the deed itself: so, deciding which of two is the original is a non-issue: both are equally guilty.

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.(available at http://www.kent.edu/policyreg/chap3/3-01-8.cfm)

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university

and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. Cheating includes, but is not limited to:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;
2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;
6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;
9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and
10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work.

"Plagiarize" means to take and present as one's own a material portion of the ideas or words of another or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;
b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and
c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

Academic Sanctions

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:

1. Refuse to accept the work for credit; or
2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took place; or
3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;
4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

Academic Appeals

The general principle that applies to the following procedures is that an appeal is directed to the administrative level immediately above the unit from which the appeal emanates.

Appeals are limited to the following reasons:

a. The decision is arbitrary or unreasonable,
b. The decision resulted from a procedural error,
c. The decision is not in accordance with the facts presented,
d. New information is available which may suggest modification of the decision.

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8, available at http://www.kent.edu/policyreg/chap3/3-01-8.cfm.

Computer Science CS 4/57205 Information Security
Call Numbers 14096 and 14100
SYLLABUS
Fall, 2008

Time and Place: Monday, Wednesday, 11:00 - 12:15 in room 115 MCS aka MSB;

Instructor: Michael Rothstein, 268 MSB, phone 330-672-9065. Email addresses: either mrothste@kent.edu or rothstei@cs.kent.edu (both addresses go to the same place, so if you send a message to both places, I will simply get it twice).

Web address: http://www.cs.kent.edu/ rothstei

Office Hours: MW 9:30 to 10:30 and 12:30 to 1:30 pm. Also, you can always send email with questions and/or to set up an appointment. Usual turnaround will be a few hours during the day. Email use is to be preferred over voicemail, which will not be checked as often.

Textbook: Pfleeger, C P and Pfleeger, S L, *Security in Computing*, 4th Ed., Prentice Hall, 2007, ISBN 0-13-239077-9

Computer Science Departmental Office: 241 MSB (after you exit the elevator, turn right, walk to the corridor, turn right again, the CS office entrance is at the end of the corridor.)
Phone 330-672-9980

Course Goals: This course is an overview of some of the issues surrounding information security; we will look at basic concepts, threats, protection mechanisms, and at the design and administration of secure systems.

Material to be covered: Wewill cover the material, mostly from the textbook, from chapters 1 to 5, then 7, 8 and 9, we can then continue with the remainig chapters according to time availability and student's wishes.

Prerequisite: Prerequisite for this class is CS 33006, Social and Ethical Issues in Computing; mainly because things will be learned here which are very dangerous: ethical use of these things is a requirement.

Attendance policy By initiative of the Provost of the University, I have been charged with keeping full attendance records, at least for the first ten weeks of the semester. Though I will not compute these records into your final averages, when I assign letter grades, I will give you a slightly better grade if you have a better attendance record. Notwithstanding the above, if you are absent, there may be material created, either spontaneously or in response to questions, and covered in the classroom; often there will not be any written notes of this material, so it might be a good idea to

team up with somebody who keeps good notes to make sure you have all the material covered. Some of this material may show up in an exam.

Read the text. Only general reading assignments will be given. The class will mostly cover material in the same order as the text book, there may be exceptions however. It is the student's responsibility to maintain an awareness of the material in the text that is currently being covered. Ask the instructor if you are unsure of the text material currently being covered.

The syllabus may be changed during the semester if necesary: changes will be announced in class; they might also show up on the instructor's website.

Class disruptions Disruptions should be kept to a minimum; these include (in increasing order of seriousness):

1. Early departure (if announced and done discreetly: please sit near the door so that as few people as possible notice.)
2. Late arrival
3. Use of electronic devices or other devices which may interfere with your or other student's participation. Laptops are acceptable for taking notes, however, please sit in the last row of the room so that your screen does not distract/block other students.
4. Conversation among students.
5. Aiding and/or abetting these or any other student's disruptive behaviors.

Guidelines pertaining to class disruptions are outlined in the University Rules and Regulations, available through:
http://www.kent.edu/policyreg/chap4/4-02-2.cfm

Grading: Your grade will be based on one midterm, one final, assorted exercises assigned as homework and a "class participation grade", based on the number of relevant questions asked: though I don't believe there are any dumb questions, questions like "What day is it?" do not qualify; specially good questions or catching my mistakes get extra points. The weights are:

Class Participation	10%
Midterm (October 15)	30%
Final (December 10 at 10:15)	30%
Homework	30%

All quizzes and exams will be comprehensive. This includes the final.

Test make-up policy: I will need signed documentation to verify *each* individual absence in order to provide make-ups; only university accepted reasons will be honored.

Grading scale: I will assign number grades during the session and only convert them to letter grades when I turn them in at the end of the session. No decision can be made regarding a converion table until the very last minute due to such imponderables as test difficulty, class attendance and participation, etc. which will influence the grade. However, I guarantee the following, worst case, table:

97-100	will convert into an A
94-96	will convert into at least an A-
91-93	will convert into at least a B+
88-90	will convert into at least a B
85-87	will convert into at least a B-
82-84	will convert into at least a C+
79-81	will convert into at least a C
76-78	will convert into at least a C-
73-75	will convert into at least a D+
66-72	will convert into at least a D

Special accommodations for Students with Disabilities: University policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Disability Services (contact 330-672-3391 or visit:
http://www.kent.edu/sas for more information on registration procedures).

Registration Requirement : The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

On cheating, plagiarism and other unethical behavior You are encouraged to discuss class problems with other students but required to work independently of anybody else except the instructors and/or tutor, unless otherwise indicated. Copying other people's work, allowing your work to be copied (even inadvertently!) and plagiarizing work will not be tolerated and will be dealt with according to University regulations, as described in the University Policies and Procedures, a condensed version of which is attached.

Notes:

1. By default, the penalty for cheating in this course is an "F" in the course.
2. University regulations require me to notify Student Conduct in case of violations.
3. Cooperation is just as bad as the deed itself: so, deciding which of two is the original is a non-issue: both are equally guilty.

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.(available at http://www.kent.edu/policyreg/chap3/3-01-8.cfm)

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. Cheating includes, but is not limited to:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;
2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;
6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;

9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and

10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work.

"Plagiarize" means to take and present as one's own a material portion of the ideas or words of another or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;

b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and

c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

Academic Sanctions

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:

1. Refuse to accept the work for credit; or

2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took place; or

3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;

4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

Academic Appeals

The general principle that applies to the following procedures is that an appeal is directed to the administrative level immediately above the unit from which the appeal emanates.

Appeals are limited to the following reasons:

a. The decision is arbitrary or unreasonable,

b. The decision resulted from a procedural error,

c. The decision is not in accordance with the facts presented,

d. New information is available which may suggest modification of the decision.

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8, available at http://www.kent.edu/policyreg/chap3/3-01-8.cfm.

Computer Science CS 33011-001 Speed Programming Techniques
Call Number 14491
SYLLABUS
Fall, 2008

Time and Place: Monday, 5:30 - 8:35, in a lab to be determined.

This course is different from many other courses: to begin with, it only meets once a week, it will also only meet until the first of November, the date of the field trip to the ACM Programming Contest (unless we win the contest; if we win the contest, we can decide what we can do about further training sessions for the finals, to be held in Stockholm, Sweden, April 18-22, 2009. One possibility would be a special elective at the 40000 level for the Spring that would work similarly to this course).

The field trip will take place on October 31st, we will leave in the afternoon, for either Ann Arbor or Youngstown (to be determined, you have a say on this), we will return on either October 1 or October 2, depending on distance and energy level (the contest isn't usually over until 6:30 or so).

Since I am shortchanging you on the contact hours by cutting the course short on November 1, I will hold longer sessions during October, eventually as long as 10:30, to simulate the 5 hours of a real contest. To alleviate hunger pangs, pizza will be made available during these longer class periods.

Instructor: Michael Rothstein, 268 MSB, phone 330-672-9065. Email addresses: either mrothste@kent.edu or rothstei@cs.kent.edu (both addresses go to the same place, so if you send a message to both places, I will simply get it twice).

Web address: http://www.cs.kent.edu/ rothstei

Office Hours: MW 9:30 to 10:30 and 12:30 to 1:30 pm. Also, you can always send email with questions and/or to set up an appointment. Usual turnaround will be a few hours during the day. Email use is to be preferred over voicemail, which will not be checked as often.

Textbook: Skiena, Steven S. and Revilla, Miguel A. *Programming Challenges, The Programming Contest Training Manual* Springer, 2003, ISBN 0-387-00163-8

Computer Science Departmental Office: 241 MSB (after you exit the elevator, turn right, walk to the corridor, turn right again, the CS office entrance is at the end of the corridor.)
Phone 330-672-9980

Course Goals: The goal of this course is to train for the ACM Programming Contest of November 1 and participate in it: participants will get the trip

paid, an Attendance Certificate, a one year student membership in ACM, and, of course, a T-shirt.

Course Description: This course will be part lectures and part practice; initially, there will be a lot of lecturing, and, as the semester goes on, the lecture will diminish and the practice will increase. Hopefully, by the end of October we will have only 10 minutes of lecture per session! The material to be covered in the lectures (hopefully) will include

1. Strategies and general points
2. Review of Data Structures
3. String Manipulation
4. Libraries for sorting and for large number arithmetic.
5. Combinatorics
6. Number theory
7. Backtracking
8. Dynamic Programming
9. Graph Algorithms and graph traversal
10. Computational Geometry

Prerequisite: Though the main prerequisite for this course is CS 33001, Data Structures, we will use material from other courses; if you already have that material, you will be better off.

Attendance policy By initiative of the Provost of the University, I have been charged with keeping full attendance records, at least for the first ten weeks of the semester. As far as this course is concerned, that is the whole course period; attendance and effort will be a major component of the final grade.

This syllabus may be changed during the semester if necesary: changes will be announced in class; they might also show up on the instructor's website.

Class disruptions Disruptions should be kept to a minimum; these include (in increasing order of seriousness):

1. Early departure (if announced and done discreetly: please sit near the door so that as few people as possible notice.)
2. Late arrival
3. Use of electronic devices or other devices which may interfere with your or other student's participation. Laptops are acceptable for taking notes, however, please sit in the last row of the room so that your screen does not distract/block other students.

4. Conversation among students.
5. Aiding and/or abetting these or any other student's disruptive behaviors.

Guidelines pertaining to class disruptions are outlined in the University Rules and Regulations, available through:
http://www.kent.edu/policyreg/chap4/4-02-2.cfm

Grading: Your grade in this course will depend both on effort and on results; since there will be very little homework after the first few weeks, attendance will be an integral part of the grade; so is class participation; if you do not come to class, you will get an "F"; coming to class and doing a bare minimum of work will get you a "D" or a "C" depending on attitude; a sustained effort will either get a "B" or an "A" depending on the degree of success.

Special accommodations for Students with Disabilities: University policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Disability Services (contact 330-672-3391 or visit:
http://www.kent.edu/sas for more information on registration procedures).

Registration Requirement : The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

On cheating, plagiarism and other unethical behavior You are encouraged to discuss class problems with other students but required to work independently of anybody else except the instructors and/or tutor, unless otherwise indicated. Copying other people's work, allowing your work to be copied (even inadvertently!) and plagiarizing work will not be tolerated and will be dealt with according to University regulations, as described in the University Policies and Procedures, a condensed version of which is attached.

Notes:

1. By default, the penalty for cheating in this course is an "F" in the course.

2. University regulations require me to notify Student Conduct in case of violations.
3. Cooperation is just as bad as the deed itself: so, deciding which of two is the original is a non-issue: both are equally guilty.

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.(available at http://www.kent.edu/policyreg/chap3/3-01-8.cfm)

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. Cheating includes, but is not limited to:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;
2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;
6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;
9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and

10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work.

"Plagiarize" means to take and present as one's own a material portion of the ideas or words of another or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;

b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and

c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

Academic Sanctions

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:

1. Refuse to accept the work for credit; or

2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took place; or

3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;

4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

Academic Appeals

The general principle that applies to the following procedures is that an appeal is directed to the administrative level immediately above the unit from which the appeal emanates.

Appeals are limited to the following reasons:

a. The decision is arbitrary or unreasonable,

b. The decision resulted from a procedural error,

c. The decision is not in accordance with the facts presented,

d. New information is available which may suggest modification of the decision.

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8, available at http://www.kent.edu/policyreg/chap3/3-01-8.cfm.

Computer Science CS 6/79995 ST: Advanced Information Security and Cryptology
Call Numbers 21804 and 21805
SYLLABUS
Spring, 2009

Time and Place: Tuesday, Thursday, 2:15 - 3:30 in room 121 MCS aka MSB;

Instructors: Michael Rothstein, 268 MSB, phone 330-672-9065. Email addresses: rothstei@cs.kent.edu . Though the university lists my email address as mrothste@kent.edu, that address will not be checked as frequently, so don't use it if you want a prompt response.
Yaser Doleh, 353 MSB, phone 330-672-3123, 440-582-0047 or
800-IBM-4YOU (ask for Yaser Doleh).
Email address: doleh@cs.kent.edu or yaser@doleh.com

Web address: http://www.cs.kent.edu/ rothstei

Office Hours: Since this is a team-taught course, we have different office hours:

- For Michael Rothstein TR 12:30 to 2:00 pm. Also, you can always send email with questions and/or to set up an appointment. Usual turnaround will be a few hours during the day. Email use is to be preferred over voicemail, which will not be checked as often.
- For Yaser Doleh TR 12:15 - 1:15

Textbook: Bishop, Matt, *Computer Security: Art and Science*, Addison-Wesley, 2003, ISBN 0201-440997

Course Goals: This course will attempt to provide some in-depth coverage of some of the topics in Information Security; we will start with chapters 1-3, but, from then on, each of you will choose a chapter/topic and a recent article on that topic, the requirements (see pages xxxvi and xxxvii of the textbook) together with the topic in question will be covered; you will then be in chrage of the paper chosen. You will also have to email me a written report on the paper.

The syllabus may be changed during the semester if necesary: changes will be announced in class; they might also show up on the instructor's website.

Grading: Your grade will be based on one midterm, one final, and the in-class presentation. In lieu of the final, a second presentation may be given.

The weights are:

Midterm (March 12)	30%
Presentation	35%
Final (May 12) (or second presentation)	35%

The final will be comprehensive.

Grading scale: I will assign number grades during the session and only convert them to letter grades when I turn them in at the end of the session. No decision can be made regarding a converion table until the very last minute due to such imponderables as test difficulty, class attendance and participation, etc. which will influence the grade. However, I guarantee the following, worst case, table:

97-100	will convert into an A
94-96	will convert into at least an A-
91-93	will convert into at least a B+
88-90	will convert into at least a B
85-87	will convert into at least a B-
82-84	will convert into at least a C+
79-81	will convert into at least a C
76-78	will convert into at least a C-
73-75	will convert into at least a D+
66-72	will convert into at least a D

Special accommodations for Students with Disabilities: University policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit:
http://www.kent.edu/sas for more information on registration procedures).

Registration Requirement : The official registration deadline for this course is February 1, 2009. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

On cheating, plagiarism and other unethical behavior You are encouraged to discuss class problems with other students but required to work independently of anybody else except the instructors and/or tutor, unless otherwise indicated. Copying other people's work, allowing your work to be copied (even inadvertently!) and plagiarizing work will not be tolerated and will be dealt with according to University regulations, as described in the University Policies and Procedures, a condensed version of which is attached.

Notes:

1. By default, the penalty for cheating in this course is an "F" in the course.
2. University regulations require me to notify Student Conduct in case of violations.
3. Cooperation is just as bad as the deed itself: so, deciding which of two is the original is a non-issue: both are equally guilty.

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.(available at http://www.kent.edu/policyreg/chap3/3-01-8.cfm)

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. **Cheating includes, but is not limited to**:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;
2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;
6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;

9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and

10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work.

"Plagiarize" means to take and present as one's own a material portion of the ideas or words of another or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;

b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and

c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

Academic Sanctions

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. **The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may**:

1. Refuse to accept the work for credit; or

2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took place; or

3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;

4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

Computer Science CS 4/59995 ST: Secure Coding
Call Numbers 19801 and 19802
SYLLABUS
Spring, 2009

Time and Place: Tuesday, Thursday, 11:00 - 12:15 in room 115 MCS aka MSB;

Instructors: Michael Rothstein, 268 MSB, phone 330-672-9065. Email addresses: rothstei@cs.kent.edu . Though the university lists my email address as mrothste@kent.edu, that address will not be checked as frequently, so don't use it if you want a prompt response.
Yaser Doleh, 353 MSB, phone 330-672-3123, 440-582-0047 or
800-IBM-4YOU (ask for Yaser Doleh).
Email address: doleh@cs.kent.edu or yaser@doleh.com

Web address: http://www.cs.kent.edu/ rothstei

Office Hours: Since this is a team-taught course, we have different office hours:

- For Michael Rothstein TR 12:30 to 2:00 pm. Also, you can always send email with questions and/or to set up an appointment. Usual turnaround will be a few hours during the day. Email use is to be preferred over voicemail, which will not be checked as often.
- For Yaser Doleh TR 12:15 - 1:15

Textbook: Chess, B and West, J, *Secure Programming with Static Analysis*, Addison-Wesley, 2007, ISBN-10: 0321424778, ISBN-13: 978-0321424778.

Material to be covered: The goal of this course is to learn how we can avoid the pitfalls of insecure programming and how to check for them through static analysis. That was the goal behind the choice of textbook, even though it is somewhat of an "ad book". The goals trascend the package described in the book. Having said that, we will pretty much follow the book verbatim.

Attendance policy By initiative of the Provost of the University, I have been charged with keeping full attendance records, at least for the first ten weeks of the semester. Though I will not compute these records into your final averages, when I assign letter grades, I will give you a slightly better grade if you have a better attendance record. Notwithstanding the above, if you are absent, there may be material created, either spontaneously or in response to questions, and covered in the classroom; often there will not be any written notes of this material, so it might be a good idea to team up with somebody who keeps good notes to make sure you have all the material covered. Some of this material may show up in an exam.

Read the text. Only general reading assignments will be given. The class will mostly cover material in the same order as the text book, there

may be exceptions however. It is the student's responsibility to maintain an awareness of the material in the text that is currently being covered. Ask the instructor if you are unsure of the text material currently being covered.

The syllabus may be changed during the semester if necesary: changes will be announced in class; they might also show up on the instructor's website.

Class disruptions Disruptions should be kept to a minimum; these include (in increasing order of seriousness):

1. Early departure (if announced and done discreetly: please sit near the door so that as few people as possible notice.)
2. Late arrival
3. Use of electronic devices or other devices which may interfere with your or other student's participation. Laptops are acceptable for taking notes, however, please sit in the last row of the room so that your screen does not distract/block other students.
4. Conversation among students.
5. Aiding and/or abetting these or any other student's disruptive behaviors.

Guidelines pertaining to class disruptions are outlined in the University Rules and Regulations, available through:
http://www.kent.edu/policyreg/chap4/4-02-2.cfm

Grading: Your grade will be based on one midterm, one final, one programming assignment, the result of a static analysis of your code on the programming assignment, and a "class participation grade", based on the number of relevant questions and comments: specially good questions or catching my mistakes get extra points. The weights are:

Class Participation	10%
Midterm (March 19)	25%
Final (May 14 at 12:45)	25%
Programming Assignment, incl. Analysis	40%

The final will be comprehensive.

Test make-up policy: I will need signed documentation to verify *each* individual absence in order to provide make-ups; only university accepted reasons will be honored.

Grading scale: I will assign number grades during the session and only convert them to letter grades when I turn them in at the end of the session. No decision can be made regarding a converion table until the very last minute due to such imponderables as test difficulty, class attendance and

participation, etc. which will influence the grade. However, I guarantee the following, worst case, table:

97-100	will convert into an A
94-96	will convert into at least an A-
91-93	will convert into at least a B+
88-90	will convert into at least a B
85-87	will convert into at least a B-
82-84	will convert into at least a C+
79-81	will convert into at least a C
76-78	will convert into at least a C-
73-75	will convert into at least a D+
66-72	will convert into at least a D

Special accommodations for Students with Disabilities: University policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit:
http://www.kent.edu/sas for more information on registration procedures).

Registration Requirement : The official registration deadline for this course is February 1, 2009. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

On cheating, plagiarism and other unethical behavior You are encouraged to discuss class problems with other students but required to work independently of anybody else except the instructors and/or tutor, unless otherwise indicated. Copying other people's work, allowing your work to be copied (even inadvertently!) and plagiarizing work will not be tolerated and will be dealt with according to University regulations, as described in the University Policies and Procedures, a condensed version of which is attached.

Notes:

1. By default, the penalty for cheating in this course is an "F" in the course.
2. University regulations require me to notify Student Conduct in case of violations.
3. Cooperation is just as bad as the deed itself: so, deciding which of two is the original is a non-issue: both are equally guilty.

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.(available at http://www.kent.edu/policyreg/chap3/3-01-8.cfm)

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. **Cheating includes, but is not limited to**:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;
2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;
6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;

9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and

10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work.

"Plagiarize" means to take and present as one's own a material portion of the ideas or words of another or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;

b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and

c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

Academic Sanctions

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. **The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may**:

1. Refuse to accept the work for credit; or

2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took place; or

3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;

4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

Computer Graphics Syllabus

Course: CS 4/57101	Fall 2008
Call Number:14109/14111	
Time: 5:30-6:45pm tt	Location: 115 MSB
Arden Ruttan	Office : 270
Phone : (330) 672-9066	Mail address : ruttan@cs.kent.edu
Office Hours : 4:30 - 5:30pm M-H and by appointment	
Grader: Debbie Stoffer	Mail address: dstoffer@cs.kent.edu
Office : 115MSB	Office Hours: to be announced

Goal:

The goal of this course is to provide an introduction to the theory and practice of computer graphics.

Prerequisites:

The course will assume a good background in programming in C or C++ and a background in mathematics including familiarity with the theory and use of coordinate geometry and of linear algebra such as matrix multiplication.

Course Content:

- Class Text: Interactive Computer Graphics: A Top-Down Approach using OpenGL, 5/E, Edward Angel, ISBN: 0-321-53586-3, Addison-Wesley,2008.
- Other References: Introduction to Computer Graphics, James D. Foley, Andries van Dam, Steven K. Feiner, John F. Hughes, Richard L. Phillips, ISBN: 0-201-60921-5, Addison Wesley, 1994.

- OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 1.4, 4/E, OpenGL Architecture Review Board,Dave Shreiner,Mason Woo Jackie Neider, Tom Davis,ISBN: 0-321-17348-1,Addison Wesley,2004.

Course Material (Chapers 1-10,13 of the class text):

1. Graphics Systems and Models.
2. Graphics Programming : Getting started with OpenGL.
3. Input and Interaction in OpenGL.
4. Geometrical Objects and Transformations in 2D and 3D: Objects representation, Coordinates transformation, windows and viewports.
5. Viewing in 3D: Orthogonal and projective views, hidden surface removal.
6. Light, Shading and Materials: Illumination and shadows, light sources, surfaces.
7. From Vertices to Fragments: Graphics pipeline, rasterization, color system.
8. Texture Mapping.

   and possibly one or more from the following in less detail:

9. Programmable shaders: Shading language, vertex and fragment shaders.
10. Modeling Techniques: Graphics data structure, modeling methods.
12. Advanced Rendering Techniques: Ray tracing, radiosity, image based rendering.

Grading:

Grades will be assigned according to the following scale:

- A 90-100.
- B 80-89.
- C 70-79.
- D 60-69.
- F 0-59.

Grade Calculations:

- Midterm 20% .
- Final 20%.
- Class Assignments 60%.
- Note:Assignments will primarily be programming assignments requiring implementation of applications employing the theory covered in the lectures and the books. There will however be some theoretical homeworks and questions as well. Students are reminded that completion of both theory and programming parts of the homework are necessary to achieve a good grade.

Cheating and Plagarism

Please note the University Policy on Cheating and Plagarism

Students with Disabilities:

University policy 3342-3-18 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Disability Services (contact 330-672-3391 or visit www.kent.edu/sds for more information on registration procedures).

CS 49901 11952

This course is an integrative experience that brings together all components of the undergraduate computer science curriculum in an applied, hands on real world setting. The course is three credits lecture and one credit lab. Prerequisite: CS 45201, 43901, 33006,43005.

In particular, we will work in project teams to design and implement middleware for cluster computing

Course Requirements

- Working in a group (3-4 members) design, implement, and demonstrate a computer cluster middleware project for computational steering. The Wikipedia definition for middleware is *" computer software that connects software components or applications. The software consists of a set of enabling services that allow multiple processes running on one or more machines to interact across a network."* The software components that we will be working with are OpenMPI and Chromium. A computational steering environment is a set of tools to enable a user to interact with and change the execution behavior of a program while it is running.
- Project teams will produce an initial report describing their project, its goals and a time line for its implementation. Over the course of the semester that report will be update with completed software and documentation. Reports are submitted as wiki pages.The class wiki is capstone.cs.kent.edu/mediawiki.
- Approximately every two weeks each team will present a progress report to the classes discussing what they have accomplished and discussing any revisions in project time lines and goals. The project reports will account for 85% of the class grade. They will be posted on the wiki and given verbally in class. The grade for a project report will be determined by quality of the implementation and the clarity of the report. Each member of the team will be expected to present part of the report.

- Class and Lab Participation (lab interactions, lab productivity, class discussions, finding solutions to posted problems, etc): 15% of the grade.

INSTRUCTOR

- Instructor: Arden Ruttan, 270 MSB. Phone 29066. User ID. ruttan. It is best to send me email when you wish to contact me.
- Office Hours: 6:45-7:30 T, 3:30-4:40 H and by **appointment.**

Lab Instructor

- Andrew Sutton
- www.cs.kent.edu/~asutton

Class Notes

Time: 2:15-3:30 TH MSB 276.

Academic Honesty

Please note the University Policy on Cheating and Plagarism and the potential penalties associated with those activities

Students with Disabilities

University policy 3342-3-18 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Disability Services (contact 330-672-3391 or visit www.kent.edu/sds for more information on registration procedures).

CS 43203 11984

Window and Unix operating system APIs: manipulating system processes, system io, system permissions, files, directories, signals, threads, sockets, terminal, etc. Prerequisite: CS 33211. Students are expected to have a basic knowledge of the C programming language, in addition, an intermediate knowledge of the Unix Operating System is assumed.

Textbook

Understanding UNIX/LINUX Programming: A Guide to Theory and Practice, Bruce Molay

Supplemental Texts include:

- Windows System Programming, 3rd Edition, Johnson M. Hart . This is the class reference for Win32/Win64 programming.Available on Safari
- Advanced Programming in the Unix Environment, W. Richard Stevens, Stephen A. Rago This is the Standard Reference for Unix System Programming
- A good shell reference book for your favorite shell.
- A good UNIX book (if you need one). O'Reilly nutshell books are good.

Grading

Grades will be assigned according to the following scale,

- A 90-100
- B 80-89
- C 70-79
- D 60-69
- F 0-59

Grade Calculation: Grades will be calculated based on the following

- Midterm 15%
- Final Exam 15%

- Class Assignments 70%

INSTRUCTOR

- Instructor: Arden Ruttan, 270 MSB. Phone 29066. User ID. ruttan. It is best to send me email when you wish to contact me.
- Office Hours: 6:45-7:30 T, 3:30-4:30 H and by **appointment.**

Class Notes

Time: 5:30-6:45 TH MSB 115.

Academic Honesty

Please note the University Policy on Cheating and Plagarism and the potential penalties associated with those activities

Students with Disabilities

University policy 3342-3-18 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Disability Services (contact 330-672-3391 or visit www.kent.edu/sds for more information on registration procedures).

CS 10001 Computer Literacy, Fall 2008

Section 004	TR 12:30 -1:45 pm, Room 221, Math & Computing Sciences Bldg
Instructor	L. Gwenn Volkert
Office	216 Mathematics and Computer Science Building
E-mail	volkert@cs.kent.edu
Webpage	http://www.cs.kent.edu/~volkert/cs23021/F08/
Office hours	Monday and Wednesday 10:30 am - 12:00 pm, other times may be scheduled by appointment. Many questions can easily be handled through email.

Course Description

Introduction to the history, language, procedures, applications, abuses, and impact of computers. Hands-on experience with microcomputers.

This course is to help you become more computer literate. To do this we will cover concepts and principles of computers of all types, and provide hands-on experience in using some basic software packages. The emphasis will be placed on personal computers and their uses; however, we will still cover concepts of larger computers.

Text

Alan Evans, Kendall Martin, and Mary Anne Poatsy, ***Technology in Action, Fifth Edition,*** 2009, Prentice Hall. ISBN-10: 0-13-513720-9, ISBN-13: 978-0-13-513720-8.

Materials and equipment

A PC is required to complete the lab assignments. You may use your own personal computer or a computer on campus. There will be an opportunity to carry out lab exercises during class time in one of the departmental computer labs. Any exercises not completed during this time can be finished on your own. You may want to save your work (when necessary) on a key drive or send your work files to your e-mail account for later access.

Prerequisites There are no prerequisites for this course.

Credits 3

Homework

Homework will consist of hands-on computer exercises. Your hands-on exercises are found at the end of each chapter and some will be given to you in the lab; these exercises can be finished in the lab or at home. Lab exercises will constitute 20% of your cumulative grade.

Class Participation

This course will not be conducted as a lecture course but as a discussion course. It will be expected that everyone will participate in discussions. Attendance will count 15% toward your semester grade. Documentation must be provided for absences. Discussion will count 10% toward your semester grade.

Exams

There will be three exams – two midterms and one final. Each midterm will be worth 15% of your cumulative grade, and the final will be worth 25% of your cumulative grade. The first midterm will be on September 25 and will cover Chapters 1 – 5, the second midterm will be on October 30 and will cover Chapters 6 – 10, and the final will cover all the chapters, but the predominant number of questions will come from Chapters 11 – 13. Your final will be on Tuesday, December 9, 2008 from 12:45 p.m. – 3:00 p.m.

All tests include material covered in the classroom, homework, and labs. Questions can be asked in all forms: true/false, multiple-choice, fill in the blank, and essay.

Policies Regarding Evaluation of Your Work

In order that work can be graded and returned promptly, late assignments will not be accepted without a valid excuse.

Any work that you do for this class is to be your own. Any violation means that the work will not be accepted and further action will be taken. Plagiarism will not be tolerated.

Further to this, I am incorporating the university's policy on plagiarism here:

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. Cheating includes, but is not limited to:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;

2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;

3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;

4. Securing, giving or exchanging information during examinations;

5. Presenting data or other material gathered by another person or group as one's own;

6. Falsifying experimental data or information;

7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;

8. Cooperating with another to do one or more of the above;

9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and

10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work.

"Plagiarize" means to take and present as one's own a material portion of the ideas or words of another or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;

b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and

c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

Academic Sanctions

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:

1. Refuse to accept the work for credit; or

2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took place; or

3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;

4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to

forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

Academic Appeals
The general principle that applies to the following procedures is that an appeal is directed to the administrative level immediately above the unit from which the appeal emanates.

Appeals are limited to the following reasons:

a. The decision is arbitrary or unreasonable,
b. The decision resulted from a procedural error,
c. The decision is not in accordance with the facts presented,
d. New information is available which may suggest modification of the decision.

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.

It is up to the student to make up any missed material. Make-ups will only be given in the case of an excused absence or a documented, valid emergency. This includes tests and homework. I encourage you to contact me if an emergency arises.

If you have any problems with the class please talk to me first. If you do not feel comfortable talking to me, or if you are unsatisfied after having talked to me, then you can speak with the Course Coordinator, Walter Pechenuk / wpechenu@cs.kent.edu. His office is 272 MSB.

Grading Scale

A 93% <= score <= 100%
A- 90% <= score < 93%
B+ 87% <= score < 90%
B 83% <= score < 87%
B- 80% <= score < 83%
C+ 77% <= score < 80%
C 73% <= score < 77%
C- 70% <= score < 73%
D+ 67% <= score < 70%
D 60% <= score < 67%
F 0% <= score < 59%

Registration Requirement

The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

Respectful Student Conduct

This class follows Kent State University's regulations regarding behavior in the classroom. It is expected that each student will be respectful to the instructor as well as to fellow classmates. Students should behave maturely and professionally.

Use of profanity, rudeness towards fellow students or the instructor, angry outbursts, refusal to participate in classroom activities, repeated tardiness, ringing of cell phones and leaving the classroom prior to class dismissal without the approval of the instructor are just some examples of disruptive behavior. The instructor will ask disruptive student to cease and desist and will inform the student of possible suspension and/or dismissal from class.

Guidelines pertaining to class disruptions are outlined in the Digest of Rules and Regulations to be found in the Kent State University Telephone Directory.

Students with Disabilities

University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through the Office of Student Accessibility Services on the Ground Floor of the DeWeese Center (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

If you wish to take your examination at the SAS Center, you must notify your instructor one week in advance. You must begin your examination at the same time that your classmates begin it.

CS 1 - Programming and Problem Solving
CS 23021 (Fall 2008) - Sections 001 & 002

Tuesday & Thursday 9:15 am - 10:30 pm
Room 115 MSB, Call No. 14069 & 14070

Instructor: **Dr. L. Gwenn Volkert**

Email : volkert@cs.kent.edu
Office: 216 MSB
Phone: (330) 672-9037
Mailbox: 239 MSB
Web: http://www.cs.kent.edu/~volkert
Office Hrs: M/W 10:30 pm - 12:00 pm, and by appointment
Class Web: http://www.cs.kent.edu/~volkert/cs23021/F08

Course Description:

This is a programming course that also introduces the object oriented programming paradigm. Object Oriented Programming is the most important and dominant programming approach today. Object Oriented Programming is quite different than functional or procedural programming, and it is difficult to learn on your own. This course is concerned primarily with beginning programmers who have never programmed in the C++ language. The course will focus on programming correctly in C++ by teaching structured and object oriented programming techniques, and proper program design. Students will design, develop, write (translate), compile, execute and debug C++ programs throughout the course. Hands-on programming will be a key part of the course. Please note that this course, CS 23021, is a prerequisite for CS 33001, Data Structures. A grade of C (2.0 honor points) or better in CS 23021 is required to take CS 33001. Note that a grade of C- (1.7 honor points) will NOT meet this requirement. Please see http://www.cs.kent.edu/programs/ugrad/planner.html for additional details.

Course Objectives:

- Introduce a disciplined approach to problem solving and algorithm development.
- Program development, including design, coding (translating), debugging, and testing.
- Basic language statements - syntax, semantics, usage of the C++ language
- Functions - syntax, semantics, usage, functional abstraction
- Familiarity with strings and vectors
- Types - syntax, semantics, usage
- Structures
- Introduce the basics of classes (limited)
- Pointers and dynamic memory for single objects (if time permits)
- Correct translation of algorithm into program
- Basic tool usage - editor (emacs or vi), compiler (g++), shell (bash), search tools (grep)

Prerequisites

CS10051 Introduction to Computer Science or CS10061 Intro to Computer programming or permission from the instructor.

Textbook

- Problem Solving With C++, The Object of Programming, 6th edition, Walter Savitch, Addison Wesley, ISBN: 0-321-41269-9, 2006.

Lectures

Students are expected to attend each lecture. I will not take roll, yet attendance and active participation during a lecture will help you learn the material and succeed in class. Please note your attendance will be noticed as I often direct questions to individual students through out the lecture.

Class Participation

20 points are given for participation. You are expected to answer questions I ask in class. The questions usually deal with the material we covered in the previous class. If you do not attend the class I consider that you do not answer questions I ask you. Rather than participate in class you may select to do a harder last project (which will earn you the extra 20 points.) If you select this option you have to *inform me by e-mail within the first two weeks of classes*. Once you choose this option, you cannot go back to class participation option. Even though I provide this alternative, I encourage you to select class participation since I believe this is the best way to learn the material.

Quizzes

There will be approximately 7 quizzes held during the class. The date of the quiz is announced about a week in advance (there will be no surprise quizzes.) A quiz is held during the first 10 minutes of the class. Late students will not be given extra time to complete the quiz. A quiz usually contains 10 multiple-choice questions. Each question is worth 1 point. I will not count your worst score towards your final grade (missing quiz is equivalent to scoring 0.)

Exams

There will be one midterm exam (held during class) and a final exam (held during finals week). All exams are closed book, closed notes, and must be individual work. It is expected that you take each exam at the scheduled time, unless you make *prior* arrangements with me, or have a *documented* illness (in which case I expect you to contact me as soon as possible). You will be tested on the material I covered in class. The textbook alone may not be sufficient for adequate preparation for the exams.

Labs

The labs are designed to augment the lecture material with practical experience using command line development tools and to reinforce programming concepts. Attending and completing the accompanying programming lab is required. **You will not receive credit for any labs that you miss**. A university-approved excuse will allow you to make up a lab.

Programming Projects

There will be approximately 6 programming projects. The programming projects involve reading, modifying and writing C++ code. You will submit your projects electronically. The projects will also be graded electronically. Details on individual programming assignments and the detailed requirements for them will be given when they are assigned. The general program requirements listed on the course website apply to all programs.

You will be provided with an account on departmental undergraduate Unix server. You are, however, free to do your work on any other Unix machine you have access to.

Late Policies

- quizzes — no late quizzes accepted, no make-up quizzes;
- exams — no late exams, no make-up exams;
- projects — late projects accepted. 10% of the grade is subtracted for each day the project is late. For penalty calculation Saturday and Sunday are counted as one day.

Late work will be accepted as stated above. I may waive the late policy conditions only in case of a *documented* illness or some extraordinary circumstance. In either case you have to contact me immediately. With respect to projects, my decision to grant you a waiver is partially influenced by the degree of completion of the work assigned. For example, if the project is assigned for 2 weeks, by 10^{th} day I expect you to complete 65-70% of the work.

In general, you will have adequate time to complete each assignment. However, you should begin working on each assignment early so that you will have plenty of time for debugging which may take significantly longer than the initial code writing. Waiting to start coding until the night before the due date for a project is a bad idea.

Academic Integrity

Student-teacher relationships are built on trust. Students must trust that teachers have made appropriate decisions about the structure and content of the courses they teach, and teachers must trust that the assignments that students turn in are their own. Acts that violate this trust undermine the educational process. Academic dishonesty in any form will be penalized up to assigning grade F in the course.

Cooperation on Homework Assignments and Programming Projects

For both homework assignments and programming projects, I strongly believe that discussion with your peers is an excellent way to learn. If you don't understand something, discussing it with someone who does can be far more productive than beating your head against the wall.

Having advocated discussion, then, I must be clear what is allowed, and what is not. In general, students are allowed to cooperate as follows: you are allowed to discuss with other students the assignment, and general methods for solving the assignment. However, you are not allowed to work with someone else to actually *solve* the assignment, or to *write code* (even pseudo-code) for a program, and you are certainly not allowed to *copy* anyone else's solution; doing any of these things will be considered cheating, and will constitute grounds for failing the course. Note that there is a fine line between discussion and cheating. If you are unsure what is allowed and what isn't, feel free to discuss the distinction with me, but if something feels uncomfortable, it's probably not allowed.

Finally, you should be careful not to give others access to your code. This means that you shouldn't keep your program in a publicly accessible directory, you shouldn't leave your terminal unattended, and you shouldn't forget to pick up your printouts.

Grading

Your final course grade will be calculated as follows:

• quizzes (approximately 6)	10 points each, worst score dropped
• class participation	25 points
• programming projects (approximately 5)	20 points for the first project
	50 points for each subsequent project
• midterm exam	100 points
• final exam	100 points
• lab points	140 points

TOTAL POINTS:	630 points

The sum of the possible scores on all assignments is considered 100% and your final course grade will be determined as follows – A = 93–100%, B = 83–86.99%, etc. **There will be no curve at the end of the course,** although individual exams may occasionally (although rarely) be curved. Note that this means that **your score will not be rounded up: if you get 69.99% you will get a D+ not a C-**. Thus you should always be able to determine how well you are doing in the course.

You will provide me with a pseudonym by the end of the second week of class. Your grades will be posted on the course's webpage under your pseudonym in a directory that is only accessible with a proper userid and password.

Letter grades will be assigned according to the following percentages. Plus/Minus grades will be given.

93% <= A <=100%
90% <= A- < 93%
87% <= B+ < 90%
83% <= B < 87%
80% <= B- < 83%
77% <= C+ < 80%
73% <= C < 77%
70% <= C- < 73%
67% <= D+ < 70%
60% <= D < 67%
0% <= F < 59%

Students with Disabilities

University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through the Office of Student Accessibility Services on the Ground Floor of the DeWeese Center (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

If you wish to take your examination at the SAS Center, you must notify your instructor one week in advance. You must begin your examination at the same time that your classmates begin it.

Student Expectations

- This is a computer-programming course. Computer-programming is best learned by actually writing lots of computer programs. You will need to spend a lot of time designing, writing and debugging programs.
- Start work on a programming assignment as soon as possible after it is given. This will allow you to discover things that aren't clear to you and ask questions about them. It is hard to write a program quickly at the last minute, it is highly recommended to avoid this situation. Also, unforeseen circumstances often occur.
- Attendance is necessary and expected. It is up to the student to make up any missed material. If a class is missed it is best to get notes from a fellow student (who has taken good notes). This will be the best record of what transpired during the class meeting.
- You have a printer quota of 50 pages. Additional pages may be purchased at 3 cents per page. At least 24 hours must be allowed for the quota to be increased.
- Please turn off any phones, beepers, or other noise-making device before class begins.
- The schedule and procedures for this course are subject to change. Changes will be announced in class and posted on the course website, it is the student's responsibility to learn and adjust to changes.
- If you have any problems, including understanding the material that we cover in class or using the computer, please email me and/or bring your questions/problems to office hours.

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. Cheating includes, but is not limited to:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;

2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;

3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;

4. Securing, giving or exchanging information during examinations;

5. Presenting data or other material gathered by another person or group as one's own;

6. Falsifying experimental data or information;

7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;

8. Cooperating with another to do one or more of the above;

9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and

10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work.

“Plagiarize” means to take and present as one’s own a material portion of the ideas or words of another or to present as one’s own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;

b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and

c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

Academic Sanctions

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:

1. Refuse to accept the work for credit; or

2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took place; or

3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;

4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

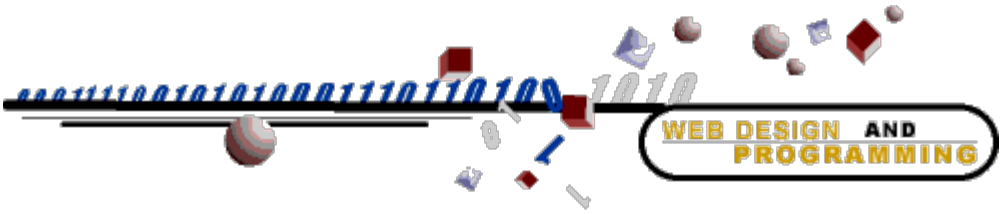
Academic Appeals

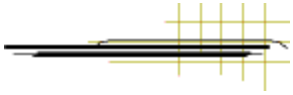
The general principle that applies to the following procedures is that an appeal is directed to the administrative level immediately above the unit from which the appeal emanates.

Appeals are limited to the following reasons:

a. The decision is arbitrary or unreasonable,
b. The decision resulted from a procedural error,
c. The decision is not in accordance with the facts presented,
d. New information is available which may suggest modification of the decision.

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.





VCD 46000 and CS 4/57105 Web Design and Programming I

The lab-oriented course is organized and taught jointly by CS and Arts faculty and meets twice a week. Students form teams to design and implement actual Web pages and present their final projects at the end of the semester.

During the first week teams, consisting of CS and VCD students, will be formed.

Prerequisites

VCD 37000 (or permission), Students should have enough programming and Web usage experience.

Requirements

- Mid-term exam 30%
- Homework assignments 40%
- Project milestones and finished team

project 30%
- No final exam

Attendance in classes is very important. Team leaders will keep attendance records for each class and report missing team members to the instructors by email. Instructors may tak roll calls from time to time. This is to follow new University rules for keeping track of student attendance.

Major Topics

- Introduction to the Internet and Web
- Web publishing tasks and tools
- Web site design methodology
- Hypertext Markup Language --- XHTML, Cascading style sheets
- Tools for creating art and content for Web pages, photoshop, IDLs.
- Cascading Style Sheets (CSS)
- Visual Communication on the Web
- Artistic and logical Design of Web pages
- HTTP and CGI and CGI applications
- Server-side scripting --- Perl, HTTP, and CGI
- Client-side scripting --- DHTML, DOM and JavaScript

Text And References

1. Textbook: *Introduction to Web Design and Programming*
   Paul Wang and Sanda Katila,
   Brooks/Cole Publishing, (to appear) Fall 2003 (Read/print Final Manuscript online)

2. See textbook website for resources and links

Expectations for Students

WDP-1 is the first course in the Web Design and Programming (WDP) sequence that includes also WDP-2 and WDP-Studio. Based on sufficient background in CS (Computer Science) and VCD (Visual Communication Design), WDP-I focuses on the basic principals and techniques for building highly effective and attractive websites conforming to open standards. The course combines and integrates computer science topics with VCD topics. It follows the textbook by Paul Wang and Sanda Katila closely.

While we are not expecting VCD students to become expert programmers we do expect them to pick up basic programming skills in HTML, CSS, and Javascript to help their design work and to collaborate with programmers effectively.

Similarly, while we do not expect CS students to become expert designers, we do expect them to understand design concepts, to know how to use design and image processing tools, and to collaborate with designers by incorporating their design into a well-constructed Web site.

Here we list what we expect the course to teach and students to learn.

Expectations for VCD students

- VCD topics: Understanding and application of core web design principles and elements, information architecture, layout grids, page structure, use of photoshop and dreamweaver.
- CS topics: HTTP concepts, XHTML, CSS, Javascript, HTML+CSS for page layout, form layout, page templates, basic DHTML/DOM, basic form processing

Expectations for CS students

- VCD topics: Understanding of core web design terminology, ability to review, critique and communicate in design terms, basic application of design principles, information architecture, layout grids and page structure.
- CS topics: HTTP protocol, hand coding of XHTML to pass validation, CSS and table for page layout, in-depth CSS, Javascript, DOM, DHTML, forms and form processing (CGI), HTML, CSS, and Javascript trouble shooting/debugging, effective collaboration with designers.

Registration Requirement

The official registration deadline for this course is 09-13-2009 . University policy requires all students to be officially registered in each class they are attending.

Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

The last day to withdraw is 11-08-2009.

Student Accessibility Policy

University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

STUDENT CHEATING AND PLAGIARISM: Condensed Version

[For the complete policy and procedure, go to www.kent.edu/policyregister and search for policy 3342-3-01.8, or see http://www.kent.edu/policyreg/chap3/3-01-8.cfm or http://www.kent.edu/policyreg

/chap3/upload/3342.3.01.8.pdf]

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied. The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. Cheating includes, but is not limited to:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;
2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;

6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;
9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and
10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work.

"Plagiarize" means to take and present as one's own a material portion of the ideas or words of another or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;

b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and

c. The presentation of work prepared by another in final or draft form as one's

own without citing the source, such as the use of purchased research papers.

Academic Sanctions, From Section D

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:

1. Refuse to accept the work for credit; or
2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took place; or
3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;
4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean

or to the vice president for the extended university a recommendation for further sanction under this rule.

Procedures for invoking sanctions (From Section E)

(1) Academic administrative procedures pertaining to paragraph (D)(1)(a) of this rule. In the event that an instructor determines that it is more probable than not that a student in a course or program under the instructor's supervision has presented work for university credit which involves an act of cheating, plagiarism or cooperation in either, then the instructor shall:

(a) Inform the student as soon as is practical, in person or by mail, of the belief that an act of cheating or plagiarism has occurred. If the student cannot be reached in a reasonable period of time, the instructor may proceed with sanctions, notifying the student in writing as promptly as possible of the belief and the procedural steps the instructor has taken.

(b) Provide the student an opportunity to explain orally, in writing, or both, why the student believes the evaluation of the facts is erroneous.

(c) If the explanation is deemed by the instructor to be inadequate or if no explanation is offered, the instructor may impose one of the academic sanctions listed in paragraph (D)(1)(a) of this rule. Where

appropriate, the instructor may recommend the imposition of academic sanctions listed in paragraph (D)(1)(b) of this rule. In addition, the instructor may refer the matter to the dean of the college, campus, or school in which the student is enrolled for imposition of academic sanctions listed in paragraph (D)(1)(b) of this rule.

(d) The instructor shall notify the office of judicial affairs of the circumstances and action taken. Such notification will be used as background information in the event that formal conduct charges are initiated against the student.

(e) The instructor shall inform the student in writing of the right to appeal, and the procedure to follow.

(f) The instructor shall keep the evidence of cheating or plagiarism in a secure place and provide it upon request to any appeals officer or the conduct officer. The instructor shall provide copies on request to the student at the student's expense.

(g) The instructor shall cooperate with academic and student conduct personnel in any appeal of the decision, and/or in adjudication of any disciplinary proceedings.

Academic Appeals

The general principle that applies to the following procedures is that an appeal is directed to the administrative level

immediately above the unit from which the appeal emanates. Appeals are limited to the following reasons:

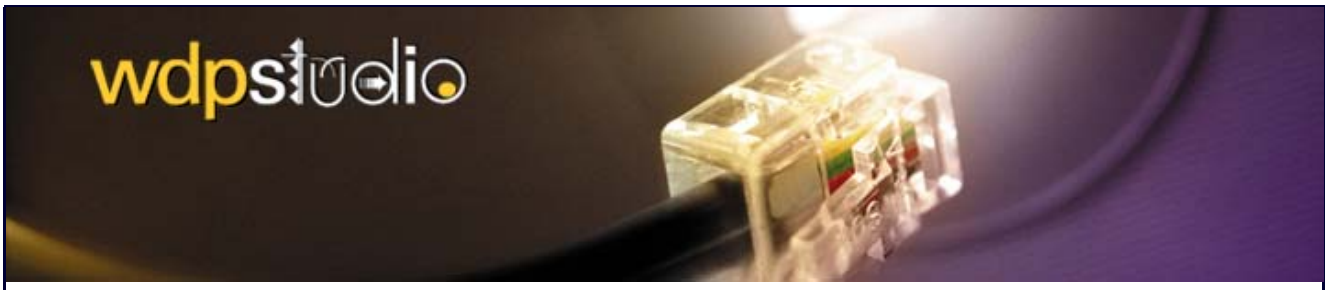
a. The decision is arbitrary or unreasonable,

b. The decision resulted from a procedural error,

c. The decision is not in accordance with the facts presented,

d. New information is available which may suggest modification of the decision.

Go to Top of Page



Overview
Syllabus
Schedule
Projects
Homework
Site Map
Resources
Q & A
Discussions
Notices

SYLLABUS

Catalog Description

Course Number: VCD 46053 and CS 4/57107

Prerequisite: Web Design and Programming-I (CS 4/57106 and VCD 46003), or permission form the instuctor, 6K level CS credit is possible through individual studies offered by the CS instructor (Wang)

Course Description

A joint Computer Science (CS) and Visual Communication Design (VCD) project course exploring advanced aspects of web design and programming. Student may pursue individually defined projects or team projects for real clients. Creative combination and integration of art and programming are emphasized through lectures and labs. Limited enrollment (12 max).

Contents:

COMPUTER SCIENCE:

E-commerce, shopoping carts, stores
Shipping, payment processing, PayPal
Object-oriented Javascript, Actionscript for Flash CS3
XML for the Web, XSL, XSLT, XLink, XPath
CSS for XML
JavaScript and DOM for XML
Defining XML markup: DTD, Schema
Web services, SOAP/REST, WSDL, RSS, PHP support for writing Web service servers and clients
Synchronized Multimedia Integration Language (SMIL), overview and examples

Scallable Vector Graphics, basics, priciples, specification and usage
SVG animation, tools, applications, and advanced features
MathML: Mathematical Markup Language
Web security
Symmetric and public-key encryption, RSA and ECC
Digital signature and message digests
Digital certificates and Certificate Authorities
Secure Socket Layer, OpenSSL, server and client certificates
Apache and OpenSSL support for SSL
Firewalls

VISUAL COMMUNICATION DESIGN:

Separating Content And Style
Design for Usability
Dreamweaver CS3
Working with video in i-Movie
Pod Casting
Designing Product Presentations
Designing Payment Systems
How to Conduct Research
Writing Creative Briefs
The Art of Digital Branding
Flash CS3
Design for Mobile Devices

Term Project:

On top of performing realistic tasks related to site creation for clients, the course also encourages creativity and novel combination of design and programming.

students are encouraged to freely express themselves and create experimental artistic/programming effects that may push the envelope. The creative results will be showcased on the course website.

Course Requirements:

In-class performance and midterm exam (30%), homework (30%), milestones and finished term project (40%).

Textbooks:

References, lecture notes and on-web literature will be used in lieu of textbooks.

Attendance in classes is very important. Team leaders will keep attendance records for each class and report missing team members to the instructors by email. Instructors may tak roll calls from time to time. This is to follow new University rules for keeping track of student attendance.

Read the University Cheating and Plagiarism Policy and Help for students with disabilities.

Expectations for Students

WDP-Studio is the third course in the Web Design and Programming (WDP) sequence that includes also WDP-1 and WDP-2. Students are expected to have taken WDP-1 and WDP-2 or have equivalent background.
WDP-Studio covers advanced topics and techniques for the Web. The class has different expectations for CS and VCD students as described below.

Expectations for VCD students

- VCD topics: The full list of VCD topics contained in the syllabus that are covered in the course.
- CS topics: E-commerce principles, installing and managing a Web store, creating SMIL pages, principles and Web applications of XML, producing SVG graphics, Web security principles

Expectations for CS students

- VCD topics: It is expected that students keep an open mind, welcoming new and creative design possibilities. I expect that they will turn their ideas into effective, aesthetically and technically elegant solutions for global audiences across multiple devices.
- CS topics: The full list of CS topics contained in the syllabus that are covered in the course.

Registration Requirement

The official registration deadline for this course is 09-13-2009 . University

policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

The last day to withdraw is 11-08-2009.

Student Accessibility Policy

University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

STUDENT CHEATING AND PLAGIARISM: Condensed Version

[For the complete policy and procedure, go to www.kent.edu/policyregister and search for policy 3342-3-01.8, or see http://www.kent.edu/policyreg/chap3/3-01-8.cfm or http://www.kent.edu/policyreg/chap3/upload/3342.3.01.8.pdf]

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied. The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. Cheating includes, but is not limited to:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;
2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;
6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;
9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and
10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work. “Plagiarize” means to take and present as one’s own a material portion of the ideas or words of another or to present as one’s own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:
    a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;
    b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and
    c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

Academic Sanctions, From Section D

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is

imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:

1. Refuse to accept the work for credit; or
2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took place; or
3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;
4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

Procedures for invoking sanctions (From Section E)

(1) Academic administrative procedures pertaining to paragraph (D)(1)(a) of this rule. In the event that an instructor determines that it is more probable than not that a student in a course or program under the instructor's supervision has presented work for university credit which involves an act of cheating, plagiarism or cooperation in either, then the instructor shall:

(a) Inform the student as soon as is practical, in person or by mail, of the belief that an act of cheating or plagiarism has occurred. If the student cannot be reached in a reasonable period of time, the instructor may proceed with sanctions, notifying the student in writing as promptly as possible of the belief and the procedural steps the instructor has taken.

(b) Provide the student an opportunity to explain orally, in writing, or both, why the student believes the evaluation of the facts is erroneous.

(c) If the explanation is deemed by the instructor to be inadequate or if no explanation is offered, the instructor may impose one of the academic sanctions listed in paragraph (D)(1)(a) of this rule. Where appropriate, the instructor may recommend the imposition of academic sanctions listed in paragraph (D)(1)(b) of this rule. In addition, the instructor may refer the

matter to the dean of the college, campus, or school in which the student is enrolled for imposition of academic sanctions listed in paragraph (D)(1)(b) of this rule.

(d) The instructor shall notify the office of judicial affairs of the circumstances and action taken. Such notification will be used as background information in the event that formal conduct charges are initiated against the student.

(e) The instructor shall inform the student in writing of the right to appeal, and the procedure to follow.

(f) The instructor shall keep the evidence of cheating or plagiarism in a secure place and provide it upon request to any appeals officer or the conduct officer. The instructor shall provide copies on request to the student at the student's expense.

(g) The instructor shall cooperate with academic and student conduct personnel in any appeal of the decision, and/or in adjudication of any disciplinary proceedings.

Academic Appeals

The general principle that applies to the following procedures is that an appeal is directed to the administrative level immediately above the unit from which the appeal emanates. Appeals are limited to the following reasons:

a. The decision is arbitrary or unreasonable,

b. The decision resulted from a procedural error,

c. The decision is not in accordance with the facts presented,

d. New information is available which may suggest modification of the decision.

| top |

| search |

Overview
Syllabus
Schedule
Projects
Homework
Site Map
Resources
Q & A
Discussion
Notices

SYLLABUS

Instructor Team

- Prof. Sanda Katila (Visual Communication Design)
- Prof. Paul S. Wang (Computer Science)

Prerequisites

- Web Design and Programming I
- CS, VCD and other students with equivalent background may register only after obtaining instructor permission.

Description

This 3-credit class is a joint Computer Science (CS) and Visual Communication Design (VCD) course on advanced Web techniques and multimedia Web design: Web servers, security, advanced design, Java servlets, applets, database connectivity, site maintenance, and creation of actual Web sites by teamwork.

Course Contnet

- Computer Science:
  - Dynamic page generation, active pages (PHP, PEAR, sessions, authentication, HTTP1.1)
  - Database connectivity, database supported webpages (SQL, MySQL, and PHP support for database access)
  - Object-oriented coding (PHP and Javascript)
  - Virtual hosting, content negotiation
  - Web servers and server configuration
  - Website access control
  - Server-side programs as web clients

    - Site and page search
    - Preparing and serving multimedia content: audio and video
    - Scripting for sound and animation (Actionscript for Flash)
- Visual Communication:
    - Programming and interface design; the essential components for effective commercial sites
    - Organizing large volumes of information; integrating effective programming with streamlined design
    - Application of typography, layout and sequencing to web sites
    - Introduction to Audio and Video for the Web for all major platforms
    - Exploring options, tools and effective uses of multimedia for the Web
    - Why, Where and How to Design Animation for the Web
    - Implementing design with advanced Web technology

Course Requirements

- Homework assignments 40%
- Mid-term exam 30%,
- Milestones and Finished team project 30%
- No final exam

Team Project

Students will form Web design and implementation teams to create new sites or upgrade existing sites. The projects will put the course materials to use in realistic situations. Teams will present finished, near production, Websites and submit a joint project report in lieu of final exams. Individual team members will also submit separate written project reports on their own activities and contributions to the team project.

Textbook

An Introduction to Web Design and Programming by Paul Wang and Sanda Katila.

Book ordering information

Expectations for Students

WDP-2 builds on a solid foundation of Web Design and Programming from WDP-1. The focus of WDP-2 is to address more advanced issues of practical importance for creating and maintaining Web sites.

We are not expecting VCD students to become programmers. Although some html, php, and actionscript coding are required when that helps design and site information architecture.

We are not expecting CS students to become designers. Although understanding design concepts, use of some design tools can help the construction and maintenance of a website.

Here we list what we expect the course to teach and students to learn.

Expectations for VCD students

- VCD topics: GOLive/Dreamweaver: understanding page layouts and site architecture. Working with layouts, tables, layers, page structure, templates, rollovers, drop down menus, type and intermediate level CSS to create aesthetically pleasing, well designed websites.

  Flash Foundation: Understanding the basic concepts and logic of interactivity and motion. Creating and working with interaction, buttons, shape and mouse interactions, basic scripting in ActionScipt. Creating and applying external elements such as sound and video in Flash.

- CS topics: what is an active page, understanding PHP in order to appreciate and perhaps use PHP in HTML to organize a page templates into reusable parts, learning about PHP's support of graphics, understanding databases and how they can be applied to help a Web site, action scripting for Flash (some programming here),

including audio and video, Flash in Web pages, understanding server-side access control, user authentication, and how to make a web site fast loading through HTTP 1.1 techniques.

Expectations for CS students

- VCD topics: Design: Learning and applying basic design principles such as hierarchy, contrast, asymmetry and positive/negative space to website design through page layout homework. Developing sensitivity and appreciation for layout and type while applying newly learned skills to homework assignments.

  Flash: Understanding the basic concepts in Flash; leaning program logic and applying ActionScripting to homework and final projects. Working alongside designers and serving as programming consultants on the final projects.

- CS topics: PHP, MySQL, PHP interface to MySQL, Apache, Apache configuration, access control, user authentication and session control, object-oriented Javascript and PHP, ActionScript, HTTP 1.1, HTTP caching, page deflation, audio, video, Flash deployment in Web pages and Javascript control of media, PHP support of graphics, actual use of database to support website operations such as user accounts

Registration Requirement

The official registration deadline for this course is 09-13-2009 . University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected

prior to the deadline.

The last day to withdraw is 11-08-2009.

Student Accessibility Policy

University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through Student Accessibility Services (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

STUDENT CHEATING AND PLAGIARISM: Condensed Version

[For the complete policy and procedure, go to www.kent.edu/policyregister and search for policy 3342-3-01.8, or see http://www.kent.edu/policyreg/chap3/3-01-8.cfm or http://www.kent.edu/policyreg/chap3/upload/3342.3.01.8.pdf]

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied. The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. Cheating includes, but is not limited to:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;

2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;
3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;
4. Securing, giving or exchanging information during examinations;
5. Presenting data or other material gathered by another person or group as one's own;
6. Falsifying experimental data or information;
7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;
8. Cooperating with another to do one or more of the above;
9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and
10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work. “Plagiarize” means to take and present as one’s own a material portion of the ideas or words of another or to present as one’s own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:
    a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;
    b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and
    c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

Academic Sanctions, From Section D

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:

1. Refuse to accept the work for credit; or
2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took place; or
3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;
4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

Procedures for invoking sanctions (From Section E)

(1) Academic administrative procedures pertaining to paragraph (D)(1)(a) of this rule. In the event that an instructor determines that it is more probable than not that a student in a course or program under the instructor's supervision has presented work for university credit which involves an act of cheating, plagiarism or cooperation in either, then the instructor shall:

(a) Inform the student as soon as is practical, in person or by mail, of the belief that an act of cheating or plagiarism has occurred. If the student cannot be

reached in a reasonable period of time, the instructor may proceed with sanctions, notifying the student in writing as promptly as possible of the belief and the procedural steps the instructor has taken.

(b) Provide the student an opportunity to explain orally, in writing, or both, why the student believes the evaluation of the facts is erroneous.

(c) If the explanation is deemed by the instructor to be inadequate or if no explanation is offered, the instructor may impose one of the academic sanctions listed in paragraph (D)(1)(a) of this rule. Where appropriate, the instructor may recommend the imposition of academic sanctions listed in paragraph (D)(1)(b) of this rule. In addition, the instructor may refer the matter to the dean of the college, campus, or school in which the student is enrolled for imposition of academic sanctions listed in paragraph (D)(1)(b) of this rule.

(d) The instructor shall notify the office of judicial affairs of the circumstances and action taken. Such notification will be used as background information in the event that formal conduct charges are initiated against the student.

(e) The instructor shall inform the student in writing of the right to appeal, and the procedure to follow.

(f) The instructor shall keep the evidence of cheating or plagiarism in a secure place and provide it upon request to any appeals officer or the conduct officer. The instructor shall provide copies on request to the student at the student's expense.

(g) The instructor shall cooperate with academic and student conduct personnel in any appeal of the decision, and/or in adjudication of any disciplinary proceedings.

Academic Appeals

The general principle that applies to the following procedures is that an appeal is directed to the

administrative level immediately above the unit from which the appeal emanates. Appeals are limited to the following reasons:

a. The decision is arbitrary or unreasonable,

b. The decision resulted from a procedural error,

c. The decision is not in accordance with the facts presented,

d. New information is available which may suggest modification of the decision.

Overview
Syllabus
Homework
Notes
Listserv
Q & A
Resources
Code Package
Hands-on
Site Map

Syllabus

In-depth course on the theory and principles of Object-based and Object-oriented programming using a language such as Java. The course covers advanced topics in inheritance and interface planning, generic (type-independent) programming, factors of polymorphism, structure and organization of class libraries, GUI and event driven programming, applets/networking, multithreading, and object-oriented design principles.

Prerequisite: CS 33001 (data structures)

Contents

- Object-based Programming --- Classes and Objects (a review)
- Principles of Inheritance and Class Modification/Refinement
- Inheritance Planning and Effective Class

Scientific Visualization - Syllabus

General Information:

Course: CS 6/79995, Fall 2008

Time: Tuesday, Thursday 2:15 pm -03:30 pm Room: MSB 276

Instructor:

Ye Zhao, Assistant Professor Office: MSB 220 Email: zhao@cs.kent.edu

Office Hours: Tuesday, Thursday 1:15 pm - 2:15 pm

Goal:

The course discusses the visualization techniques of scientific, medical, and information data sets. We will show how the important human factor will play a significant role in many science, business, health and industry areas.

Topics include: data acquisition of computed, sampled or synthetic data, visual perception, basic graphics and imaging concepts, volume and surface visualization, and information visualization. This course presents introductory as well as some advanced topics on visualization, and students will have the opportunity to further explore an interested topic in a final programming project.

Prerequisite:

Programming skills in C, C++ or java are required. Knowledge of OpenGL is preferred.

Course Syllabus:

Topics covered will include (may be changed):

Introduction to visualization
Graphics and visualization system
- *Structure and pipeline*
- *Programming language*
- *Graphics hardware and software model*

Visual perception
- *Eye model and basic perception concept*
- *Color system*

Data model
- *1D, 2D and 3D data*
- *Scalar and vector field*
- *Data acquisition techniques*

2D image data
- *Histogram*
- *Image processing basics*

Sampling theory and anti-aliasing
3D volume data
- *Spatial transformation*
- *Ray-casting*
- *Transfer function*
- *Illumination and shading*
- *Isovalue, isosurfaces and surface visualization*

Information visualization techniques
- *Multiple variant database visualization*
- *Parallel Coordinates*
- *Time-varied data set*

Text:

No single textbook covers all the material of this course. We will make class notes and papers available instead.

We recommend the textbooks below:

Helen Wright, ***Introduction to Scientific Visualization,*** Springer, 2006

James D. Foley, Andries van Dam, Steven K. Feiner andJohn F. Hughes, ***Computer Graphics : Principles & Practices***, Addison Wesley, 2nd edition in C, 1995

K. Engel, M. Hadwiger, J. Kniss, C. Rezk-Salama, and D. Weiskopf , ***Real Time Volume Graphics,*** A.K. Peters, 2006

Colin Ware, ***Information Visualization: Perception for Design***, Morgan-Kaufman, 1999

Assessment:

No paper examinations for the course.

Grading: Home works: 40%; Programming projects: 40%; Final project and presentation: 20%

Submission:

All programs should conform to the submission standards given in URL
http://www.cs.kent.edu/~zhao/vis08/submission.htm

Notes:

Home works and project submission deadlines are firm. There will be a penalty for late submission. This syllabus and most subsequent information on the course will be available using the WWW. The home page for the course is: www.cs.kent.edu/~zhao/vis08/index.htm

Academic Integrity:

All programs submitted must be your own work, and you are expected to develop your programs independently. You may receive as much help as you wish on the use of the operating system, text editors, debuggers, file transfer protocols and so on. You may consult with other members of the class about interpreting the projects, and you may get help in finding bugs, but not fixing bugs, but you are not allowed to look at or copy another person's code or discuss design decisions with others, and you cannot show your code to others. Students found to be in violation of these guidelines will fail the project, and will be reported to the dean.

Students with Disabilities:

University Policy 3342-3-01.3 requires that students with disabilities be provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact the instructor at the beginning of the semester to make arrangements for necessary classroom adjustments. **Please note, you must first verify your eligibility for these through Student Accessibility Services** (contact 330-672-3391 or visit www.kent.edu/sas for more information on registration procedures).

Registration Requirement: The official registration deadline for this course is September 7, 2008. University policy requires all students to be officially registered in each class they are attending. Students who are not officially registered for a course by published deadlines should not be attending classes and will not receive credit or a grade for the course. Each student must confirm enrollment by checking his/her class schedule (using Student Tools in FlashFast) prior to the deadline indicated. Registration errors must be corrected prior to the deadline.

Excerpt from the University's

Administrative policy and procedures regarding student cheating and plagiarism.

Excerpted from University Policy Register #3342-3-07

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

"**Cheat**" means intentionally to misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. Such misrepresentations may, but need not necessarily, involve the work of others. As defined, **cheating includes, but is not limited to**:

Obtaining or retaining partial or whole copies of examination, tests or quizzes before these are distributed for student use;

Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;

Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;

Securing, giving or exchanging information during examinations;

Presenting data or other material gathered by another person or group as one's own;

Falsifying experimental data or information;

Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;

Cooperating with another to do one or more of the above; and

Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented.

Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work.

"**Plagiarize**" means to take and present as one's own a material portion of the ideas or words of another or to present as one's own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

(a) The copying of words, sentences and paragraphs directly from the work of another without proper credit;

(b) The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings, of another without proper credit; and

(c) The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

ADMINISTRATIVE POLICY AND PROCEDURES REGARDING STUDENT CHEATING AND PLAGIARISM

Condensed Version

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.

Cheating and plagiarism constitute fraudulent misrepresentation for which no credit can be given and for which appropriate sanctions are warranted and will be applied.

The university affirms that acts of cheating and plagiarism by students constitute a subversion of the goals of the institution, have no place in the university and are serious offenses to academic goals and objectives, as well as to the rights of fellow students.

"Cheat" means to intentionally misrepresent the source, nature, or other conditions of academic work so as to accrue undeserved credit, or to cooperate with someone else in such misrepresentation. Cheating includes, but is not limited to:

1. Obtaining or retaining partial or whole copies of examinations, tests or quizzes before these are distributed for student use;

2. Using notes, textbooks or other information in examinations, tests and quizzes, except as expressly permitted;

3. Obtaining confidential information about examinations, tests or quizzes other than that released by the instructor;

4. Securing, giving or exchanging information during examinations;

5. Presenting data or other material gathered by another person or group as one's own;

6. Falsifying experimental data or information;

7. Having another person take one's place for any academic performance without the specific knowledge and permission of the instructor;

8. Cooperating with another to do one or more of the above;

9. Using a substantial portion of a piece of work previously submitted for another course or program to meet the requirements of the present course or program without notifying the instructor to whom the work is presented; and

10. Presenting falsified information in order to postpone or avoid examinations, tests, quizzes, or other academic work.

“Plagiarize” means to take and present as one’s own a material portion of the ideas or words of another or to present as one’s own an idea or work derived from an existing source without full and proper credit to the source of the ideas, words, or works. As defined, plagiarize includes, but is not limited to:

a. The copying of words, sentences and paragraphs directly from the work of another without proper credit;

b. The copying of illustrations, figures, photographs, drawings, models, or other visual and nonverbal materials, including recordings of another without proper credit; and

c. The presentation of work prepared by another in final or draft form as one's own without citing the source, such as the use of purchased research papers.

Academic Sanctions

The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. Kent campus instructors shall notify the department chairperson and the student conduct office each time a sanction is imposed. Regional campus instructors shall notify the regional campus dean and the student conduct officer each time a sanction is imposed. Regional campus student conduct officer shall notify the Kent student conduct office each time a sanction is imposed by a regional campus Instructor. The following academic sanctions are provided by this rule for offenses of cheating or plagiarism. In those cases the instructor may:

1. Refuse to accept the work for credit; or

2. Assign a grade of "F" or zero for the project, test, paper, examination or other work in which the cheating or plagiarism took place; or

3. Assign a grade of "F" for the course in which the cheating or plagiarism took place; and/or;

4. Recommend to the department chair or regional campus dean that further action specified in the rule be taken. The department chairperson or regional campus dean shall determine whether or not to forward to the academic dean or to the vice president for the extended university a recommendation for further sanction under this rule.

Academic Appeals

The general principle that applies to the following procedures is that an appeal is directed to the administrative level immediately above the unit from which the appeal emanates.

Appeals are limited to the following reasons:

a. The decision is arbitrary or unreasonable,

b. The decision resulted from a procedural error,

c. The decision is not in accordance with the facts presented,

d. New information is available which may suggest modification of the decision.

For complete policy and procedure go to www.kent.edu/policyregister 3342-3-01.8.

http://www.kent.edu/policyreg/chap3/upload/3342.3.01.8.pdf

Section B3:
CS Colloquium Talks (August 2004 – August 2009)

Spring 2009

1. February 11, 2009, *Virus Quasispecies Assembly using Network Flows*, Dr. Alexander Zelikovsky, Georgia State University, Atlanta, GA
2. February 20, 2009, *Graph Coloring: history, generalizations, algorithms and open problems*, Dr. Vitaly I. Voloshin, Troy University, Troy, AL
3. March 6, 2009, *Computational Modeling of Gene Regulatory Networks using Temporal Gene Expression Data*, Dr. Joe Song, New Mexico State University, Las Cruces, NM
4. March 11, 2009, *Computational methods for haplotype inference from general pedigrees*, Dr. Jing Li, Case Western Reserve University, Cleveland, Ohio
5. April 1, 2009, *Computational Thinking*, Dr. Jeannette M. Wing, President's Professor of Computer Science, Carnegie Mellon University, Pittsburgh, PA, Assistant Director, CISE Directorate, National Science Foundation
6. April 15, 2009, *Graph sandwich problems*, Dr. R. Sritharan, The University of Dayton, Dayton, Ohio
7. April 24, 2009, *Spatiotemporal Stream Mining using EMM*, Professor Margaret H. Dunham, Southern Methodist University, Dallas, Texas
8. May 1, 2009, *Incremental Pattern Discovery on Streams and Graphs*, Dr. Jimeng Sun, IBM Thomas J. Watson Research Center, Hawthorne, NY

Fall 2008

1. September 26, 2008, *Solvers*, Dr. Victor W. Marek, University of Kentucky, Lexington, KY
2. October 1, 2008, *Functional Coherence in Molecular Interaction Networks*, Dr. Mehmet Koyuturk, Case Western Reserve University, Cleveland, OH
3. October 15, 2008, *Decomposition of graphs into factors*, Dr. Martín Matamala, Universidad de Chile, Santiago, Chile
4. October 24, 2008, *Information Retrieval support for Concept Location in Software*, Dr. Andrian Marcus, Wayne State University, Detroit, Michigan
5. November 5, 2008, *A Workflow for Analyzing Large Histological Image Datasets in Biomedical Research*, Dr. Kun Huang, Ohio State University, Columbus, Ohio
6. November 19, 2008, *Sparse Recovery Using Sparse Random Matrices*, Dr. Piotr Indyk, MIT, Cambridge, Massachusetts

Spring 2008

1. February 6, 2008, *Recent results on graph searching*, Dr. Derek Corneil, University of Toronto, Ontario, Canada
2. February 27, 2008, *The Inframetric Model for the Internet*, Dr. Laurent Viennot, Université Paris Diderot, Paris France
3. April 4, 2008, *Synchrony, Modality and Semantics: Exploration of Predicate Detection*, Chunbo Chu, Wayne State University, Detroit, MI
4. April 7, 2008, *Formal Verification and Testing in SAM*, Gonzalo Argote-Garcia, Florida International University, Miami, FL
5. April 16, 2008, *It's Not "Junk": Using Genetic Algorithms to Search for Functional Regions*, Dr. Clare Bates Congdon, University of Southern Maine, Portland, ME

Fall 2007

1. September 7, 2007, *ChemXSeer: A Digital Library for Chemical Kinetics Data & Scientific*, Prasenjit Mitra, Pennsylvania State University, Pennsylvania
2. September 19, 2007, *PET/CT: Theory and Applications*, Chi-Hua Tung, Phillips Medical Systems, Cleveland, OH
3. October 3, 2007, *Research on Adaptive Contents Service for Sensibility and Context*, Jae-Dong Lee, Dankook University, Korea
4. October 31, 2007, *Computing in the Era of Multi-core and Virtualisation*, Hong Ong, Oak Ridge National Laboratory, Oak Ridge, TN
5. November 14, 2007, *Data-Intensive Super Computing: Taking Google-Style Computing Beyond Web Search*, Randal E. Bryant, Carnegie Mellon University, Pittsburgh, PA

Spring 2007

1. January 26, 2007, *Data Mining using Fractals and Power Laws*, Christos Faloutsos, Carnegie Mellon University, Pittsburgh, PA
2. February 14, 2007, *Communication predicates: A high level abstraction for coping with transient and dynamic faults*, André Schiper, Swiss Federal Institute of Technology (EPFL), Switzerland
3. February 21, 2007, *DipZoom: An Internet Measurements Marketplace*, Misha Rabinovich, Case Western Reserve University, Cleveland, Ohio
4. March 7, 2007, *Emerging Networking Technologies and Protocols for Next Generation Clusters*, Dhabaleswar K. (DK) Panda, Ohio State University, Ohio
5. March 14, 2007, *Virtual Authentication Ring for Securing Network Operations*, David Lee, The Ohio State University, Ohio
6. March 21, 2007, *Modeling Natural Phenomena with Lattice Boltzmann Method*, Ye Zhao, Kent State University, Kent, OH
7. April 11, 2007, *Applications of Operations Research in Justice Studies*, David Kessler, Kent State University, Kent, OH
8. April 18, 2007, *A self-stabilizing algorithm for computing the median set of a squaregraph*, Dr. Yann Vaxès, Université de la Mediterranée, Marseille, France
9. April 25, 2007, *The Problems With Counting Ancestors in a Simple Genetic Algorithm*, Mark Wineberg, University of Guelph, Ontario, Canada
10. May 2, 2007, *Mining Evolution of Graph Data*, Qiankun Zhao, Penn State University, University Park, PA

Fall 2006

1. September 18, 2006, *Network Approaches and Cellular Metabolism*, Eivind Almaas, Lawrence Livermore National Laboratory, California
2. September 27, 2006, *Bandwidth Allocation in Sense-and-Respond Systems*, Vincenzo Liberatore, Case Western Reserve University, Cleveland, Ohio
3. October 6, 2006, *Internet 3.0: Ten Problems with Current Internet Architecture and Solutions for the Next Generation*, Raj Jain, Washington University of St. Louis, Missouri
4. October 18, 2006, *A Dynamic Quality-of-Service based Dynamic Resource Manager*, Frank Drews, Ohio University, Athens, Ohio
5. October 25, 2006, *Resource Management for Dynamic, Distributed Real-Time Systems*, Dazhang Gu, CIDDS of Ohio University, Ohio
6. November 8, 2006, *Scattered Data Visualization*, Yingcai Xiao, University of Akron, Akron, Ohio
7. November 15, 2006, *Toward Large Scale Self-stabilizing Systems*, Sébastien Tixeuil, Université Paris Sud, France
8. November 27, 2006, *Opportunistic Routing and Middleware for Sensor and Actuator Networks*, Joel Branch, Rensselaer Polytechnic, New York

9. November 29, 2006, *Techniques for Optimizing Loop Scheduling*, Tim O'Neil, University of Akron, Akron, Oh

Spring 2006

1. January 27, 2006, *Tools and Techniques for the Data Grid*, Prof. Gagan Agrawal, Ohio State University, Ohio, USA
2. February 13, 2006, *Effects of Large, High-Resolution Displays on Geospatial Information Visualization*, Robert Ball, Virginia Polytechnic Institute and State University, VA, USA
3. February 20, 2006, *Using Lattice Boltzmann Method for Natural Phenomena Modeling*, Ye Zhao, Stony Brook (SUNY), NY, USA
4. February 22, 2006, *High-Fidelity, Illustrative, and Exploratory Visualization*, Xiaoru Yuan, University of Minnesota, MN, USA
5. February 24, 2006, *Point-Based Graphics: A Modeling, Animation, and Simulation Paradigm*, Xiaohu Guo, Stony Brook University, NY, USA
6. February 27, 2006, *Interactive, Time-Dependent Flow Visualization Techniques for CFD Simulation Data*, Dr. Robert Laramee, Vienna University of Technology, Vienna, Austria
7. March 10, 2006, *Static Techniques for Concept Location in Object-Oriented Code*, Prof. Vaclav Rajlich, Wayne State University, Michigan, US
8. March 15, 2006, *Extraterrestrials, Cryptanalysis, and Genomes: Perspectives on Bioinformatics Research*, Prof. Lonnie Welch, Stuckey Professor of Electrical Engineering & Computer Science Center for Intelligent, Distributed and Dependable Systems, Ohio University, Ohio, USA
9. April 7, 2006, *High Performance Computing and Atmospheric Modeling*, John Michalakes, NCAR, Boulder,CO, USA
10. April 12, 2006, *Distributed Smart Cameras: Algorithms and Architectures*, Prof. Wayne Wolf, Princeton University, Princeton, NJ
11. April 14, 2006, *On partitioning of hypergraphs*, Dr. Sergei Bezrukov, University of Wisconsin - Superior, WI, USA
12. April 19, 2006, *Computational Tire Modeling*, Dr. Timothy A. Davis, Goodyear, Ohio
13. April 26, 2006, *Grid Robots: Getting the computer to program your robot for you*, Prof. Daniel Ashlock, University of Guelph, Ontario, Canada
14. May 5, 2006, *A rounding algorithm for approximating minimum Manhattan networks*, Prof. Victor Chepoi, Universite de la Mediterranee, Marseille, France

Fall 2005

1. Friday October 7, 2005, *Tutorial on Geometric Level Set for Medical Imagery*, Jasjit S. Suri, Ph.D., Columbus, Ohio
2. October 12, 2005, *A Network Management Framework for Next Generation Networks*, Dr. Augustine Samba, Kent State University, Kent, Ohio
3. October 19, 2005, *Science Informatics and Information Management*, Dr. John Rumble, Information International Associates, Inc., Oak Ridge, TN.
4. October 26, 2005, *Wireless Networked Sensors: Introductions, Applications, Demonstrations and Future Work*, Prof. Mikhail Nesterenko, Kent State University, Kent, Ohio
5. November 9, 2005, *An Overview of the CISE Directorate at NSF*, Dr. Michael Foster, Division Director, National Science Foundation, Arlington, VA
6. November 18, 2005, *Robust Resource Allocation in Parallel and Distributed Computing Systems*, Prof. H. J. Siegel, Colorado State University, Colorado, USA
7. November 21, 2005, *Frequent Pattern Mining: Algorithm, Research Issues, and Applications*, Prof. Ruoming Jin, Kent State University, Ohio, USA
8. November 30, 2005, *The case for vigilance*, Prof. Michael Rothstein, Kent State University, Ohio, USA

9. December 7, 2005, *Next-Generation Information Systems,* Professor Avi Silberschatz, Yale University, Connecticut

Spring 2005

1. February 2, 2005, *Computational Problems in Population Haplotyping,* Prof. Alex Zelikovsky, Georgia State University, Atlanta, GA
2. February 18, 2005, *Strategyproof Network Protocol Design,* Prof. Xiang-Yang Li, Illinois Institute of Technology, Chicago, IL
3. February 25, 2005, *Algorithms for Universal DNA Tag Array Design and Optimization,* Prof. Ion Mandoiu, University of Connecticut, Storrs, CT
4. March 4, 2005, *Network Lifetime and Power Assignment in Ad-Hoc Wireless Networks,* Prof. Gruia Calinescu, Illinois Institute of Technology, Chicago, IL
5. March 7, 2005, *Image Processing Methods For Large-Scale Studies of Neurodegenerative Disease,* Dr. Owen Carmichael (Faculty Candidate), University of Pittsburgh and Carnegie Mellon University, Pittsburgh, PA
6. March 9, 2005, *Scalable Data Mining: System and Algorithms,* Ruoming Jin (Faculty Candidate), Ohio State University, Columbus, OH
7. March 11, 2005, *Some Issues in the Extreme Scaling of Wireless Sensor Networks,* Prof. Anish Arora, The Ohio State University, Columbus, OH, USA
8. March 14, 2005, *Similarity-Cornerstone of Bioinformatics, or "How One Can Compare Apples to Oranges"*, Dr. Mikhail Popescu (Faculty Candidate), University of Missouri-Columbia, Columbia, Missouri
9. March 16, 2005, *Mass Spectrometry Data Analysis for Biomarker Discovery,* Dr. Weichuan Yu, Associate Research Scientist (Faculty Candidate), Center for Statistical Genomics and Proteomics, New Haven, CT
10. March 30, 2005, *SORTING NETWORKS,* Prof. Kenneth E. Batcher, Kent State University, Kent, OH
11. April 6, 2005, *On Self-stabilization and Sensor networks,* Prof. Sebastien Tixeuil, Universite Paris Sud, France
12. April 22, 2005, *Efficient Algorithms for Exploring Large-Scale Multidimensional Time Series Data*, Prof. Joseph F. JaJa, University of Maryland, College Park, MD
13. April 27, 2005, *On clique separators, nearly chordal graphs and the Maximum Weight Stable Set problem*, Prof. Dr. Andreas Brandstädt, Universität Rostock, Rostock, Germany
14. May 4, 2005, *High-Performance Computing for Reconstructing Evolutionary Trees from Gene-Order Data,* David A. Bader, Associate Professor and Regents' Lecturer, Univ. of New Mexico, Albuquerque, NM

Fall 2004

1. September 15, 2004, *Green's functions for singularly perturbed problems,* Dr. Torsten Linss, Technical University of Dresden, Germany
2. October 29, 2004, *Metric Methods: Algorithms and Applications,* Dr. Anupam Gupta, Carnegie Mellon University, Pittsburgh, PA
3. November 12, 2004, *Computational Geometry and Biology,* Dr. Sergey (Bereg) Bespamyatnikh, University of Texas at Dallas, Richardson, TX
4. November 29, 2004, *Static and Dynamic Analyses for Reverse-Engineered UML Sequence Diagrams,* Dr. Atanas (Nasko) Rountev, The Ohio State University, Columbus, OH
5. December 1, 2004, *A Theoretical and Applied Program to Simulate Very Large Social and Infrastructure Systems,* Dr. Madhav Marathe, Los Alamos National Laboratory, Los Alamos, NM , USA

Appendix C

Section C1:

Ethnicity	Number of Males	Number of Females
African-American or Black	0	0
Asian or Pacific Islander	6	0
Hispanic or Latin American	1	0
Native American	0	0
White	11	2
Multiple ethnicities or other	0	0

Section C2:

FACULTY MEMBER NAME (Last, First)	APPT	STATUS	RANK	BOOK	CHAP	ART	MONO	CONF. PAPER	NR PUBS	PRES	COLL	REV
Baker, Johnnie	1973	Ft-tt	Prof	0	0	0	0	10	2	6	0	0
Bansal, Arvind	1988	Ft-tt	Prof	0	0	4	0	11	0	2	3	0
Batcher, Kenneth	1989	Ft-tt	Prof	0	0	1	0	1	0	1	0	0
Breitbart, Yuri	2002	Ft-tt	Prof	1	0	8	0	12	0	0	0	0
Dragan, Feodor	2000	Ft-tt	Assoc	0	0	19	0	18	0	13	13	0
Farrell, Paul	1986	Ft-tt	Prof	0	0	4	0	9	1	6	0	0
Guercio, Angela	2008	Ft-tt	Asst	0	1	1	0	7	0	2	0	0
Jin, Ruoming	2005	Ft-tt	Asst	0	2	9	0	33	0	0	8	0
Khan, Javed	1997	Ft-tt	Assoc	2	0	9	0	42	0	23	0	0
Lu, Cheng-Chang	1988	Ft-tt	Prof	0	0	0	0	12	0	9	0	0
Maletic, Jonathan	2001	Ft-tt	Assoc	0	2	0	0	41	2	19	3	0
Melton, Austin	1996	Ft-tt	Assoc	0	0	0	0	9	1	11	0	0
Nesterenko, Mikhail	1998	Ft-tt	Assoc	0	0	7	0	20	0	7	0	0
Peyravi, Hassan	1985	Ft-tt	Prof	0	0	2	0	6	0	10	0	0
Rothstein, Michael	1980	Ft-tt	Assoc	0	0	0	0	0	0	1	0	0
Ruttan, Arden	1983	Ft-tt	Prof	0	0	0	0	5	0	0	0	0
Volkert, Gwenn	2001	Ft-tt	Assoc	0	0	0	0	11	0	4	6	0
Walker, Robert	1997	Ft-tt	Prof	0	0	1	0	8	0	0	0	0
Wang, Paul	1977	Ft-tt	Prof	1	0	0	0	14	0	0	0	0
Zhao, Ye	2006	Ft-tt	Asst	0	1	5	0	10	0	16	0	0

Section C3:

FACULTY MEMBER (Last, First)	GCOURSE HRS	UCOURSE HRS	R MENTOR	T MENTOR	PLACE	MAS CTM	MAS CHR	DIS CTM	DIS CHR
Baker, Johnnie	5.2	2	6	5		15	2	3	1
Bansal, Arvind	0.9	6.9	3	0		5	2	2	1
Batcher, Kenneth	4.8	7.2	2	2		8	0	8	2
Breitbart, Yuri	4.5	2.1	5	2		6	3	2	1
Dragan, Feodor	7.5	2.1	15	3		5	12	6	2
Farrell, Paul	4.8	0.6	4	1		12	1	5	0
*Guercio, Angela	0	25.5	0	0					
Jin, Ruoming	4.5	1.5	11	4		10	4	2	0
Khan, Javed	4.8	4.5	7	8		7	6	2	5
Lu, Cheng-Chang	3.8	3.4	6	3		7	1	1	3
Maletic, Jonathan	3.4	4.2	24	6		2	5	3	1
Melton, Austin	0	4.8	13	0		7	13	6	0
Nesterenko, Mikhail	2.1	5.1	11	1		6	8	1	0
Peyravi, Hassan	7.5	1.5	7	3		17	3	9	0
Rothstein, Michael	3.3	8.4	4	1		8	3	0	0
Ruttan, Arden	2.4	3.6	4	2		9	3	4	1
Volkert, L. Gwenn	2.5	4.9	3	3		4	2	2	0
Walker, Robert	3.3	2.7	7	3		10	3	1	2
Wang, Paul	5.7	7.2	9	6		6	2	3	0
Zhao, Ye	5.5	0.5	8	2		3	1	0	0

*Regional Campus

Appendix C4
List the full references for scholarly or creative works that best represent the quality of the scholarship being produced by program faculty over the past five years:

Baker, Johnnie

1. Shannon Steinfadt and Johnnie. W. Baker, "*SWAMP: Smith-Waterman using Associative Massive Parallelism*", IEEE Workshop on Parallel and Distributed Scientific and Engineering Computing, 2008 International Parallel and Distributed Processing Symposium (IPDPS) at Miami, 8 pages, published on CD, April 14-18, 2008.
2. Mingxian Jin and Johnnie W. Baker "*Two Graph Algorithms on an Associative Computing Model"*, 2007 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'07), Las Vegas, 7 pages, June 25-28, 2007.
3. Shannon Steinfadt, Michael Scherger, and Johnnie W. Baker, "*A Local Sequence Alignment Algorithm Using an Associative Model of Parallel Computation*", Proc. of IASTED Computational and Systems Biology (CASB 2006), Dallas, pp. 38-43, Nov. 13-14, 2006.

Bansal, Arvind

1. A. Bansal, "Role of Bioinformatics in the Development of Anti Infective Therapy," *Expert Review of Anti-Infective Therapy*, Vol. 6, Issue 1, 2008, pp. 51-63
2. A. Guercio, A. K. Bansal, T. Arndt, "Language Constructs and Synchronization in Reactive Multimedia Systems," *International Transaction on Computers and Software Engineering*, No. 1, Vol. 1, 2007, ISSN: 1797-1152, pp. 52-59
3. A. Guercio and A. K. Bansal, "Towards a Formal Semantics for Distributed Multimedia Computing," *International Conf. on Distributed Multimedia Systems*, San Francisco, USA, 2007, pp. 81-86

Breitbart, Yuri

1. Breitbart, Y., Blott, S., Baulier, J., Korth, H., "Real-Time Event-Processing System With Service Authoring Environment", US patent #6,681,230, 2004.
2. Breitbart, Y., Garofalakis, M., Martin, C., Rastogi, R., Seshadri, S., Silberschatz, A., "Determination of Physical Topology of A Communication Network", US patent #6,697,338, 2004.
3. Breitbart, Y., Blott, S., Martin, C., "Method And Apparatus For Applying Once-Only Processing In A Data Network", US patent #6,721,314, April 2004

Dragan, Feodor

1. Effective Monitor Placement in Internet Networks Y. Breitbart , F.F. Dragan, H. Gobjuka, *Journal of Networks* 4 (2009), 657-666.
2. Compact and Low Delay Routing Labeling Scheme for Unit Disk Graphs C. Yan, Y. Xiang and F.F. Dragan, *The Algorithms And Data Structures Symposium (WADS 2009)*, Banff Conference Centre, Banff, Alberta, Canada, 21-23 August 2009, Springer, *Lecture Notes in Computer Science*, 5664, pp. 566–577.
3. Collective Tree Spanners for Unit Disk Graphs with Applications, F.F. Dragan, Y. Xiang and C. Yan, *Electronic Notes in Discrete Mathematics* 32 (2009) 117–124.

Farrell, Paul

1. P.A. Farrell, E. O'Riordan, G.I. Shishkin, Robust numerical methods for singularly perturbed semilinear differential equations with interior layers, Proceedings of BAIL 2004, Toulouse, France, 6 pages, (2004)
2. P.A.Farrell, A.F.Hegarty, J.J.H. Miller, E. O'Riordan and G.I. Shishkin, Global maximum norm parameteruniform numerical method for a singularly perturbed convection-diffusion problem with discontinuous convection coefficient, Mathematical and Computer Modelling, 40, pp. 1375-1392, (2004)
3. P.A. Farrell, E. O'Riordan, G.I. Shishkin, A class of singularly perturbed semilinear differential equations with interior layers, Math. Comp., 74, 2005, 1759-1776

Guercio, Angela

1. A. Guercio, A. Bansal, T. Arndt, "Languages Constructs and Synchronization in Reactive Multimedia Systems", *ISAST Transactions on Computers and Software Engineering*, ISSN 1797-1152, vol. 1, no.1, pp. 52-58, 2007.
2. A. Guercio, T. Arndt, "Towards Synchronization of a Distributed Orchestra", *to appear in the Proc. of DMS 2009, The 15th Intl. Conf. on Distributed Multimedia Systems*, San Francisco Bay, USA, September 10-12, 2009.
3. T. Arndt, A. Guercio, "XML-Based Course Material Transformations For Ubiquitous E-learning Applications", *to appear in the Proc. of KMIS 2009, The Intl. Conf. on Knowledge Management and Information Sharing*, Madeira, October 6-8, Portugal, 2009.

Jin, Ruoming

1. MMIS07, 08: mining multiple information sources workshop report, Xingquan Zhu, Ruoming Jin, Yuri Breitbart, Gagan Agrawal, SIGKDD Explorations 10(2): 61-65 (2008)
2. Data Discretization Unification, Ruoming Jin, Yuri Breitbart, and Chibuike Muoh, in Knowledge and Information System (KAIS journal), Volume 19, Number 1, Pages 1-29, April, 2009.
3. Middleware for Data Mining Applications on Clusters and Grids, Leonid Glimcher, Ruoming Jin, Gagan Agrawal, Journal of Parallel and Distributed Computing (JPDC), 68(1): 37-53 (2008).

Khan, Javed

1. Knowledge Based Characterization of Test Questions, Javed I. Khan & Manas S. Hardas, Kent State University, Handbook of Research on Modern Systems Analysis and Design Technologies and Applications, ISBN: 978-1-59904-887-1, Publisher: Information Science Reference, Pub Date: July 2008.
2. Extreme Rate Distributed Video Transcoding System, Seung S. Yang, Virginia State University, USA, Javed I. Khan, Kent State University, USA, Multimedia Transcoding in Mobile and Wireless Networks, ISBN: 978-1-59904-984-7, Publisher: Information Science Reference, Pub Date: July 2008.
3. WIC: Javed I. Khan & Sajid S. Shaikh, A Phenotype Reputation Estimation Function and its Study of Resilience to Social Attacks, **Journal of Network and Computer Application**, Elsevier, (April 2009, accepted Dec 2008, 1084-8045, Published by Elsevier Ltd. doi:10.1016/j.jnca. 2008.12.003).

Lu, Cheng-Chang

1. Yujun Guo, Wayne Cheng, Cheng-Chang Lu, ``Non-Rigid Mammogram Registration using Demons Algorithm,'' IASTED International Conference on Signal and Image Processing, Honolulu, Hawaii, August 2007.
2. Dee Wu, Yujun Guo, Cheng-Chang Lu, Jasjit S. Suri, "Improvement to Functional Magnetic Resonance Imaging (fMRI) Methods Using Non-Rigid Body Image Registration Methods for Correction in Presence of Susceptibility Artifact Effects," IEEE Engineering in Medicine and Biology Society Conference, pp. 1018-1020, August 2006, New York.
3. Yujun Guo, Cheng-Chang Lu, "Multi-modality Image Registration Using Mutual Information Based on Gradient Vector Flow," International Conference on Pattern Recognition 2006, Hong Kong, p.697-700, August 2006.

Maletic, Jonathan

1. Sutton, A. and Maletic, J. I., (2008), "Automatically Identifying C++0x Concepts in Function Templates", in Proceedings of the 24th IEEE International Conference on Software Maintenance (ICSM'08), Beijing China, Sept. 28 – Oct. 4, pp. 57-66.
2. Kagdi, H. and Maletic, J. I., (2007), "Mining Evolutionary Dependencies from Web-Localization Repositories", Journal of Software Maintenance and Evolution: Research and Practice, vol. 19, no. 5, September, pp. 315-337.
3. Sutton, A. and Maletic, J. I., (2007), "Recovering UML Class Models from C++: A Detailed Explanation", Information and Software Technology, vol. 49, no. 3, March, pp. 212-229.

Melton, Austin

1. Austin Melton, Introducing Lagois Correspondences, Categorical Structures and Their Applications, (Proceedings of the North-West European Category Seminar,) World Scientific Publishing Company, editors W. Gaehler and G. Preuss, Berlin, 2004, 207-217.
2. Beverly M. Reed, Austin Melton, and A. Bathi Kasturiarachi, Using History to Teach Functions within an APOS Approach to Learning, Delta:05 Proceedings, Fifth Southern Hemisphere Symposium on Undergraduate Mathematics Teaching, Australia, 2005, 110-116.
3. A. Bathi Kasturiarachi, Austin Melton, and Beverly M. Reed, Classroom-Relevant Advanced Mathematics for Middle School and High School Teachers, Proceedings Third International Conference on the Teaching of Mathematics at the Undergraduate Level, Turkey, 2006.

Nesterenko, Mik

1. M. Nesterenko, S. Tixeuil, "Discovering Topology in the Presence of Byzantine Faults", *IEEE Transactions on Parallel and Distributed Computing*, to appear;
2. P. Danturi, M. Nesterenko, S. Tixeuil, "Self-Stabilizing Philosophers with Generic Conflicts" *ACM Transactions on Autonomous and Adaptive Systems*, 4(1), 2009;
3. S. Dolev, M. Kopeetsky, T. Clouser, M. Nesterenko "Low Overhead RFID Cryptography", in *"RFID Handbook: Applications, Technology, Security and Privacy*", CRC Press, March 2008;

Peyravi, Hassan

1. Y Drabu and H. Peyravi. An adaptive bandwidth control algorithm for IP routers. In Proceedings International Conference on Communications in Computing (CIC), pages 311–317, Las Vegas, USA, June 2004.

2. M. K. Khan and H. Peyravi. Delay and jitter analysis of generalized demand-assignment multiple access (DAMA) protocols with general traffic. In HICSS '05: Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05) - Track 9, page 304.1, Washington, DC, USA, January 2005. IEEE Computer Society.
3. Y. Drabu and H. Peyravi. Fault tolerant routing in tri-sector wireless cellular mesh networks. In Proceedings International Conference Parallel and Distributed Computing Systems, pages 209–216, San Francisco, California, USA, September 2006.

Ruttan, Arden

1. N. Malhis, A. Ruttan, and H. H. Refai An Efficient Approach for Candidate Set Generation, Journal of Information & Knowledge Management, Vol. 4, No. 4 (2005) 287-291.
2. N. Malhis, and A. Ruttan, Predicting Gene Regulatory Networks from Micro Array Time Series Data by E limination, Proceedings of the Second Annual BioTechnology and BioInformatics Symposium BIOT -05, October 20-21, 2005, Colorado Spring, Colorado, USA, pp. 37-42.
3. N. Malhis, and A. Ruttan, Detecting Gene Regulation Relations from Microarray Time Series Data, Proceedings of the 2006 International Conference on Machine Learning; Models, Te chnologies & Applications MLMTA'06, June 26-29, 2006, Las Vegas, Nevada, USA, pp 122-147.

Volkert, Gwenn

1. Amin Assareh and L. Gwenn Volkert (2009) "Fusing Fuzzy Rule Base Classifiers for Improving Protein Mass Spectra Based Ovarian Cancer Diagnosis ", Proceedings of the IEEE 2009 Symposium on Computational Bioinformatics and Computational Biology, CIBCB'09 Nashville, TN, USA, March 30 - April 2, 2009, IEEE Press, Piscataway, NJ
2. Deborah A. Stoffer and L. Gwenn Volkert, "Evolving Chaos Automata for Fractal Based Visualizations of Protein Sequence Data", in proceedings of IEEE 2008 Symposium on Computational Bioinformatics and Computational Biology, Sept. 15-17, 2008 Sun Valley, Idaho, IEEE Press, Piscataway, NJ.
3. Amin Assareh, Mohammad Hassan Moradi, L. Gwenn Volkert (2008) "A Hybrid Random Subspace Classifier Fusioin Approach for Protein Mass Spectra Classification", 6th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics, Springer Lecture Notes in Computer Science Vol 4973.

Walker, Robert

1. "Using Hardware Multithreading to Overcome Broadcast/Reduction Latency in an Associative SIMD Processor", Kevin Schaffer and Robert A. Walker, Parallel Processing Letters, 18(4):491–509, December 2008.
2. "Multithreading in an Associative SIMD Processor", Kevin Schaffer and Robert A. Walker, in Proc. of the 22nd International Parallel and Distributed Processing Symposium (Workshop on Large-Scale Parallel Processing), full text on CDROM. Miami, Florida, April 2008.
3. "Dynamic Round-Robin Task Scheduling to Reduce Cache Misses for Embedded Systems", Ken W. Batcher and Robert A. Walker, in Proc. of the 2008 Design, Automation and Test in Europe Conference, pp. 260–263. Munich, Germany, March 2008.

Wang, Paul

1. Java with Object-Oriented and Generic Programming, E-book published by sofpower.com, ISBN 978-1-4276-3452-8, Sept. 2008
2. "DMAS: A Web-based Distributed Mathematics Assessment System" (with Saleh Al-shomrani) International Conference on Learning 2008, University of Illinois at Chicago, USA, June 3-6, 2008
3. "WME: a Web-based Mathematics Education System for Teaching and Learning." (with M. Mikusa, S. Al-shomrani, X. Lai, X. Zou, D. Zeller) ICME 11 TSG 22 Theme 3 the 11the International Congress on Mathematical Education, Monterrey, Mexico, July 6 - 13, 2008.

Zhao, Ye

1. Zhe Fan, Yu-chuan Kuo, Ye Zhao, Feng Qiu, Arie Kaufman, Bill Arcieri, Visual Simulation of Thermal Shock in a Pressurized Water Reactor, *The Visual Computer, International Journal of Computer Graphics*, Springer, To appear, 2009.
2. Ye Zhao, Lattice Boltzmann based PDE solver on the GPU, *The Visual Computer, International Journal of Computer Graphics*, volume 24, number 5, pages 323-333, May, 2008.
3. Ye Zhao, Yiping Han, Zhe Fan, Feng Qiu, Yuchuan Kuo, Arie Kaufman and Klaus Mueller, Visual Simulation of Heat Shimmering and Mirage, *IEEE Transactions on Visualization and Computer Graphics, Vol 13, No 1, pages 179-189*, January/February, 2007

Appendix C5
List the faculty honors, prize and awards received over the past five years. Exclude sponsored awards that primarily support research, scholarly or creative activities.

Breitbart, Yuri

Awards:

- Lucent Technology Gold award for participation in the design and development of NetInventory Product 2004
- Lady Davis Fellowship, Israel Technological Institute, 2007

Jin, Ruoming

- ACM KDD (Knowledge Discovery and Data Mining) student travel award, 2005

Volkert, Gwenn

Honors:

- IEEE Senior Member - March 2008

Walker, Robert

Award and Honors:

- October 2006, Recognized as a Distinguished Member of the Association for Computing Machinery (ACM) "in recognition of significant accomplishments in, and impact on, the computing field"
- June 2008, Received the Outstanding Contribution to ACM Award from the Association for Computing Machinery (ACM) "for a sustained record of dedicated and conscientious leadership within the ACM Special Interest Groups, including service as Chair of the SIG Governing Board, Chair of SIGDA, SGB Representative to Council, as well as leadership in ACM conference organization"
- June 2006, Received the Distinguished Service Award from the Association for Computing Machinery Special Interest Group on Design Automation (ACM / SIGDA) "for dedicated service as SIGDA Chair (2001-2005) and over a decade of service to SIGDA, DAC, and the EDA profession"

Appendix C6

Provide a representative list of professional service activities of the faculty over the past five years (i.e. Officer of professional society, program chair or organizer of professional meeting, service on boards, editorships of professional publication/journals, reviewer or consultant for federal, state, local or private agencies or institutions). Limit the listing to two pages.

Baker, Johnnie

1. 2009 Member of the Steering Committee for the Workshop on Large-Scale Parallel Processing, (OSPP) at the IEEE/ACM sponsored International Parallel and Distributed Processing Symposium (IPDPS), Rome, Italy, May 29, 2009.
2. 2008 Member of the Steering Committee for the Workshop on Large-Scale Parallel Processing, (OSPP) at the IEEE/ACM sponsored International Parallel and Distributed Processing Symposium (IPDPS), Miami FL, April 18, 2008.
3. 2007 Member of the original 2007 Organizing Committee for the Large Scale Parallel Processing (LSPP) Workshop that meets annually at the IEEE/ACM sponsored International Parallel and Distributed Symposium (IPDPS). This workshop was created to replace and broaden the scope of the earlier Workshop on Massively Parallel Processing.

Bansal, Arvind

1. 2009 PC member, International Conf. on Signal Processing and Multimedia Appl., Milan, Italy
2. 2009 PC member, International Conf. on E-business and Telecommunications, Milan, Italy
3. 2009 PC member, International Conf. on Distributed Multimedia Systems (DMS 2009)

Breitbart, Yuri

1. NSF Workshop On Wireless Infrastructure, WDAS2003, WDAS2004, ICDM2004
2. I2CS International Workshop 2006,
3. I2CS International Workshop 2007,

Dragan, Feodor

1. *WG 2009:* 35th International Workshop on Graph-Theoretic Concepts in Computer Science (WG 2009), June 24-26, 2009, Montpellier, France
2. *SAWN 2006*: 2nd ACIS International Workshop on Self-Assembling Wireless Networks, June 19 - 20, 2006, Excalibur Hotel, Las Vegas, Nevada, USA
3. *SAWN 2005*: 1st ACIS International Workshop on Self-Assembling Wireless Networks, May 24, 2005, Towson University, Maryland, USA

Farrell, Paul

1. Member of the Program Committee for 6th International Symposium on Parallel and Distributed Processing and Applications (ISPA-08, Sydney, Australia 2008
2. Member of the Program Committee for OCCBIO Ohio Collaborative Conference on Bioinformatics 2008
3. Member of the Program Committee for IADIS Computer Graphics and Visualization (CGV), Lisbon 2007

Guercio, Angela

1. Program Committee of the 15th International Conference on Distributed Multimedia Systems 2009 (DMS '09).
2. Publicity Chair, Ohio Celebration of Women in Computing (OCWIC 2009)
3. Session Organizer of the Track on Multimedia Software Engineering and member of the Program Committee of the 14th International Conference on Distributed Multimedia Systems 2008 (DMS '08).

Jin, Ruoming

1. Sponsorship Co-Chair for SIAM Conference on Data Mining (SDM), 2009.
2. Program Committee Member for ACM conference on knowledge discovery and data mining (KDD), 2009.
3. Program Committee Member for International Conference on Data Mining (ICDM), 2009

Khan, Javed

1. Served as proposal reviewer for NSF, DOE, NASA.
2. Served as a member of the program committees in various professional conferences including WIC 2008-2004, ICME 2006, ICCIT 2007-2005, IWAN 2005, PV 2006-2006, SAINT 2004, etc.

3. Organizing Committee of SAINT 2009, Workshop Chair

Lu, Cheng-Chang

1. Member of International Program Committee, IASTED International Conference on Telehealth, 2005-Present.
2. Member of Program Committee, IADIS Virtual Multi Conference on Computer Science and Information Systems, 2006.
3. Session Chair, Medical Image and Processing, 2007 IASTED International Conference on Signal and Image Processing.

Maletic, Jonathan

1. Program Co-Chair IEEE 14th Working Conference on Reverse Engineering (WCRE) 2007 Vancouver, B.C., Canada, October 29-31
2. General Chair IEEE 13th International Workshop on Program Comprehension (IWPC) 2005 St. Louis, Missouri, May 15-16
3. Steering Committee IEEE International Conference on Program Comprehension (ICPC) (Chair '07-'10) 5/2005-current

Melton, Austin

1. Chair, Statewide Users Group for the Ohio Supercomputer Center, 2003 - 2005.
2. Ex officio member of the Ohio Supercomputer Center Governing Board, 2003 - 2005.
3. Co-chair, Computer Science Judges, Intel International Science and Engineering Fair, Cleveland, 2003.

Nesterenko, Mik

1. 11th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS) 2009;
2. 8th International Workshop on Assurance in Distributed Systems and Networks (ADSN) 2009;
3. 10th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS) 2008;

Peyravi, Hassan

1. Member of College Advisory Committee, 2006-2007.
2. Coordinator of Graduate Studies Committee, 2006-2008.
3. Coordinator of Preliminay Examination Committee, 2006-2008.

Rothstein, Michael

1. 2006-2008 Alternate member, university-wide transportation committee
2. 2008-2010 Alternate Member University-wide Requirements Curriculum Committee
3. 2008-2009 Member and Chair, University-wide transportation committee

Volkert, Gwenn

1. Associate Editor BioSystems appointed January 2009
2. Associate Editor IEEE Computational Intelligence Magazine appointed in 2006
3. General Chair: 2010 *IEEE Symposium on Computational Intelligence for Bioinformatics and Computational Biology (CIBCB'2010)* May. Quebec, Canada

Walker, Robert

1. July 2006 – present Member, ACM Council (as SGB Council Representative)
2. January 2006 – present Chair, ACM Senior Member Committee
3. January 2006 – present Member, ACM Distinguished Member Committee

Wang, Paul

1. Invited keynote speaker, East Cost Computer Algebra Day, Shepherd University, Shepherdstown, West Virginia, Saturday, May 10, 2008.
2. WME Tutorial Workshop by P. Wang and the WME Team Shepherd University, Shepherdstown, West Virginia, May 9, 2008.
3. Improvements and new implementation for MathPASS, a Web-based drill-and-practice system for remedial mathematics, by P. Wang, Su Wei and Dean Zeller in cooperation with the Department of Mathematical Sciences, started Spring 2008.

Zhao, Ye

1. NSF Proposal Panel, Panelist June, 2009
2. IEEE International Conference on Shape Modeling and Application, Program Committee Member 2009
3. Center for Material Informatics, Kent State, Fellow 2009

Section C7:

Five year Biographical Sketch – for F2009 Dept Review – 8/31/04-8/31/09

Johnnie W. Baker

Professor of Computer Science

EDUCATIONAL BACKGROUND:

Hardin-Simmons University (Texas)	Mathematics	B.A.	1958
University of Texas at Austin	Mathematics	M.S.	1965
University of Texas at Austin	Mathematics	Ph.D.	1968

BRIEF EMPLOYMENT HISTORY:

2001-	Professor of Computer Science, Kent State University
2001-2004	Professor and Chair, Department of Computer Science, Kent State University
1995- 2001	Professor of Computer Science, Department of Mathematics and Computer Science, Kent State University
1983-84	Visiting Associate Professor of Computer Science, University of Texas at Austin
1975-95	Associate Professors of Mathematics and Computer Science, Kent State University
1973-75	Assistant Professor of Mathematics, Kent State University
1968-73	Assistant Professor of Mathematics, Florida State University

REPRESENTATIVE PUBLICATIONS: (Some papers posted at http://www.cs.kent.edu/~parallel/)

1. Mike Yuan, Johnnie Baker, Frank Drews, and Will Meilander, "A*n Efficient Implementation of Air Traffic Control using the ClearSpeed CSX620 System"*,Parallel and Distributed Computing Systems (PDCS 2009), Cambridge, 8 pages, to appear in November 2009.
2. Jerry Trahan, Mingxian Jin, Wittaya Chantamas, and Johnnie Baker, "*Relating the Power of the Multiple Associative Computing Model (MASC) to that of Reconfigurable Bus-Based Models*", Journal of Parallel & Distributed Computing (JPDC), 9 journal pages, Accepted August 15, 2009, Available online at DOI link http://dx.doi.org/10.1016/j.jpdc.2009.08.001.
3. Steinfadt, S and J. W. Baker , "*SWAMP: Smith-Waterman using Associative Massive Parallelism*", IEEE *Workshop on Parallel and Distributed Scientific and Engineering Computing*, 2008 International Parallel and Distributed Processing Symposium (IPDPS) at Miami, 8 pages, published on CD, April 14-18, 2008.
4. Steinfadt, S., Scherger, M, Baker, J , "*A Local Sequence Alignment Algorithm Using an Associative Model of Parallel Computation*", Proc. of IASTED Computational and Systems Biology (CASB 2006), Dallas, pp. 38-43, Nov. 13-14, 2006.
5. Weiguo Fan, Xin Lin, Yu-Wei Hsieh, Boren Lin, Paul Durand, Johnnie Baker and Chun-che Tsai, , "*Chemical molecular similarity analysis and its applications in structure-activity visualization*", The Life Science Society (LSS) Computational Systems Bioinformatics Conference, Stanford University, California, 6 pages in proceeding CD, August 14-18, 2006.
6. Wittaya Chantamas, Johnnie Baker, and Michael Scherger, "*An Extension of the ASC Language Compiler to Support Multiple Instruction Streams in the MASC Model using the Manager-Worker Paradigm,*", Proc. of the 2006 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2006), 7 pages in proceedings CD, June 2006.
7. Wittaya Chantamas, Johnnie Baker, and Michael Scherger, "*Compiler Extension of the ASC Language to Support Multiple Instruction Streams in the MASC Model using the Manager-Worker Paradigm*", 2006 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'06), Las Vegas, 7 pages, June 26-29, 2006.

8. Stewart Reddaway, Will Meilander, Johnnie Baker, and Justin Kidman, "*Overview of Air Traffic Control Using a SIMD COTS System*", *International Parallel and Distributed Processing Symposium*, 9 pages, Published on CD, April 2005.
9. Wittaya Chantamas and Johnnie Baker, "*A Multiple Associative Model to Support Branches in Data Parallel Applications using the Manager-Worker Paradigm*", Proc. of the 19th International Parallel and Distributed Processing Symposium (IEEE WMPP Workshop), 8 pages, April 2005.
10. Weiguo Fan, Xin Lin, Yu-Wei Hsieh, Boren Lin, Johnnie W. Baker and Chun-che Tsai, "*Chemical Structure-Activity Relationship Visualization Using Structure-Activity Maps*", 2005 IEEE Computational Systems Bioinformatics Conference, Stanford University, California, USA, 2 pages, August 8 - 11, 2005.

UNDERGRADUATE STUDENTS MENTORED

- Research advisor for Tristan Cuevas, an undergraduate honors student, in the STARS undergraduate research program. 2007-2008

MASTERS AND DISSERTATION STUDENTS SUPERVISED

- **Dissertation Students Supervised:** Michael Scherger (completed), Rashid Muhammad, Wittaya Chantamas, Weiguo Fan, Shannon Steinfadt, and Mike Yuan.
- **Master Thesis Students Supervised:** Xin Lin (completed), Jeffrey Daniel Frey (completed), Justin Kidman, Pallav Laskar, Venkata Marella, Sagar Panchariya

LIST OF DEPARTMENT OR COLLEGE COMMITTEES AND ROLE

- **AY 2009:** CAC dept representative, member of the OBR Budget Committee and Colloquium Committee. Chaired the organization of the Chair Review and served as an elected member,
- **AY 2008:** CAC dept representative, colloquium committee, and handbook revision committee. I continued to served in Spring while on Sabbatical,
- **AY 2007:** Member of the OBR Budget Committee, the Colloquium Committee, and the Handbook Revision Committee.
- **AY 2006:** Chair of the OBR Budget Committee and member of the Colloquium Committee
- **AY 2005:** I am fairly sure that I was chair of the OBR Budget Committee during this time – but failed to include this in my vita. My term as dept chair had just ended.

PROFESSIONAL SERVICE ACTIVITES

- **AY 2009:** Member of Steering Committee for the Large Scale Parallel Processing Workshop at the IEEE IPDPS conference, outside reviewer for Dr. Frank Drews' promotion to associate professor at OU, member of the program committee for both the PDCN and PDCS IASTED conferences.
- **AY 2008:** Member of the Steering Committee for the Large Scale Parallel Processing Workshop at the IEEE sponsored IPDPS International Conference, member of the program committee for both the PDCN and the PDCS IASTED international conferences
- **AY 2007:** External reviewer for Dr. Victor Matos promotion to full professor at CSU College of Business Adm., Member of both the original organizing committee and the Steering Committee for the Workshop on Large Scale Parallel Processing (LSPP) at the IEEE sponsored IPDPS'07 conference, Member of program committee for both the PDCS and PDCN ISConferences.
- **AY 2006:** Editor for Parallel Processing Letters Journal (PPL), outside reviewer for Dr. Julie Barnes promotion to full professor at BGU, co-chair of special session at APDCM Workshop at the IEEE IPDPS Conference. Program Committee for PDCS & PDCN
- **AY 2005:** Editor PPL Journal, Organizing and Steering Committee for Massively Parallel Workshop in IEEE sponsored IPDPS'05 conference, Program committee for PDCS & PDCN conferences.

Arvind K. Bansal, Ph.D., Full Professor

Education

1988 Ph. D. Computer Science, Case Western Reserve University, Cleveland, OH, USA.
1983 M. Tech. Computer Science, Indian Institute of Technology, Kanpur, UP, India.
1979 B. Tech. Electrical Engineering, Indian Institute of Technology, Kanpur, UP, India.

Appointments (post PhD)

2005 -	Full Professor, Computer Science, Kent State University, Kent, Ohio, USA
2004	Visiting Professor, Benares Hindu University, Benares, India (sabbatical)
2004	Visiting Professor, Indian Institute of Technology, Kanpur, (sabbatical)
1993-2005	Assoc. Professor, Computer Science, Kent State University, Kent, Ohio, USA
1996	Senior Research Fellow, University of Melbourne, Australia (sabbatical)
1995	Visiting Professor, EMBL, Heidelberg, Germany (sabbatical)
1993	Summer Faculty, Argonne National Laboratory, Illinois, USA
1988-1993	Assistant Professor, Computer Science, Kent State University, Kent, OH, USA

Representative Refereed Publications (Total 15 since August 2004)

1. A. Bansal, "Role of Bioinformatics in the Development of Anti Infective Therapy," *Expert Review of Anti-Infective Therapy*, Vol. 6, Issue 1, 2008, pp. 51-63
2. A. Guercio, A. K. Bansal, T. Arndt, "Language Constructs and Synchronization in Reactive Multimedia Systems," *International Transaction on Computers and Software Engineering*, No. 1, Vol. 1, 2007, ISSN: 1797-1152, pp. 52-59
3. A. Guercio and A. K. Bansal, "Towards a Formal Semantics for Distributed Multimedia Computing," *International Conf. on Distributed Multimedia Systems*, San Francisco, USA, 2007, pp. 81-86
4. A. K. Bansal and Y. Hijazi, "Low Bandwidth Video Conversation Using Anatomical Reconstruction of Facial Expressions Over the Internet", *IADIS International Conf. on WWW/Internet* , October 2006, Murcia, Spain, pp. 154-161
5. A. K. Bansal, "Incorporating Fault Tolerance in Distributed Agent Based Systems by Simulating Biocomputing Model of Stress Pathways," SPIE Defense and Security Symposium, April 2006, Orlando, Florida, pp. Vol. 6201, pp. 620108-01 - 62010810
6. A. K. Bansal, Bioinformatics in microbial biotechnology - a mini review, Microbial cell Factories, volume 4, issue 19, 2005, **Highly accessed article, over 10,000 downloads (including Biomed Central and Pubmed Central), Known citations: 20.**
7. T. E. Meyer and A. K. Bansal, "Elevated CG Content in Hyperthermophile can Resolve Evolutionary Discrepancies between Analysis using Whole Genome Comparisons and 16SrRNA," Biochemistry, **2005;** *44*(34), pp 11458 – 11465
8. A. Guercio and A. K. Bansal et. el, "Synchronization in Distributed Multimedia Systems," Proc. of the Internat. Conf. of Distributed Multimedia Systems, Banff, Canada, 2005, pp. 34-39
9. A. K. Bansal, "Exploiting Biological Model to Incorporate Event Based Adaptation in Networked Intelligent Multi-Agent Systems**,** Proc. of the 'IEEE International Conf. on Tools with Artificial Intelligence 2004, ' Boca Raton, USA, Nov. 2004, pp. 761-768
10. B. Simoes, A. Guercio, and A. K. Bansal, "Towards Large Scale Voice Activated Dynamic and Interactive Internet Based Animation and Modeling," Proc. of the 'IASTED International Conf. on Software Engineering and Applications', Cambridge, USA, Nov. 2004, pp. 749-754

Representative Presentations/Invited Talks

1. Towards a Formal Semantics for Distributed Multimedia Computing," *International Conference on Distributed Multimedia Systems*, San Francisco, USA, 2007
2. **L**ow Bandwidth Video Conversation Using Anatomical Reconstruction of Facial Expressions

Over the Internet", *IADIS International Conf. on WWW/Internet*, Murcia, Spain, 2006
3. "Incorporating Fault Tolerance in Distributed Agent Based Systems by Simulating Biocomputing Model of Stress Pathways," SPIE Defense and Security Symp., Orlando, 2006
4. Advanced seminar series in Bioinformatics, Benares Hindu University, India, Fall 2004

Funded Grants

Research Council Grant, Kent State University, US$ 6, 500, Project: Genome Comparison for Anti infective Therapy, Summer 2008
Wright Patterson Base-Air Force, US$69, 000, Project: Fault Tolerant Adaptive Multimedia Agent Based Systems, One year, June 2004-May 2005

Graduate Thesis Supervision

A. Guercio, PhD Dissertation, (Fall 2004), Coauthored publications: #2, #3, #8, #11, #15
J. Burke, MS thesis, (Spring 2008)
Y. Hijazi, MS Thesis, (Spring 2005), Coauthored publications: #4
S. Uddin, MS Thesis, (Spring 2004), Coauthored publication: #12

Committee Membership

Curr. Xun Lai (CS); Xiao Zhu (CS); Manas Hardas (CS)
2008 Saleh Al Shomrani, Computer Science, Kent State University
2007 Obyebisi Jegede, Biological Sciences, External Committee Member, Kent State University
2005 External Thesis Examiner, D. Phil Student, University of Allahabad, India

Professional Service

Editorial Board: International Journal for Tools with Artificial Intelligence

2009 PC member, International Conf. on Signal Processing and Multimedia Appl., Milan, Italy
2009 PC member, International Conf. on E-business and Telecommunications, Milan, Italy
2009 PC member, International Conf. on Distributed Multimedia Systems (DMS 2009)
2008 PC member, International Conf. on Distributed Multimedia Systems (DMS2008)
2008 PC member, IEEE International Conf. on Computational Intelligence and Security
2008 PC member, IASTED International Conf. on Distributed and Intelligent Multimedia Systems
2007 PC Member, IADIS International Conf. WWW/Internet, Lisbon, Portugal
2006 PC Member, International Conf. on Distributed Multimedia Systems, (DMS 2006)
2006 PC Member, International Conf. Bioinformatics and Bioengineering (BIBE 2006)
2005 PC Member, International Conf. on Distributed Multimedia Systems (DMS2005), Canada
2005 PC Member, Second World Congress on Lateral Computing (WCLC 2005), Bangalore, India
2005 PC Member, IEEE Symp. on Bioinformatics and Biomedical Engineering, Washington DC

Journal Reviewer: Biotechnology Progress, Microbial Cell Factories, Biomed Central, IEEE Multimedia, IEEE Software, IEEE Transactions on Knowledge and Data Engineering

International Conference Reviewer (multiple years): (1) IADIS International Conference WWW/Internet; (2) IASTEAD International Conference on Communications, Internet, & Information Technology; (3) IASTED International Conference on Distributed and Intelligent Multimedia Systems; (4) International Conference on Distributed Multimedia Systems; (5) International Conference on Computational Intelligence and Security; (6) International Conference on Signal Processing and Multimedia Application

Federal Funding Agencies

1. Reviewer, National Science Foundation, USA
2. Panelist, National Institute of Health – Bioinformatics and Bioengineering Division, USA
3. Independent expert, European Commission Research Directorate General, Belgium

Yuri Breitbart – OBR Distinguished Professor of Computer Science

Education

- D.Sc. in Computer Science –Computer Science Department Israel Technological Institute(TECHNION), Haifa, Israel.
- MS in Mathematics -Mathematics Department, Moscow Pedagogical Institute, Moscow, Russia.

Professional Experience (last 5 Years)

- 2002 -current: Ohio Board of Regents Distinguished Professor of Computer Science, Kent State University, Kent, OH 44240

Other Academic Appointments

- 2007-2008: Lady Davis Visiting Professor, Department of Computer Science, Israel Technological Institute (TECHNION), Haifa, Israel.
- 2007: Senior Researcher, University of Basel, Switzerland.

Awards

- Lucent Technology Gold award for participation in the design and development of NetInventory Product 2004
- Lady Davis Fellowship, Israel Tecnological Institute, 2007. (External Support)
- European Union Grant to work at University of Basel

Memberships: Member of ACM (Fellow), SIGMOD, and IEEE Computer Society

Publications

- Y. Breitbart, M. Garofalakis, Ben Jai, R. Rastogi, A. Silberschatz Topology Discovery in Heterogeneous IP Networks: The NetInventory System IEEE/ACM Transaction on Networking, 12, 3, 2004
- Y. Bejerano, Y. Breitbart, R. Rastogi, A. Orda, A. Sprintson Algorithms for Computing QoS path with Restoration, IEEE/ACM Transaction on Networking, 13,3, June 2005
- Gurm, Hitinder; Breitbart, Yuri; Vivekanathan, Deepak; Yeh, Michael; Fathi, Robert; Ziada, Khaled; Whitlow, Patrick, Ellis, Stephen Preprocedural statin use is associated with a reduced hazard of post-procedural myonecrosis in patients undergoing rotational atherectomy -a propensity adjusted analysis. American Heart Journal, 151(5):pp 1-6, 2006
- YuriBreitbart and Hassan Gobjuka, Characterization of Layer 2 Unique Topologies,Information Proccessing Letters, 2008.
- Ruoming Jin, Yuri Breitbart and Chibuike Muoh, Data Discretization Unification, Knowledge and Information Systems International Journal,(invited paper),SpringerVerlag,29pages, vol. 19,No. 1,April
- Yuri Breitbart, Feodor Dragan, Hassan Gobjuka Effective Monitor Placement in Internet Networks, Journal Of Networks,(to appear)
- Y.Breitbart,F.Dragan,H.Gobjuka”Effective Network Monitoring”, Proceedings of the 13th International Conference on Computer Communications and Networks (ICCCN’04),Chicago, October 2004.
- Fuat Akal, Yuri Breitbart, Torsten Grabs, Hans-Jorg Schek, Can Turker, Lourens Veen ”Fine-Grained Replication, and Scheduling with Freshness and Correctness Guarantees”,Proceedings of the 31th International Conference on VLDB, Trondheim, Norway, August 2005

- Yuri Breitbart, Minos Garofalakis, Anupam Gupta, Amit Kumar, Rajeev Rastogi ”On Configuring BGP Route Reflectors” The Second International Conference on Communication Systems SoftWAre and MiddlewaRE, Jan 2007, Bangalore, India
- YuriBreitbart,HassanGobjuka ”Ethernet Topology for Networks with Incomplete Information” Proceedings of the International Conference on Computer Communication(ICCCN),2007,Honolulu,Hawaii
- Hassan Gobjuka,Yuri Breitbart, ”Discovering Network Topology of large Multisubnet Ethernet Networks”, Proceedings of the 32d Annual IEEE Conference on Local Computer Networks, 2007, Dublin, Ireland.
- Ruoming Jin, Yuri Breitbart, Chibuke Muoh, ”Data Discretization Unification”, Proceedings of International Conference on Data Mining, 2007, Omaha NE.
- Ruoming Jin, Scott McCallen, Yuri Breitbart, David Fuhry, Dong Wang ”Estimating the Number of Frequent Item sets in a Large Database”, Proceedings of EDBT 2009, Saint Petersburg, Russia.
- L. Voicu, Fuat Akal, Y. Breitbart, H. Schuldt, and H.-J. Schek ”Re:GRIDiT -Coordinated Distributed Update Transactions on Replicated Data in the Grid”, Proceedings of GRID 2009 Conference, Banff, Canada 2009
- Ruoming Jin, Yuri Breitbart, and Rong Li ”A Tree-based Framework for Differences Summarization” Proceedings of ICDM 2009 Conference, Miami, Florida 2009

Reviewer for:

CACM; Data and Knowledge Engineering; Transactions on Knowledge Engineering; BIT; Transactions of ACM on Database Systems; Computing Surveys; The VLDB Journal; Distributed and Parallel Databases International Journal; SIAM Journal on Computing, Information Systems; Software - Practice and Experience; IEEE, Transaction On Networking, IEEE Communication Newsletter.

Committees:

WDAS2004, ICDM2004, I2CS International Workshop 2006, I2CS International Workshop 2007, VLDB 2006, PODS 2007, CAC – 2003 FAC, Curriculum Committee, OBR Committee.

Master and PhD Students supervised

Deepakraj Shanthilal 9/29/2005
Vibha Tripathi 1/25/2005
Olena Andriyevska 3/24/2006
Hassan Gobjuka 5/17/2007
Rung Lee - current

Master and PhD Committee Member

Dong Wang 11/13/2008
Anh Tran 5/8/2006
Nitin Bafna 4/30/2007
David Fuhry 4/14/2008
Xiaoxi Du 4/3/2009
Huzefa Kagdi 6/30/2008
Siddarth Raina 11/1/2006
Fuat Akal 2005 (ETH, Zuerich)

Biographical Sketch: Dr. Feodor F. Dragan, Associate Professor

Professional Preparation

3/90 Ph.D. in Computer Science, Institute of Mathematics of the Belorussian Academy of Sciences, Minsk, Belarus
6/85 M.S. (with honors) in Applied Mathematics, Moldova State University, Dept. of Mathematics and Cybernetics, Kishinev, Moldova
5/83 B.A. (with honors) in Applied Mathematics, Moldova State University, Dept. of Mathematics and Cybernetics, Kishinev, Moldova

Appointments

08/04 - Associate professor, Department of Computer Science, Kent State University
08/00 - 07/04 Assistant professor, Department of Computer Science, Kent State University
10/99 - 08/00 Research associate, University of California at Los Angeles, Computer Science Department, UCLA VLSI CAD Lab
10/96 - 09/99 Research associate, University of Rostock, Germany: Research supported by the German Research Community (DFG) and by the Volkswagen Foundation (VW)
03/94 - 12/95 Research associate, University of Duisburg, Germany: Research supported by the Volkswagen Foundation (VW) and DAAD Research Fellowship
07/95 - 09/99 Associate professor, Dept. of Mathematics and Cybernetics, Moldova State University
09/89 - 06/95 Assistant professor, Dept. of Mathematics and Cybernetics, Moldova State University

Recent Visiting Positions

05/26/07 - 06/20/07 Algorithms Research Group, Dep. of Informatics, U. of Bergen, Norway
06/25/07 - 09/30/07 Combinat. Optimiz. & Graph Algorithms Group at TU Berlin, Germany
10/01/07 - 12/20/07 Res. Group on Algorithms and Combinatorics at University Paris 7
05/10/08 - 07/10/08 Discrete Maths and Theor. Comput. Science, University of Chile, Santiago
06/01/09 - 06/30/09 Laborat. d'Informatique Fondamentale de Marseille, U. de la Mediterranée, France

List of representative publications.

Altogether, from the end of August 2004 until the end of August 2009, I published **13 journal articles** and **18 refereed conference articles**.

1. C. Yan, Y. Xiang and F.F. Dragan, Compact and Low Delay Routing Labeling Scheme for Unit Disk Graphs, WADS 2009, LNCS 5664, pp. 566-577.
2. F.F. Dragan, F. Fomin and P. Golovach, Spanners in sparse graphs, ICALP 2008, LNCS 5125 (Part I), pp. 597–608.
3. V.D. Chepoi, F.F. Dragan, B. Estellon, M. Habib and Y. Vaxès, Diameters, centers, and approximating trees of delta-hyperbolic geodesic spaces and graphs, SoCG 2008, pp. 59–68.
4. A. Brandstädt, Dragan F.F., H.-O. Le, V.B. Le and R. Uehara, Tree spanners for bipartite graphs and probe interval graphs, Algorithmica 47 (2007), 27-51.
5. Dragan F.F. and I. Lomonosov, On compact and efficient routing in certain graph classes, Discrete Applied Mathematics 155 (2007), 1458-1470.
6. Y. Dourisboure, Dragan F.F., C. Gavoille, and C. Yan, Spanners for bounded tree-length graphs, Theoretical. Computer Science 383 (2007), 34-44.
7. F.F. Dragan, C. Yan and D.G. Corneil, Collective Tree Spanners and Routing in AT-free Related Graphs, Journal of Graph Algorithms and Applications, Vol. 10, no. 2, 2006, 97-122.
8. V.D. Chepoi, F.F. Dragan, Y. Vaxès, Distance and routing labeling schemes for non-positively curved plane graphs, Journal of Algorithms 61 (2006), 60-88. (one of the TOP25 Hottest Articles within the Journal of Algorithms).
9. V.D. Chepoi, F.F. Dragan, Y. Vaxès, Addressing, distances and routing in triangular systems with applications in cellular networks, Wireless Networks 12 (2006), 671-679. The special issue "Best papers of the WMAN 2004 conference".
10. F.F. Dragan, C. Yan and I. Lomonosov, Collective tree spanners of graphs, SIAM J. Discrete Math. 20 (2006), 241-260.

11. V.D. Chepoi, Dragan F.F., and Chenyu Yan, Additive Sparse Spanners for Graphs with Bounded Length of Largest Induced Cycle, Theoretical Computer Science 347 (2005), 54-75.
12. Dragan F.F., Estimating All Pairs Shortest Paths in Restricted Graph Families: A Unified Approach, Journal of Algorithms 57 (2005), 1-21. (one of the TOP25 Hottest Articles within the Journal of Algorithms).
13. A. Brandstädt, Dragan F.F., H-O. Le, R. Mosca, New Graph Classes of Bounded Clique Width, Theory of Computing Systems 38 (2005), 623-645.

List of representative presentations. Altogether, from the end of August 2004 until the end of August 2009, I gave **14 invited conference & colloquia talks** and **16 contributed conference talks**. Here I list only some invited conference & colloquia talks.

06/09 Laboratoire d'Informatique Fondamentale de Marseille, U. de la Mediterranée, Marseille, France
03/09 Computer Science Department, University of Rostock, Germany (invited by Prof. Andreas Brandstädt)
05/09 2nd Canadian Discrete and Algorithmic Mathematics Conference, Montréal, Canada
03/09 "Algorithmic Graph Theory"- AGT 2009 conference, University of Warwick, UK
06/08 Discrete Maths and Theor. Computer Science Group (DMTCS), University of Chile, Santiago, Chile
10/07 Laboratoire d'Informatique Fondamentale de Marseille, U. de la Mediterranée, Marseille, France
10/07 Research Group on Algorithms and Combinatorics at University Paris Diderot - Paris 7
09/07 Combinatorial Optimization & Graph Algorithms Group at the TU Berlin, Germany
07/07 Combinatorial Optimization & Graph Algorithms Group at the TU Berlin, Germany
06/07 Algorithms Research Group, Department of Informatics, U. of Bergen, Norway
05/07 Dagstuhl-Seminar on "Exact, Approximate, Robust and Certifying Algorithms on Particular Graph Classes", Dagstuhl, Germany
06/06 SIAM Conference on Discrete Mathematics, Victoria, BC, Canada
12/04 RUTCOR Colloquia - Rutgers University, Piscataway, NJ

Honors, prizes, and awards

- 2 papers made a number of times the lists **TOP25 Hottest Articles** within the Journal of Algorithms
- 2 conference papers were included in the lists of **"Best papers of ..."**
- 1 conference paper was nominated for the **Best Paper Award**

Students graduated

- **Two PhD students**: Irina Lomonosov (2005), Chenyu Yan (2007).
- **Twelve MS students**: Sudha Elavarti (2005), Mutasem Najdawi (2005), Amit Borwankar (2005), George Powell (2005), Tran Anh Tuan (2006), Raina Siddharth (2006), Rajesh Jadhav (2007), Bafna, Nitin (2007), Viyyure, Udaykiran V. (2008), Bhaduri, Sudipta (2008), Rahul Sehgal (2009), Rab Harbart (2009).
- **Two MA students (with research projects)**: Dwarakanath Raghunathan (2005), Pankaj, Amitabh (2007).

Synergetic Activities

- Referee for Journal of Algorithms, Computational Geometry: Theory and Applications, IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems, Wireless Networks, SIAM Journal on Computing, SIAM Journal on Discrete Mathematics, Discrete Mathematics, Discrete Applied Mathematics, Journal of Graph Algorithms and Applications, Ars Combinatorica, etc.
- Reviewer for Computing Reviews.
- External reviewer for WG 2005, WG 2006, WG 2007, WG 2009, DISC 2005, DISC 2006, SODA 2006, SODA 2007, SODA 2008, ISAAC 2006, ICALP 2009.
- Program committee member of WG 2009, SAWN 2006, SAWN 2005.
- At Kent State University, Committee Member of Graduate Study Committee (2004 - 2006, 2008 - till now), Curriculum Committee (2005 - 2007), Subcommittee of the Curriculum Committee (2008), Chair Evaluation Committee (2005), Merit Committee (2008), Qualifying Exam Committee (2003 - 2005), Prelim. Exam Committee (2006 - 2007, 2008 - till now).
- At Kent State University, chaired Curriculum Committee (2006/2007) and Colloquium Committee (2003 - 2005, 2008 - till now).

Paul A. Farrell
Professor, Department of Computer Science, Kent State University, Kent, OH 44242.

EDUCATION

Trinity College Dublin	Mathematics	B.A.(Mod.)	1974
Trinity College Dublin	Experimental Physics	B.A.(Mod.)	1974
Trinity College Dublin	Numerical Analysis and Computing	M.Sc.	1978
Trinity College Dublin	Numerical Analysis and Computing	Ph.D.	1983

WORK EXPERIENCE

1998-99	Professor, Mathematics and Computer Science, Kent State University
1991-97	Associate Professor, Mathematics and Computer Science, Kent State University
1986-91	Assistant Professor, Mathematical Sciences, Kent State University
1985-86	Visiting Assistant Professor, Mathematical Sciences, Kent State University
1983-85	Lecturer in Mathematics and Computing, Dublin Institute of Technology
1981-83	Lecturer in Mathematics and Data Processing, Dundalk Institute of Technology
1976-78	Wetenschappelijk Medewerker, (Scientific Assistant), Mathematisch Instituut, Katholieke Universiteit, Nijmegan, The Netherlands

REPRESENTATIVE PUBLICATIONS:

- P.A. Farrell, E. O'Riordan, G.I. Shishkin, *Robust numerical methods for singularly perturbed semilinear differential equations with interior layers*, Proceedings of BAIL 2004, Toulouse, France, 6 pages, (2004)
- P.A.Farrell, A.F.Hegarty, J.J.H. Miller, E. O'Riordan and G.I. Shishkin, *Global maximum norm parameter-uniform numerical method for a singularly perturbed convection-diffusion problem with discontinuous convection coefficient*, Mathematical and Computer Modelling, 40, pp. 1375-1392, (2004)
- P.A. Farrell, E. O'Riordan, G.I. Shishkin, *A class of singularly perturbed semilinear differential equations with interior layers*, Math. Comp., 74, 2005, 1759-1776
- Dan Bennett, Paul A. Farrell, *Experiences Instrumenting a Distributed Molecular Dynamics Program* , Proceedings of PACISE'O6, pp 45 -- 50 (2006)
- Cara Stein, Daniel Bennett, Paul A. Farrell, Arden Ruttan, *A Steering and Visualization Toolkit for Distributed Applications*, PDPTA'06, pp 451--457, (2006)
- Dan Bennett, Paul A. Farrell, Cara Stein, *A Chromium Based Viewer for CUMULVS* , PDPTA'06, pp 472--477, (2006)
- Paul A. Farrell, Hong Ong, Arden Ruttan, Yang-Ming Zhu, *A Visualization Environment for Nematic Liquid Crystal Materials* , Proceeding of IADIS Multi Conference on Computer Science and Information Systems, Lisbon, Part III - Computer Graphics and Visualization, pp. 83--91, 2007
- P.A. Farrell, E. O'Riordan, *Examination of the performance of robust numerical methods for singularly perturbed quasilinear problems with interior layers*, BAIL 2008, Limerick, 2 pages (extended abstract) (2008).
- P.A. Farrell, E. O'Riordan, G.I. Shishkin, *A class of singularly perturbed quasilinear differential equations with interior layers* , Math. Comp. 78 (2009), 103-127.
- P.A. Farrell, E. O'Riordan, *Examination of the performance of robust numerical methods for singularly perturbed quasilinear problems with interior layers*., Lecture Notes in Computational Science and Engineering, 69 (2009), 141-151.
- Dan Bennett, Paul Farrell, and Arden Ruttan, *An Extensible Parameter Viewer for CUMULVS* , Proceedings of PACISE 20O9, 10--15, (2009)

STUDENTS SUPERVISED:

- Harit Desai, MS, Graduated Spring 2007
- Joseph Melnyk, Darren Brust, Deepakraj Shanthilal, Research
- Kiran Simon, Doug Stanley,Jeremiah Schilens, MS
- Dan Bennett, PhD

COMMITTEES:

International, National, State or Professional Service :

- Member of IMACS Technical Committee on Partial Differential Equations 1992-
- Member of Ohio Supercomputer Center, Statewide Users Group 1993-
- Director of Electronic Publishing for the Institute of Numerical Computation and Analysis (INCA) 1995-
- Member of Organizing Committees for RSV80
- Member, Steering Committee, Ohio Biomedical Science Research Conf., 2004
- Kent Liasion, WCI Visualization Initiative 2004-
- Member of the Program Committees for 6th International Symposium on Parallel and Distributed Processing and Applications (ISPA-08), Sydney, Australia 2008; OCCBIO Ohio Collaborative Conference on Bioinformatics 2008; IADIS Computer Graphics and Visualization (CGV), Lisbon 2007; Workshop on Grid Education, at CCGrid 2005, Cardiff, United Kingdom ; First International workshop on Operating Systems, Programming Environments and Management Tools for high-performance Computing on Clusters (COSET-1); COSET-2

University and Departmental Committee Chairs and Positions of Responsibility:

- Vice-Chair, Faculty Senate, 2007-08, 2008-09, 2009-10
- Chair, Faculty Senate Committee on Committees, 2007-08, 2008-09, 2009-10
- Chair, Faculty Senate Budget Committee 2007-2008
- Chair, University Council on Technology (UCT - created 1996), 1996-2000, 2005-2006
- Co-Chair, UCT Research Computing Working Group 2004-2007
- Systems Coordinator, Department of Computer Science,2008-10
- Assistant Chair, Department of Computer Science, 2005-2008
- Co-Chair, First Computer Science Alumni Reunion steering committee 2008
- Chair, Computer Science Handbook Review Committee, 2005-08
- Chair, Computer Science Chair Search Committee, 2006

University and Inter-collegial Service

- Faculty Senate (at large) 1999-2005 (A&S) 2007-2012
- University Council on Technology 1996-2001, 2004-2010
- Institute for Complex Adaptive Matter (ICAM) Steering Committee 2003--
- Center for Materials Informatics Executive Committee 2009--
- Educational Policies Committee (EPC) - Graduate division 2008-2011
- Professional Standards Committee 2009-2011
- Faculty Senate Budget Advisory Committee 2007-2010
- Faculty Senate Committee on Administrative Officers 2007-10
- Faculty Senate Committee on Committees 2005-2010
- Provost's Ad-Hoc Committee on Faculty Professional Improvement Leaves 2009
- Classroom Enhancement Committee 2008-9
- Faculty Senate Budget Committee 2006-2008
- Strategic Planning Steering Committee, 2007-08
- RCM Working Group, 2007-08
- Provost's Committee on Undergraduate Education in the 21st Century - First Year Experience Stakeholder Group, 2007-08
- University Council on Technology Security Subcommittee 2002-2007
- University Council on Technology Infrastructure Subcommittee 2004-2005
- University Council on Academic Computing Subcommittee 2004-2005
- University Council on Technology Supported Software Committee 2004-2005

Departmental Committees

- Member, Faculty Advisory Committee 1988-91, 1995-2009-
- Member, Curriculum Committee 2005-2008
- Member, Handbook Review Committee, 2005-08
- Member, Computer Science Search Committee 2005-2006
- Member, Computer Systems Committee 1990-2009
- Member, OBR Funding Committee 2001, 2003-2004, 2005-08

Dr. Angela Guercio
Assistant Professor, Department of Computer Science,
Kent State University at Stark, North Canton, OH 44720

Education:

- Ph.D. in Computer Science from Kent State University, Kent, OH, Dec. 2004. Dissertation title: "Introducing Triggers and Active Repository for Asynchronous and Nondeterministic Events in Reactive Multimedia Systems". Advisor: Prof. Arvind Bansal.
- Master of Science in Computer and Information Sciences from the Knowledge Systems Institute, Chicago, August 20, 2000.
- Doctor in Computer Science "cum laude" from the University of Salerno, Italy, March 20, 1984.

Work Experience:

- **TT Assistant Professor** at the Department of Computer Science Kent State University Stark, N.W. Canton, OH, August 2008 to present.
- **NTT Assistant Professor** at the Department of Computer Science Kent State University Stark, N.W. Canton, OH, August 2005-August 2008.
- **Assistant Professor** at the Department of Computer Science of Hiram College, OH, August 2002 – May 2005.
- **Senior Research Associate** since 1989 at the Department of Computer Science and Applications (May 1986-July 1996), Department of Pharmacy (July 1996-July 1998), and Department of Mathematics and Computer Science (July 1998 until 2002), **University of Salerno**, Italy.
- **Research Associate** from May 1986 until May 1989 at the Department of Computer Science and Applications, **University of Salerno**, Italy.
- November 1998 to 2000 – **Visiting Researcher** at Department of Computer and Information Science, **Cleveland State University**, Cleveland, OH.
- January 1997 to August 1998 – **Professor of Post Graduate School of Specialization** in Hospital Related Pharmacy, **University of Salerno**, Italy.
- September 1994 to August 1998 - **Adjunct Professor** at the Department of Computer and Systems Engineering, Faculty of Engineering, **University of Naples** "Federico II", Naples, Italy.
- December 1993 to May 1994 – **Visiting Assistant Professor** at **Southwest Texas State University**, San Marcos, Texas.
- September 1991 to Spring 1993 – **Adjunct Research Assistant Professor** at the Department of Computer Science, **Lamar University**, Beaumont, Texas.
- September 1987 to September 1988 - **Visiting Researcher** at the Department of Computer Science, **University of Pittsburgh**, Pittsburgh, Pennsylvania.

Publications: Journal Papers

[1] A. Guercio, A. Bansal, T. Arndt, "Languages Constructs and Synchronization in Reactive Multimedia Systems", *ISAST Transactions on Computers and Software Engineering*, ISSN 1797-1152, vol. 1, no.1, pp. 52-58, 2007.

[2] T. Arndt, S.K. Chang, A. Guercio, P. Maresca "An XML-Based Approach to Multimedia Software Engineering for Distance Learning", *Journal of Distance Education Technologies*, vol. 1, no.1, Jan. 2003, pp. 40-62.

[3] A. Guercio, T. Arndt, Guest Editors' Foreword to special issue on Multimedia Computing on the World Wide Web, *Journal of Visual Languages and Computing*, n.13, pp 1-2, February 2002.

[4] P. Maresca, T. Arndt, A. Guercio, "Unifying Distance Learning Resources: The Metadata Approach", *Journal of Computers*, vol.13 no.2, June 2001, pp.60-76.

[5] P. Maresca, A. Guercio, T. Arndt, G. Tortora "Multimedia Indexing with the SMART System", *Journal of Visual Languages and Computing*, vol. 11, pp. 405-438, 2000

[6] T. Arndt, S.K. Chang, A. Guercio, "Formal Specification and Prototyping of Multimedia Applications", *International Journal of Software Engineering and Knowledge Engineering*, vol. 10, no.4, pp.377-409, 2000.

[7] C. Crimi, A. Guercio, G. Nota, G. Pacini, G. Tortora, M. Tucci, "Relation Grammars and their Application to Multi-Dimensional Structures", *Journal of Visual Languages and Computing*, vol. 2, pp. 333-346, 1991.

[8] C. Crimi, A. Guercio, G. Pacini, G. Tortora, M. Tucci, "Automating Visual Language Generation", *IEEE Transactions on Software Engineering*, vol. SE-16, no. 10, pp. 1122-1135, October 1990.

[9] C. Crimi, A. Guercio, G. Pacini, G. Tortora, M. Tucci, "Grammatical Inference Algorithms for the Generation of Visual Languages", *Journal of Visual Languages and Computing*, vol. 1, pp. 355-368, 1990.

[10] S.-K. Chang, B. Yu, A. Guercio, G. Tortora, "Icon Purity: Toward a Formal Theory of Icons", *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 1, no. 3&4, pp. 377-392, 1987.

Refereed Book Chapters:

[11] T. Arndt, S.K. Chang, A. Guercio, P. Maresca, "An XML-Based Approach to Multimedia Software Engineering for Distance Learning", in Future Directions in Distance Learning and Communication Technologies (Advances in Distance Education Technologies - vol. 1) edited by Timothy Shih and Jason Hung, Idea Group Inc., pp. 106-134, 2007.

[12] A. Guercio, "Chapter 6: Queues" book chapter of the text book "*Data Structures and Algorithms*" published in 2003 by World Scientific Pub.

[13] P. Maresca, A. Guercio, T. Arndt, P. Donadio, "Transformation Dataflow In Multimedia Software Engineering Using TAO_XML: A Component-Based Approach", in *Multimedia Databases and Image Communication*, (edited by M. Tucci), *LNCS* 2184, Springer Berlin / Heidelberg, pp.77-89, 2001.

[14] A. Guercio, S.K. Chang, "Learning New Icons from Visual Sentences", in *Visual Languages and Visual Programming*, (S.K. Chang ed.), Plenum Press, New York, pp. 31-44, 1990.

[15] A. De Santis, A. Guercio, S. Levialdi, G. Tortora, "Extending Prolog for a Robotic Environment", *Image Analysis and Processing*, (V. Cantoni et al., eds.), Plenum Press, New York, pp. 235-240, 1985.

Refereed Conference Papers:

[16] A. Guercio, T. Arndt, "Towards Synchronization of a Distributed Orchestra", to appear in the Proc. of DMS 2009, The 15th Intl. Conf. on Distributed Multimedia Systems, San Francisco Bay, USA, September 10-12, 2009.

[17] T. Arndt, A. Guercio, "XML-Based Course Material Transformations For Ubiquitous E-learning Applications", *to appear in the Proc. of KMIS 2009, The Intl. Conf. on Knowledge Management and Information Sharing*, Madeira, October 6-8, Portugal, 2009.

[18] A. Guercio, A. K. Bansal, "Towards a Formal Semantics for Distributed Multimedia Computing", *Proc. of DMS 2007, The 13th Intl. Conf. on Distributed Multimedia Systems*, San Francisco Bay, USA, September 6-8, pp. 81-86, 2007.

[19] A. Guercio, A.K. Bansal, T. Arndt, "Synchronization for Multimedia Languages in Distributed Systems", *Proc. of DMS 2005, The 11th Intl. Conf. on Distributed Multimedia Systems*, Banff, Canada, September 5-7, pp. 34-39, 2005.

[20] A. Guercio, A. K. Bansal, "A Model for Integrating Deterministic and Asynchronous Events in Reactive Multimedia Internet Based Languages", *Proc. of the 5th Intl. Conf. on Internet Computing (IC 2004)*, Las Vegas, June 21-24, pp. 46-52, 2004.

[21] A. Guercio, A. K. Bansal, "TANDEM – Transmitting Asynchronous Nondeterministic and Deterministic Events in Multimedia Systems over the Internet", *Proc. of the 10th Intl. Conf. on Distributed Multimedia Systems*, San Francisco, pp. 57-62, September 2004.

[22] A. Guercio, B. Simoes A. K. Bansal, "Towards Large Scale Voice Activated Dynamic and Interactive Internet Based Animation and Modeling", *Proc. of IASTED-SEA04*, Nov. 9-11, Cambridge, MA, pp. 749-754, 2004.

Posters:

[23] "Bioinformatics Efforts at Hiram College", *25th Annual Crown Gall Conference in Champaign-Urbana*, IL (Univ. of Illinois), August 13-17, 2004. (Authors: Adam Ewing*, Luke Chaney*, Ryosuke Kadoi*, Spring 2004 Bioinformatics course, Angela Guercio*, & Brad Goodner+ (*Departments of Biology & Computer Science, +Departments of Biology, Hiram College, Hiram, OH 44234), & the K84/S4 Genome Collaboration).

[24] "TANDEM Constructs for a Real Time Distributed Interactive Conducting", *2009 Celebration of Scholarship*, Kent State University, Feb 10-12, 2009.

Technical Reports:

[25] A. Cafiero, A. Guercio, "Symbol Relation Grammars for Teleaction Objects", Technical Report, Dipartimento di Informatica ed Applicazioni, University of Salerno, 1997.

University Service:

- Member of the Search Committee for a position of Assistant Professor of Mathematics at Stark (Spring 2009)
- Advisor of the Stark Computer Club (2005 to Present)
- Member of Stark Technology Committee (2005-7 and 2008 to Present) (Kent State Univ.)
- Member of Women's History Month Committee (2007-8) (Kent State Univ.)
- Member of AURCO Committee (2007-8) (Kent State Univ.)
- Member of the Library Committee (2004-5) (Hiram College)
- Member of the Technology Advisory Committee (2003-4) (Hiram College)
- Member of the Campus Art Committee (2003-4) (Hiram College)
- Member of the Committee for the Organization and Creation of the newly instituted Diploma in Computer Science (1997-98), (Univ. of Salerno)

Dr. Ruoming Jin
Assistant Professor, Kent State University, Kent, OH 44242

Educational background
Ph.D. (2005) in Computer Science CSE Dept., Ohio State University
M.S. (2001) in Computer Science CIS Dept., University of Delaware
M.E. (1999), B.E. (1996) in Computer Engineering CSE Dept., Beihang University, China

Brief employment history
Assistant Professor, Aug. 2005 - Present
Computer Science Department (CS) , Kent State University
Research Scientist, Aug. 2001 - Sep. 2002
Department of Computer Science and Engineering (CSE), The Ohio State University
Member of Technical Staff, Feb. 1999 - June 1999
Bell Labs, Lucent Technology (Beijing, China)

List of representative publications:

1. Data Discretization Unification, Ruoming Jin, Yuri Breitbart, and Chibuike Muoh, in Knowledge and Information System (KAIS journal), Volume 19, Number 1, Pages 1-29, April, 2009.
2. Fast and Exact Out-of-Core and Distributed K-Means Clustering, Ruoming Jin, Anjan Goswami, and Gagan Agrawal, in Knowledge and Information System (KAIS journal), 10(1): 17-40 (2006) .
5. Cartesian Contour: A Concise Representation for a Collection of Frequent Sets, Ruoming Jin, Yang Xiang, and Lin Liu, in the Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'09), pages 417-426 (Full Paper, Long Presentation, Acceptance rate: 10%) .
6. Migration Motif: A spatial-temporal pattern mining approach for financial market, Xiaoxi Du, Ruoming Jin, Ling Ding, Victor Lee, and John Thornton, in the Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'09), pages1135-1144 (Industry/Application Track, Full Paper, Short Presentation, Acceptance rate: 27%).
7. Multiple Information Sources Cooperative Learning, Xingquan Zhu and Ruoming Jin, in the Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI), pages 1369-1375, Pasadena, California, USA (Full Paper, Acceptance rate: 25%).
8. 3-HOP: A High-Compression Indexing Scheme for Reachability Query, Ruoming Jin, Yang Xiang, Ning Ruan, and Dave Fuhry, in the Proceedings of ACM SIGMOD conference (SIGMOD'09), pages 813-826 (Full Paper, Acceptance rate: 15.9%).
9. Identify Dynamic Network Modules with Temporal and Spatial Constraints, Ruoming Jin, Scott McCallen, Chun-Chi Liu, Yang Xiang, Eivind Almaas, and Xianghong Jasmine Zhou, in the Proceedings of Pacic Symposium on Biocomputing (PSB'09) 14:203-214.
10. Efficient Skyline Computation in Metric Space, Dave Fuhry, Ruoming Jin, and Donghui Zhang, in the Proceedings of 12th International Conference on Extending Database Technology (EDBT'09), pages 1042-1051, March, 2009 (Full Paper, Acceptance rate: 32%).
11. Estimating the Number of Frequent Itemsets in a Large Database, Ruoming Jin, Scott Mccallen, Yuri Breitbart, Dave Fuhry, and Dong Wang, in the Proceedings of 12th International Conference on Extending Database Technology (EDBT'09), pages 505-516, March, 2009 (Full Paper, Acceptance rate: 32%).
12. Overlapping Matrix Pattern Visualization: a Hypergraph Approach, Ruoming Jin, Yang Xiang, Dave Fuhry, and Feodor Dragan, in International Conference on Data Mining (ICDM'08), pages 313-322 (Full Paper, Acceptance rate: 9.7%).
13. A Topic Modeling Approach and its Integration into the Random Walk Framework for Academic Search, Jie Tang, Ruoming Jin, Jing Zhang, in International Conference on Data Mining (ICDM'08), pages 1055-1060 (Short Paper, Acceptance rate: 19.9%).
14. Effective and efficient itemset pattern summarization: regression-based approaches, Ruoming Jin, Muad Abu-Ata, Yang Xiang, and Ning Ruan, in the Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'08), August, 2008, Pages 399-407 (Full Paper, Acceptance rate: 18.5%).

15. Succinct summarization of transactional databases: an overlapped hyperrectangle scheme, Yang Xiang, Ruoming Jin, Dave Fuhry, and Feodor Dragan, in the Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'08), August, 2008, Pages 758-766 (Full Paper, Acceptance rate: 18.5%).
16. Efficiently Answering Reachability Query on Very Large Directed Graphs, Ruoming Jin, Yang Xiang, Ning Ruan, and Haixun Wang, in the Proceedings of ACM SIGMOD conference (SIGMOD'08), June, 2008, Pages 595-608 (Full Paper, Acceptance rate: 18%).
17. Cost-Based Query Optimization for Complex Pattern Mining on Multiple Databases, Ruoming Jin, Dave Fuhry, and Abdulkareem Alali, in the Proceedings of 11th International Conference on Extending Database Technology (EDBT'08), March, 2008, pages 380-391 (Full Paper, Acceptance rate: 17.5%).
18. Data Discretization Unification, Ruoming Jin, Yuri Breitbart, and Chibuike Muoh, in the Proc. of the Seventh IEEE International Conference on Data Mining (ICDM'07), Oct. 2007, pages 183-192, (Full Paper, Acceptance rate: 7%, received two best paper nomination).\
19. Trend Motif: A Graph Mining Approach for Analysis of Dynamic Complex Networks, Ruoming Jin, Scott McCallen, and Eivind Almaas, in the Proc. of the Seventh IEEE International Conference on Data Mining (ICDM'07), Oct. 2007, pages 541-546 (Short Paper, Acceptance rate: 19%).

List of representative presentations:

1. Efficiently Answering Reachability Query on Very Large Directed Graphs, presentation at SIGMOD'08 (June 2008), invited talk at Penn State (Oct. 2008), Beihang University, IBM China Research Lab, AOL Beijing Research Lab (July, 2008).
2. Database Supports for Efficient Frequent Pattern Mining, invited talk at IBM T.J. Watson Research Center (May, 2008), Colloquium at Department of Computer Science, Wayne State University (March, 2008), invited talk at Max-Planck-Institut fcken, Germany, (August, 2007).
3. Scalable Data Mining: System Support and Algorithms, invited talk at Beihang University and Mathematics and Systems Institute of Chinese Academy of Sciences, June, 2007.
4. Towards a Systematic Approach for Genome-Wide Rice Gene Annotation, invited talk on the 3rd Rice Annotation Project Meeting (RAP3), Tsukuba, Japan, Dec. 9-10, 2006.

List undergraduate students (by name) mentored

Dave Stanfill, (B.S., Honors thesis, April 2008)

List masters and dissertation students (by name) supervised

Xiaoxi Du (M.S., April 2009), Dong Wang (M.S., Nov. 2008), Dave Furhy (M.S., April 2008), Scott McCallen (M.S., Dec. 2007)

Victor Lee (Ph.D. Student, Expected graduation year: 2011), Ning Ruan (Ph.D. Student, Expected graduation year: 2012), Muad Abuata (Ph.D. Student, Expected graduation year: 2012), Hui Hong (Ph.D. Student, Expected graduation year: 2013), Lin Liu(Ph.D. Student, Expected graduation year: 2013), Chibuike Muoh (Masters Student, Expected graduation date: Dec. 2009)

List of committees served and role, and professional service activities.

1. Sponsorship Co-Chair for SIAM Conference on Data Mining (SDM), 2009.
2. Senior Program committee Member for SIAM Conference on Data Mining (SDM), 2009.
3. Workshop Co-Chair for International Workshop on Mining Multiple Information Sources (MMIS), in conjunction with KDD'07, KDD'08, and ICDM'09.
4. Program committee (PC) Member for ACM conference on knowledge discovery and data mining (KDD), 2009, 2010; International Conference on Data Mining (ICDM), 2009; SIAM Conference on Data Mining (SDM), 2007, 2009; Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), 2007, 2008, 2009.
5. Members of ACM and IEEE.

DR. JAVED IQBAL KHAN

Education and Degree:

- Ph.D. in EE in Computer Track, Title: "Attention Modulated Associative Computing and Content Associative Search in Image Archive", University of Hawaii at Manoa (UHM), USA. August 1995[1].
- MS in EE in Computer Track, Title: "A Message Passing Approach to the Parallel Search of the Unification Tree", University of Hawaii at Manoa (UHM), August 1990, (GPA 4.0/4.0).
- B.Sc. in Electrical & Electronics Engineering, Bangladesh University of Engineering & Technology (BUET), 1987, (with the Distinction of Honors, 3rd out of 155).

Professional Appointments:

- Fulbright Visiting Professor, IIUM, Malaysia 2007-2006, BRAC University, Bangladesh 2005-2006
- Associate Professor: Dept. of Computer Science, Kent State University, Nov. 2002- May 2007.
- Assistant Professor: Dept. of Computer Science, Kent State University, Nov. 1997-2002.
- Post Doctoral Research Associate: Department of Electrical Engineering, University of Hawaii, USA, (April 1995-December 1997). Technical Manager of project MISSION, a $2.8 million NASA/DARPA sponsored Advanced Communication Technology Satellite, Medical Imaging and Supercomputing Triage Project. The project received the honor to be the semifinalist for the GII'97 award from over 900 submissions nationwide.
- Consultant Researcher: 1. ADTECH Co. (January 1997-December 1997) (MPEG-2 over ATM expert for a new high speed protocol analyzer/ test system project.). 2. Pacific International Center for High Technology Research (PICHTR), Hawaii, April 1992-August 1993.
- Graduate Research Assistantships: 1. Project MISSION, Sponsor DARPA/ NASA, Data/Image, Processing Lab. EE Dept, University of Hawaii, 1994-1995. 2. Reverse Engineering project, Sponsor Fujitsu Ltd of Japan, Software Engineering Research Lab., Dept. of Information and Computer Science University of Hawaii, 1992-93.
- Lecturer, Department of Computer Science & Engineering, Bangladesh University of Engineering & Technology (BUET), 1987-1989.

Special Appointments, Awards & Achievements:

- University Special Advisor on Cyber Infrastructure, Kent State University, 2007-2008.
- NASA Consultant, Lunar Communication Architecture for Exploration Missions, 2006-2007.
- US Air Force Research Fellow, AFRL/HE WRIGHT-PATT AFB, OH, Neurobiological/Multimodal Methods for Enhancing Unmanned Vehicle Operator Performance, Collaboration, and Communication, Human Effectiveness & Visual Communications Group, Wright-Patterson Air Force Research Lab, May 2007.
- Fulbright Senior Specialist, International Research and Higher Education Networks, Selected in the National Roster of Experts by the US Presidentially appointed J. William Fulbright Foreign Scholarship Board May, 2005-may 2010.

List of Publications: (Currently, more than 80+ peer reviewed first authored publications. GS Index shows number of citations (excluding self-citations) found in Google Scholar©. List follows)

Books & Book Chapters:

1. Knowledge Based Characterization of Test Questions, Javed I. Khan & Manas S. Hardas, Kent State University, Handbook of Research on Modern Systems Analysis and Design Technologies and Applications, ISBN: 978-1-59904-887-1, Publisher: Information Science Reference, Pub Date: July 2008.

2. Extreme Rate Distributed Video Transcoding System, Seung S. Yang, Virginia State University, USA, Javed I. Khan, Kent State University, USA, Multimedia Transcoding in Mobile and Wireless Networks, ISBN: 978-1-59904-984-7, Publisher: Information Science Reference, Pub Date: July 2008.

Publication:

1. WIC: Javed I. Khan & Sajid S. Shaikh, A Phenotype Reputation Estimation Function and its Study of Resilience to Social Attacks, **Journal of Network and Computer Application**, Elsevier, (April 2009, accepted Dec 2008, 1084-8045, Published by Elsevier Ltd. doi:10.1016/j.jnca.2008.12.003).
2. WIC: Javed I. Khan, Kailas Bobade and Manas Hardas, Personalized Negotiation Based on Individualization: Incorporating Personalization into Peer to Peer System, 5th IEEE International Conference on Information Technology: New Generations, ITNG 2008, Las Vegas, Nevada, USA, April 7-9, 2008, Pages 615-621.

[1] Selected research for the keynote presentation at the 11th Annual Governor' Symposium on High Technology 1995, State of Hawaii.

3. P2P: Javed I. Khan & Adam Wierzbicki, Guest editors' introduction Foundations of peer-to-peer systems, **Journal of Computer Communications**, Volume 31, Issue 2, 5 February 2008, pp187-189.
4. P2P: Javed I. Khan & Adam Wierzbicki, Guest editors' introduction: Disruptive networking with peer-to-peer systems, **Journal of Computer Communications**, Volume 31, Issue 3, 25 February 2008, pp419-422.
5. P2P: Javed I. Khan, Asrar U. Haque, Computing with Data Non-Determinism: Wait Time Management for Peer-to-peer Systems, Special Issue on Disruptive Networking with Peer-to-Peer Systems, **Journal of Computer Communications**, Elsevier Volume 31, Issue 3, 25 February 2008, pp.629-642.
6. MM: Oleg V. Komogortsev, and Javed I. Khan, Eye Movement Prediction by Oculomotor Plant Kalman Filter with Brainstem Control, Journal of Control Theory and Applications, Springer Publications, Volume 7. Number 1, February 2009, pp-14-22.
7. MM: Oleg V. Komogortsev, and Javed I. Khan, Eye Movement Prediction by Kalman Filter with Integrated Linear Horizontal Oculomotor Plant Mechanical Model, Proceedings of the Eye Tracking Research & Applications Symposium, ETRA 08, Savanna, GA, March 26-28, 2008, Pages 229-236.

Department Committee Membership

2003-2004, Library Representative
2004-2005, Chair Review Committee
2004-2005, Colloquium Committee
2004-2005, Faculty Search Committee
2003-2004, Library Representative
2002-2003, Library Representative
2004-2005, OBR Budget Committee
2005-2006, OBR Budget Committee
2006-2007, OBR Budget Committee
2006-2008, Graduate Studies Committee
2006-2008, Industrial Affiliates Committee
2008-2009, Chair, OBR Budget Committee
2008-2009, Industrial Affiliates Committee
Qualifying Exam Committee 2004-1998
Preliminary Exam Committee, 2004-2007

University Committee Membership

2008-2011, University Research Council (URC)
2008- 2009, Cyber Infrastructure Advisor to the Vice President for Information Services.

MASTERS THESIS DIRECTED:

- Kailash Bobade, Personalized Credential Negotiation Based on Policy Individualization in Federation, July 2008.
- Davu, Sandeep, Connection Oriented Mobility without Infrastructure with Transparent Networking, May 2008 (submitted).
- Sajid Shaikh, Computation Possibilities in Social Networks, July 2007.
- Olufunke Ibitesho Olaleye, Jitter and Delay Reduction for Audio Traffic, May 2007.
- Hardas, Manas Sudhakar, A Novel Approach for Test Problem Assessment Using Course Ontology, October 2006.
- Guo, Zhong, An Algorithm for Fast Perceptual Object Tracking in a Coded Stream by Analysis of Composite Scene Motion's Reflection on the Motion Vectors, June 2006.

DOCTORAL DISSERTATIONS DIRECTED:

- Oleg Komogortsev, Eye Movement Prediction by Oculomotor Plant Modeling with Kalman Filter, Doctoral Dissertation, December 2007.
- Nouman Bantan, Routing Protocol and Algorithm for Space Networking, PhD Dissertation, May 2007.
- Asrar Ul Haque, Bounded Wait Communication for Systems with Distributed And Autonomous Components, PhD Dissertation, 2006.
- Zaghal Raid, Interactive Protocols for Extensible Networking, PhD Dissertation, 2005.
- Yang, Seng-Su, A Framework for Complex Composition of Netcentric Systems, PhD Dissertation, 2004.

POST CANDIDACY DOCTORAL DISSERTATIONS:

- Omar Tahboub
- Manas Hardas

Cheng-Chang Lu
Professor

Cheng-Chang Lu received the B.S. degree in control engineering from the National Chiao Tung University, Hsinchu, Taiwan in 1983 and the M.S.E.E. and Ph.D degrees in electrical engineering from Southern Methodist University, Dallas, Texas, in 1985 and 1988 respectively. Since August 1988 he has been with Kent State University and he is currently Professor of Computer Science. His research interests are data compression, image processing, pattern recognition and multimedia data mining. In past, he has been funded by Cray Research Inc., Motorola Inc., NASA and Air Force.

Ph.D Dissertation: (in last 5 years)

1. Zhen Ye, ``Wavelet Domain Multiresolution Markov Model for Image Segmentation and Denoising Applications,'' December 2004.
2. Yujun Guo, ``Medical Image Registration and Application to Atlas-Based Segmentation,'' May 2007.
3. Mohammed Jirari, ``Computer Aided System for Detecting Masses in Mammograms,'' May 2008.

M.S. Thesis: (in last 5 years)

1. Elaine Ciocca, ``Automated Image Boundary Detection for Radiometric Adjustment of Monochrome Imagery Used in Flight Simulator Databases," Master of Science, December 2003.
2. Anthony Martinez, ``Determining the Three-Dimensional Location of a Cube from A Stereo Image," Master of Science, May 2004.
3. Mingming Lu, ``Color Image Segmentation using Color Quantization,'' August 2004.
4. Qiyu Zhnag, ``Dimension Reduction of Image Features for Multimedia Data Mining, '' August 2005.

Research Publications: (in last 5 years)

1. Mingming Lu, Qiyu Zhang, Wayne Cheng, Cheng-Chang Lu, "Retrieval of Multimedia Objects using Color Segmentation and Dimension Reduction Features," International Conference on Image and Graphics, Xi'an, China, Sep 2009.
2. Wayne Cheng and Cheng-Chang Lu, "Acceleration of Medical Image Registration using GPU in Computing Normalized Mutual Information," International Conference on Image and Graphics, Xi'an, China, Sep 2009.
3. Yujun Guo, Wayne Cheng, Cheng-Chang Lu, ``Non-Rigid Mammogram Registration using Demons Algorithm,'' IASTED International Conference on Signal and Image Processing, Honolulu, Hawaii, August 2007.
4. Dee Wu, Yujun Guo, Cheng-Chang Lu, Jasjit S. Suri, "Improvement to Functional Magnetic Resonance Imaging (fMRI) Methods Using Non-Rigid Body Image Registration Methods for Correction in Presence of Susceptibility Artifact Effects," IEEE Engineering in Medicine and Biology Society Conference, pp. 1018-1020, August 2006, New York.
5. Yujun Guo, Cheng-Chang Lu, "Multi-modality Image Registration Using Mutual Information Based on Gradient Vector Flow," International Conference on Pattern Recognition 2006, Hong Kong, p.697-700, August 2006.

6. Yujun Guo, Cheng-Chang Lu, "Multi-modality Image Registration Using Gradient Vector Flow Intensity," International Workshop on Medical Imaging and Augmented Reality 2006, Shanghai, China, pp.277-284, August 2006
7. Yujun Guo, Radhika Sivaramakrishna, Cheng-Chang Lu, Jasjit S. Suri, Swamy Laxminarayan, "Breast Image Registration Techniques: a Survey." Medical & Biological Engineering & Computing, Vol. 44, No.1-2, pp. 15-26, March 2006.
8. Qiyu Zhang and Cheng-Chang Lu, ``An Effective Scheme for Content Based Image Retrieval Systems,'' Proceedings of 2005 World Congress in Applied Computing, VISION'05, Las Vegas , June 2005.
9. Chi-Hsiang Lo, Yujin Guo, Cheng-Chang Lu, and Chi-Hua Tung, ``Future of Image Registration: A Multi-Resolution Approach to Medical Image Registration Using Binning Technique," Hand Book of Medical Image Analysis: Volume III - Registration Models: Part A, pp. 535-554, Springer/Kluwer, 2005.
10. Xiao-Hong Zhu, Cheng-Chang Lu, and Yang-Ming Zhu, ``Stereo and Temporal Retinal Image Registration by Mutual Information maximization,'' Hand Book of Medical Image Analysis: Volume III - Registration Models: Part A, pp. 151-184, Springer/Kluwer, 2005.
11. Zhen Ye and Cheng-Chang Lu, ``A Wavelet Domain Hierarchical Hidden Markov Model," Proceedings of 2004 IEEE International Conference on Image Processing, full text on CDROM, October 2004.

Professional societies/activities/awards

1. Member of Technical Program Committee, IASTED International Conference on Computer Graphics and Imaging, 1999, 2003-Present.
2. Member of International Program Committee, IASTED International Conference on Signal and Image Processing, 2003-Present.
3. Member of International Program Committee, IASTED International Conference on Telehealth, 2005-Present.
4. Member of Program Committee, IADIS Virtual Multi Conference on Computer Science and Information Systems, 2006.
5. Session Chair, Medical Image and Processing, 2007 IASTED International Conference on Signal and Image Processing.

Dept/College/Univ Committee Served

1. Graduate Coordinator and Chair of Graduate Studies Committee, 2003 - 2006.
2. Computer Science Colloquium Chair, 2006 – 2008.
3. Assistant Chair, 2008 – Present.
4. Assessment Committee, 2008 – Present.
5. College Graduate Council, 2003 - 2006.
6. International Students Advisory Committee, 2003 – 2005.

Jonathan I. Maletic, Associate Professor

Education

Ph.D.	Computer Science	Wayne State University	1995
M.S.	Computer Science	Wayne State University	1989
B.S.	Computer Science, Mathematics Minor	The University of Michigan-Flint	1985

Academic Experience

- *Associate Professor*. Department of Computer Science, Kent State University, Kent, Ohio. 8/04- present.
- *Assistant Professor*. Department of Computer Science, Kent State University, Kent, Ohio. 7/01 -8/04.
- *Assistant Professor*. Division of Computer Science, Department of Mathematical Sciences, The University of Memphis, Memphis, Tennessee. 9/97 – 5/02.
- *Lecturer*. Department of Computer Science, Wayne State University, Detroit, Michigan. 9/95 - 8/97.

Selected Publications (from over 80)

Hammad, M., Collard, M.L., Maletic, J. I., (2009), "Automatically Identifying Changes that Impact Code-to-Design Traceability", in Proceedings of the 16th IEEE International Conference on Program Comprehension (ICPC'09), Vancouver, BC, Canada, May 17-19, pp. 20-29. (*27% acceptance of full papers*)

Sutton, A. and Maletic, J. I., (2008), "Automatically Identifying C++0x Concepts in Function Templates", in Proceedings of the 24th IEEE International Conference on Software Maintenance (ICSM'08), Beijing China, Sept. 28 – Oct. 4, pp. 57-66. (*25% acceptance*)

Kagdi, H., Hammad, M., and Maletic, J. I., (2008), "Who Can Help Me with this Source Code Change?", in Proceedings of the 24th IEEE International Conference on Software Maintenance (ICSM'08), Beijing China, Sept. 28 – Oct. 4, pp. 157-166. (*25% acceptance*)

Alali, A., Kagdi, H., and Maletic, J. I., (2008), "What's a Typical Commit? A characterization of Open Source Software Repositories", in Proceedings of the 16th IEEE International Conference on Program Comprehension (ICPC'08), Amsterdam, The Netherlands, June 10-13, pp. 182-191. (*35% acceptance*)

Kagdi, H., Collard, M. L., and Maletic, J. I., (2007), "An Approach to Mining Call-Usage Patterns with Syntactic Context", in Proceedings of ACM/IEEE International Conference on Automated Software Engineering (ASE 2007), Atlanta, GA, Nov 5-7, pp. 457-460. (*11% acceptance of full paper, 24% of full and short*)

Sutton, A. and Maletic, J. I., (2007), "How we Manage Portability and Configuration with the C Preprocessor", in Proceedings of IEEE 23rd International Conference on Software Maintenance (ICSM'07), Paris, France, Oct. 2-4, pp. 275-284. (*21% acceptance*)

Kagdi, H. and Maletic, J. I., (2007), "Mining Evolutionary Dependencies from Web-Localization Repositories", *Journal of Software Maintenance and Evolution: Research and Practice*, vol. 19, no. 5, September, pp. 315-337. *(Invited submission to special issue on **Best Papers** of WSE'06*).

Yusuf, S., Kagdi, H., and Maletic, J. I., (2007), "Assessing the Comprehension of UML Diagrams via Eye Tracking", in Proceedings of 15th IEEE International Conference on Program Comprehension (ICPC'07), Banff, Canada, June 26-29, pp. 113-122. (*33% acceptance*)

Sutton, A. and Maletic, J. I., (2007), "Recovering UML Class Models from C++: A Detailed Explanation", *Information and Software Technology*, vol. 49, no. 3, March, pp. 212-229. (*Invited submission to special issue on **Best Papers** from WCRE'05*).

Kagdi, H., Collard, M. L., and Maletic, J. I., (2007), "A Survey and Taxonomy of Approaches for Mining Software Repositories in the Context of Software Evolution", *Journal of Software Maintenance and Evolution: Research and Practice*, vol. 19, no. 2, March/April pp. 77-131.

Dragan, N., Collard, M. L., and Maletic, J. I., (2006), "Reverse Engineering Method Stereotypes", in Proceedings of IEEE 22nd International Conference on Software Maintenance (ICSM'06), Philadelphia, Pennsylvania USA, September 25-27, pp. 24-34. (*28% acceptance*)

Marcus, A., Maletic, J. I., and Sergeyev, A., (2005), "Recovery of Traceability Links Between Software Documentation and Source Code", *International Journal of Software Engineering and Knowledge Engineering 2005*, vol. 15, no. 5, October, pp. 811-836.

Kagdi, H., Maletic, J. I., and Sutton, A., (2005), "Context-Free Slicing of UML Class Models", in Proceedings of 21st IEEE International Conference on Software Maintenance (ICSM'05) Budapest Hungary, September 25-30, pp. 635-638. (*30% acceptance*)

Marcus, A., Sergeyev, A., Rajlich, V., and Maletic, J. I., (2004), "An Information Retrieval Approach to Concept Location in Source Code", in Proceedings of 11th IEEE Working Conference on Reverse Engineering (WCRE'04), Delft, The Netherlands, Nov. 9th-14th, pp. 214-223. (*36% acceptance*)

Maletic, J. I. and Collard, M. L., (2004), "Supporting Source Code Difference Analysis", in Proceedings of 20th IEEE International Conference on Software Maintenance (ICSM'04), Chicago, Illinois, September 11-17th, pp. 210-219. (*32% acceptance*)

Funding Awards

J. Maletic, J. Hayes (University of Kentucky), J. Cleland-Huang (DePaul University), "CPA-SEL-T: Collaborative Research: Traceability+: A Service Oriented Framework to Support Value-Added Software Traceability", $750,000 Total, **$250,000** per institution, National Science Foundation, CCF 08-11021, 8/1/2008-7/31/2011.

J. Maletic (PI), M. Collard, "Automatic Platform Change via a Transformation Appraoch use srcML", **$60,199** ABB Inc., 7/1/2008-6/30/2009.

J. Maletic (PI), J. Ortiz, R. Selinger, J. Portman, S. Lee, "S-STEM Scholarships for Broadening Participation in Sciences", **$499,926** National Science Foundation, DUE 06-31088, 9/15/2006-8/31/2011.

J. Maletic (PI), L. Bartolo, K. Stemem (student), NIST Summer Undergraduate Research Fellowship - MSEL, **$6,576**. National Institute of Standards 70NANB3H1025, 5/2003-8/2003.

J. Maletic (PI), "A Framework to Combine Semantic and Structural Information for Static Analysis", **$150,000**, National Science Foundation, C-CR 02-04175, 8/1/2002-7/31/2005.

Students

Ph.D. Completed (2): Michael L. Collard, Ph.D., Huzefa Kagdi, Ph.D.

Ph.D. Currently Directing (9): Abdulkareem Q. Alali, Hakam Alomari, Gregory Delozier, Natalia Dragan, Xiaoxi Du, Maen Hammad, Dale Haverstock, Bonita Simoes, Andrew Sutton

M.S. (thesis) Completed (5): Abdulkareem Q. Alali, Natalia Dragan, Alice Lewis, Andrew Sutton, Shehnaaz Yusuf

M.S. (thesis) Currently Directing (1): Ryan Holeman

Undergraduate Directed (10): Brian Bartman, James Brundage, Rodney Johnston. Michael Lopez, Edward Morrison, Dylan Shearer, Kyle Stemen, Matthew Suhay, Christopher Tuttle, Chris Wagner

Professional Service

Program Co-Chair for the IEEE 14th Working Conference on Reverse Engineering (WCRE 2007), Vancouver, B.C., Canada, October 29-31.

Finance Chair for the ACM/IEEE 22nd International Conference on Automated Software Engineering (ASE 2007), Atlanta, Georgia Nov. 5-9.

General Chair for the IEEE 13th International Workshop on Program Comprehension (IWPC 2005) St. Louis, Missouri, May 15-16

Steering Committee Membership: IEEE International Conference on Program Comprehension (ICPC) – Elected 5/02 to current (Chair '07-'10), IEEE Workshop on Visualizing Software for Understanding and Analysis (VISSOFT) 5/06-current, ACM Int. Workshop on Traceability in Emerging Forms of Software Engineering (TEFSE) 6/05 to current, ACM Symposium on Document Engineering (DocEng) 11/02-10/05

Conference Program Committee Membership: ACM/IEEE International Conference on Automated Software Engineering (ASE) '09-'04, IEEE International Conference on Software Maintenance (ICSM) '09, '07-'05**,** IEEE International Conference on Program Comprehension (ICPC) '09-'06, '04, ACM/IEEE International Conference on Software Engineering (ICSE'09) -Research Demonstrations Track, IEEE Working Conference on Reverse Engineering (WCRE) '09-'08, '06-'05, ACM Workshop on Mining Software Repositories (MSR) '09-'06, ACM International Workshop on Traceability in Emerging Forms of Software Engineering (TEFSE'09), IEEE Workshop on Visualizing Software for Understanding and Analysis (VISSOFT'09), IEEE International Workshop on Principles of Software Evolution (IWPSE) '09, '07, '05, '04, ACM Symposium on Software Visualization (SoftVis) '08, '06, '05, ACM/IEEE International Conference on Model Driven Engineering Languages and Systems (MoDELS'05), ACM Symposium on Document Engineering (DocEng'04)

NSF Grant Proposal Reviewer: Three times since 2004

Reviewer for Journal: IEEE Transactions on Software Engineering, ACM Transactions on Software Engineering and Methodologies, Journal of Software Maintenance and Evolution, Journal of Systems and Software

Austin Melton
Professor of Computer Science and Mathematical Sciences

Brief Education History:
M.S. (1974), Ph.D. (1980) mathematics Kansas State Univ
B.A. (1971) English, High Honors, Friends University

Brief Employment History:
Kent State University, 1996-present, Dept. Chair Math and CS (1996-2000), Interim Associate Dean (2004-2005)
Michigan Tech, 1992-1996, CS Dept Chair and Chair Computational Science and Engineering Ph.D. Program
Kansas State University, CS, 1984-1992
Wichita State University, CS, 1982-1984
Marshall University, Math, 1980-1982
Visiting Positions Universities of Bremen (Germany), Cape Town, Copenhagen, Stellenbosch (South Africa)

Representative Publications:
Kehinde O. Jolayemi and Austin Melton, “Software Measurement Needs Its Own Theory”, Proceedings of The 2008 International Conference on Software Engineering Research and Practice, {\em SERP 2008, Volume II}, Hamid R. Arabnia and Hassan Reza -- editors, ISBN 1-60132-086-8, 1-60132-087-6 (1-60132-088-4), CSREA Press, Las Vegas, Nevada, 2008, pages 725-728.
P. N. Gedela, P. Kosaraju, and A. Melton, “Measurement Theory Principles in Software Measurement”, Proceedings of the 16th International Conference on Software Engineering and Data Mining, M. Al-Mubaid and M. Garbey -- editors, ISBN 978-1-880843-63-5, Las Vegas, Nevada, 2007, pages 344-348.
BB Senthil Kumar and Austin Melton, “Understanding and Developing Internet Ethics”, The International Journal of Technology, Knowledge, and Society}, Volume 2, Number 7, 2007, pages 11-14.
Austin Melton and Sonja Melton, “Using PowerPoint in Theme Development”, The International Journal of the Book, Volume 4, Number 3, 2007, pages 23-25.
A. Bathi Kasturiarachi, Austin Melton, and Beverly M. Reed, Classroom-Relevant Advanced Mathematics for Middle School and High School Teachers, Proceedings of the Third International Conference on the Teaching of Mathematics at the Undergraduate Level, Turkey, 2006.
Austin Melton, “Introducing Lagois Correspondences”, Categorical Structures and Their Applications, World

Scientific Publishing Company, editors W. Gaehler and G. Preuss, Berlin, 2004, 207-217.

Awards:
Co-principal Investigator for KSU's part in Wright State University's and business leaders of Dayton's $11,000,000 Wright Center of Innovation for Advanced Data Management and Analysis. Paul Farrell is KSU's Principal Investigator. KSU's part is $400,000 for visualization equipment to be part of a state-wide visualization project.
Co-Principal Investigator with PI James Blank and co-PI Roger Gregory, Research and Education in Cell Systems Biology, U.S. Department of Education, 30 September 2003-29 September 2006, $1,241,875.

Representative Presentations:
29th Linz Seminar on Fuzzy Set Theory: Foundations of Lattice-Valued Mathematics with Applications to Algebra and Topology, Linz, Austria, 2008
Categorical Methods in Algebra, Topology and Computer Science Workshop, University of Coimbra, Portugal, 2007
Technology Conference, Cambridge University, England, 2007
Tenth Prague Topological Symposium, Czech Republic, 2006
Youngstown State University, 2006
Fifth Southern Hemisphere Symposium on Undergraduate Mathematics Teaching, Australia, 2005

Master's Students Supervised:
Roger Kurian, Summer 2009
Binamra Dutta, Spring 2009
Pradeep Kumar Punnam, Fall 2008
NV Praveen Babu Gedek, Fall 2008
Amruta Hingane, Summer 2008
Vikash Sananda, Spring 2008
Srikanth Saladi, Spring 2007
Sumit Sukhramani, Fall 2006
Abhishel Asokan, Fall 2006
Jidesh Soudamini, Fall 2006
Abha Gupta, Summer 2006
Senthil Bogana Balaraj, Summer 2006
Shashikant Shinde, Fall 2005

Professional Service Activity:
Chair, Statewide Users Group for Ohio Supercomputer Center, 2003-2005

Mikhail Nesterenko

Work Experience

- **Associate Professor**, Dept. of Computer Science, Kent State University (2004-present);
- **Invited Professor,** Université Pierre et Marie Curie - Paris 6, France (May 2009);
- **Invited Professor,** Université Paris Sud, France (December 2004, July 2006, May-June 2007);
- **Invited Professor,** École Polytechnique Fédérale de Lausanne, Switzerland (February-March 2006);
- **Assistant Professor**, Dept. of Computer Science, Kent State University (1998-2004);
- **Computer and Network Manager,** Computing and Info. Sciences Dept., Kansas State University (1996 - 1998);
- **Systems Administrator,** Computing and Info. Sciences Dept., Kansas State University (1994-1996).

Publications Journal, Book Chapters

- M. Nesterenko, S. Tixeuil, "Discovering Topology in the Presence of Byzantine Faults", *IEEE Transactions on Parallel and Distributed Computing*, to appear;
- P. Danturi, M. Nesterenko, S. Tixeuil, "Self-Stabilizing Philosophers with Generic Conflicts" *ACM Transactions on Autonomous and Adaptive Systems*, 4(1), 2009;
- S. Dolev, M. Kopeetsky, T. Clouser, M. Nesterenko "Low Overhead RFID Cryptography", in *"RFID Handbook: Applications, Technology, Security and Privacy*", CRC Press, March 2008;
- A. Vora, M. Nesterenko "Secure Location Verification Using Radio Broadcast", *IEEE Transactions on Dependable and Secure Computing*, 3(4) 2006, pp.377-385;
- A. Arora, E. Ertin, R. Ramnath, M. Nesterenko, B. Leal "Kansei: A High Fidelity Sensing Testbed" *IEEE Internet Computing*, March-April (Vol. 10, No. 2) 2006, pp. 35-47;
- A. Arora, M. Nesterenko "Unifying Stabilization and Termination in Message-Passing Systems" *Distributed Computing* 17(3), March 2005, pp. 279-290;

Conference

- R. Nor, M. Nesterenko and P. Lavrentyev "Oxybuoy: Constructing a Real-Time Inexpensive Hypoxia Monitoring Platform", *International Workshop on Advanced Sensor Integration Technology (ASIT 2009)*, to appear;
- S. Delaet, S. Devismes, M. Nesterenko and S. Tixeuil "Stabilisation instanée dans les systèmes à passage de messages", *AlgoTel*,2009, to appear;
- S. Delaet, S. Devismes, M. Nesterenko and S. Tixeuil "Snap-Stabilization in Message-Passing Systems" *10th International Conference on Distributed Computing and Networking (ICDCN 2009)*, pp. 281-286, Hyperabad, India, January 2009;
- T. Clouser, M. Miyashita, M. Nesterenko "Fast Geometric Routing with Concurrent Face Traversal", *12th International Conference on Principles of Distributed Systems (OPODIS 2008)*, pp. 346-362, Luxor, Egypt, December 2008;
- A. Vora, M. Nesterenko, Sébastien Tixeuil, S. Delaët "Universe Detectors for Sybil Defense in Ad Hoc Wireless Networks", *10th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS'08)*, LNCS vol. 5340, pp. 63-78, Detroit, MI, November 2008.
- T. Clouser, M. Nesterenko, C. Scheideler "Tiara: A Self-Stabilizing Deterministic Skip List", *10th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS 2008)*, LNCS vol. 5340, pp 124-140, Detroit, MI, November 2008.
- N. Alam, T. Clouser, R. Thomas and M. Nesterenko "Poster Abstract: Emuli - Model Driven Sensor Stimuli for Experimentation" *6th ACM Conference on Embedded Networked Sensor Systems (SenSys 2008)*, pp. 423-424, Raleigh, NC, November 2008.
- S. Delaet, S. Devismes, M. Nesterenko and S. Tixeuil "Snap-Stabilization in Message-Passing Systems (brief announcement)" *27th ACM Symposium on Principles of Distributed Computing (PODC)*, p. 443, Toronto, Canada, August 2008.
- P. Danturi, M. Nesterenko. S. Tixeuil. "Self-Stabilizing Philosophers with Generic Conflicts", *Eighth International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS 2006)*, LNCS Volume 4280, pp. 213-230, Dallas, TX, November 2006;
- S. Pleisch, T. Clouser, M. Nesterenko, A. Schiper "DRIFT: Efficient message ordering in ad hoc networks using virtual flooding" *25th IEEE Symposium on Reliable Distributed System (SRDS 2006)*, pp. 119-131, Leeds, UK, October 2006;
- R.S. Whittlesey-Harris, M. Nesterenko "Fault-Tolerance Verification of the Fluids and Combustion Facility of the International Space Station" *5th International Workshop on Assurance in Distributed Systems and Networks (ADSN 2006)*, pp. 5-14, Lisbon, Portugal, July 2006;

- M. Nesterenko, S. Tixeuil "Discovering Network Topology in the Presence of Byzantine Faults" *13th Colloquium on Structural Information and Communication Complexity (SIROCCO 2006)*, LNCS Volume 4056, pp. 212-226, Chester, UK, July 2006;
- E. Ertin et al (10 co-authors) "Kansei: A Testbed for Sensing at Scale" *4th Symposium on Information Processing in Sensor Networks (IPSN/SPOTS 2006)*, pp. 399-406, Nashville, TN, April 2006;
- S. Karthikeyan, M. Nesterenko "RFID Security without Extensive Cryptography" *the 3d ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN)* pp. 63-67, Washington, DC, November, 2005;
- M. Nesterenko, A. Vora "Void Traversal for Guaranteed Delivery in Geometric Routing" *the 2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS)*, Washington, DC, November 2005;

Students Supervised

- Rizal Mohd Nor, MS *"OxybuoyOxybuoy: Constructing a Real-Time Inexpensive Hypoxia Monitoring Platform"*, August 2009;
- Richie John Thomas, MS, "*Experimental Evaluation of Emuli: A Tool for Sensor Abstraction in Wireless Sensor Networks*", October 2007;
- Praveen Danturi, MS, "*Self-Stabilizing Philosophers with Generic Conflicts*", February 2007;
- Mark Miyashita, MS, *"Double Face for Geometric Routing"*, August 2006;
- Thomas Clouser, MS, "*Efficient Message Ordering in Ad Hoc Networks*", June 2006;
- Pradeep Pullagurala, MA, "*Firewall Verification for Virtual Private Networks*", October 2005;
- Romil D. Shah, MS, *"TosGUI: A Visualization Tool for Wireless Sensor Networks"*, October 2005;
- Sindhu Karthikeyan, MS, "*RFID Authentication: Security without Cryptography*", July 2005;
- Raquel Whittlesey-Harris, MS, "*Fault-Tolerance Verification of the Fluids and Combustion Facility of the International Space Station*", July 2005;
- Prasad Karnik, MA, "*Geometric Routing for Wireless Sensor Networks*", July 2005;

Thesis and dissertation committee member for about 20 MS and PhD students

Professional Activities: Program Committee Member

- 11th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS) 2009;
- 8th International Workshop on Assurance in Distributed Systems and Networks (ADSN) 2009;
- 10th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS) 2008;
- 19th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC) 2008;
- 28th International Conference on Distributed Computing Systems (ICDCS) 2008;
- 9th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS) 2007;
- 1st International Workshop on Localized Algorithms and Protocols
- for Wireless Sensor Networks (LOCALGOS) 2007;
- 27th International Conference on Distributed Computing Systems (ICDCS) 2007;
- 8th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS) 2006;
- 10th International Conference On Principles Of Distributed Systems (OPODIS) 2006;
- 26th International Conference on Distributed Computing Systems (ICDCS) 2006;
- 5th International Workshop on Assurance in Distributed Systems and Networks (ADSN) 2006;
- 2nd Workshop on Pervasive and Wireless Networking (PWN) 2006;

Session Chair

- International Workshop on Advanced Sensor Integration Technology (ASIT 2009);
- 12th International Conference on Principles of Distributed Systems (OPODIS 2008);
- 8th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS) 2006;

Proposal Reviewer

- NSF CommNets panel, April 2009;
- US Civilian Research and Development Foundation (CRDF) Cooperative Grants Program;
- Ohio Board of Regents Research Challenge.

University Activities

- CS Department technical report series coordinator 2004--;
- Faculty search committee chair, Fall 2007;
- Colloquium chair - 1999-2000;
- Systems committee member 2001-2003, 2008-;
- Graduate studies committee member 2001-2003, 2007-2008.

Short Biography

Dr. Peyravi has joined the faculty of Computer Science at Kent State University in 1985. From 1987 to 1989, he was a Member of Technical Staff in AT&T Bell Laboratories in New Jersey. During this period, he conducted research in network architecture, adaptive routing algorithms for ISDN network controllers, and frame relay. He has been awarded two research grants from NASA Space Communication Division to design, study and evaluate the performance of multiple access control (MAC) protocols for the Mars Regional Network. He has been awarded by CAIDA (Cooperative Association for Internet Data Analysis) to establish an Internet Engineering and Teaching Laboratory at Kent State University, and to study traffic management, network management and routing protocols for IP routers. Recently, He has been awarded by the ¡a Internet2 Technology Evaluation Center ITEC to study the QoS provisioning and traffic management in the next generation of IP networks. Dr. Peyravi has participated in the review process of numerous transactions papers and journal articles for the major computer and communication societies including ¡a IEEE, ACM, Elsevier, The Society for Modeling and Simulation International, and The International Society for Optical Engineering. He has also participated in the review process of numerous NSF panels and a few academic review boards. Dr. Peyravi has served as a member of the executive committee for the Internet2 Technology Evaluation Center (ITEC-Ohio), and a member of the program committees for several international conferences including International Conference on Parallel Processing (ICPP), Wireless Networks and Mobile Computing, QoS over Next Generation Data Networks, and the International Society for Optical Engineering. He has served as curriculum director and graduate programs director in Mathematical Sciences and Computer Science Departments. Dr. Peyravi's research encompasses multiple access protocols for wireless and satellite communications, traffic management and congestion control, optical switching and transmissions, interconnection networks, systems modeling and performance evaluations.

Education

Ph.D. Computer Science, School of Electrical Engineering and Computer Science, University of Oklahoma, Norman, Oklahoma, August 1985. Dissertation: *Study of Interconnection Networks*.

M.S. Computer Science, School of Electrical Engineering and Computer Science, University of Oklahoma, Norman, Oklahoma, December 1980.

B.A. Arts & Sciences, Shiraz (formerly Pahlavi) University, Shiraz, Iran, August 1977.

Professional Experience

Academic Positions

- Professor of Computer Science, Kent State University, Kent, Ohio, 2000-present.
- Tenured Associate Professor of Computer Science, Kent State University, 1993-00.
- Curriculum Coordinator, Department of Computer Science and Department of Mathematical Sciences, Kent State University, 1993-99.
- Tenure track Assistant Professor of Computer Science, Kent State University, (1985-92).

Industrial Positions

AT&T Bell Laboratories, 1987-89: Member of Technical Staff, Network System and Computer System Architecture groups. During this period, he was the sole project leader for the following projects.

- Performance analysis of RISC processors for 4ESS/5ESS switching systems.
- Adaptive routing schemes for bandwidth management offering in ISDN networks.
- Digital circuit emulation (DICE) terminal adapter protocol.

Space Communications Division, NASA Glenn Research Center,1995-97:

- Feasibility study of wide area networks on the Mars planet through simulation and theoretical analysis.

Publications

[1] Y Drabu and H. Peyravi. An adaptive bandwidth control algorithm for IP routers. In *Proceedings International Conference on Communications in Computing (CIC)*, pages 311–317, Las Vegas, USA, June 2004.

[2] M. K. Khan and H. Peyravi. Delay and jitter analysis of generalized demand-assignment multiple access (DAMA) protocols with general traffic. In *HICSS '05: Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05) - Track 9*, page 304.1, Washington, DC, USA, January 2005. IEEE Computer Society.

[3] Y. Drabu and H. Peyravi. Fault tolerant routing in tri-sector wireless cellular mesh networks. In *Proceedings International Conference Parallel and Distributed Computing Systems*, pages 209–216, San Francisco, California, USA, September 2006.

[4] Y. Drabu and H. Peyravi. Gateway placement with QoS constraints in wireless mesh networks. In *Proceedings of the IEEE International Conference on Networking*, pages 46–51, Cancun, Mexico, April 2008. IEEE Computer Society.

[5] Y. Drabu and H. Peyravi. Fault recovery with QoS provisioning in wireless mesh networks. *The 2009 International Conference on Wireless Networks (ICWN'09)*, pages 1–8, July 2009.

[6] MD. Amiruzzaman, H. Peyravi, M. Abdullah-Al-Wadul, and Yoojinn Chung. An improved steganographic covert channel.

Theses Directed

[1] Qun Du. Performance of class based priority queuing and token bucket filtering in IP routers with self-similar traffic. Master's thesis, Kent State University, October 2004.

[2] Ranjana Tripathi. Performance comparison of RED algorithms for self-similar differentiated services. Master's thesis, Kent State University, July 2004.

[3] Jalaa Hoblos. Selfish node misbehaving statistical detection with active MAC layer NAV attack in wireless networks. Master's thesis, Kent State University, November 2006.

[4] Maha Allouzi. A dynamic routing algorithm under shared risk link group constraints for WDM mesh optical networks. Master's thesis, Kent State University, August 2008.

[5] Asif Iqbal. Monitoring remote financial transaction control devices using SNMP over TCP. Master's thesis, Kent State University, March 2009.

[6] Yasir Drabu. *QoS Provisioning in wireless networks*. PhD thesis, Kent State University, In Progress.

[7] Jalaa Hoblos. *Intrusion Detection Techniques*. PhD thesis, Kent State University, In Progress.

Committees Served (2004-2009)

- Member University Teaching Council 2004-2005.
- Member of College Advisory Committee, 2006-2007.
- Coordinator of Graduate Studies Committee, 2006-2009.
- Coordinator of Preliminary Examination Committee, 2006-2009.
- Member of Collage Graduate Council, 2006-2009.

Publications Reviewed for (2204-2009)

- IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems, IEEE Communications Letters, IEEE Communications Magazine, Journal of Parallel and Distributed Computing, Parallel Processing Letters, Computer Communications. IEEE Communications, IEEE Networks, IEEE Transactions on Vehicular Technology. IEEE Transactions on Wireless Communications. Computer Communications Journals, IEEE INFOCOM.

Funding Panel Review (2004-2009)

- National Science Foundation, Engineering Division, Wireless/Sensor Networks.
  February 2004, March 2006, and March 2007, March 2008.
- National Science Foundation, Engineering Division, Networks Architecture, August 2009.
- National Science Foundation, Engineering Division, Security, September 2009.
- Research and Graduate Studies, KSU, Oct 2007.

Name: Michael Rothstein

Current Title: Associate Professor

Educational Background:

1970: Degree in Mathematics, (B. S.) Universidad Nacional de Colombia
1973 M. S. in Computer Science, University of Wisconsin, Madison
1976 Ph. D. in Computer Science, University of Wisconsin, Madison

Brief Employment History:

1966-1971 Universidad Nacional de Colombia
1970-1971 Professor, Universidad Catolica de Colombia
1971-1975 Teaching/Research Assitant, University of Wisconsin, Madison
1975-1976 Postdoctoral Fellow, The University of Utah
1977-1979 Universidad Simon Bolivar, Caracas, Venezuela
1980- Kent State University

Masters Students Supervised:

Vyeni Ghildyal
Swetha Vasudevan
Amit Kumar Pandey
Nathan Truhan
Sam Thomas
Aruna Vanukuru
Aditya Prahalad
Amruta Hingane
Sujatha Gnaneswaran
R Ryan Kenyon

Committees served :

Departmental Systems Committee
Departmental Curriculum Committee
University Transportation Committee (One year as chair)
University Requirements Curriculum Committee (alternate member)

Arden Ruttan
Department Computer Science, Kent State University, Kent, OH 44242.

Education

1969 B.A. Mathematics Lehigh University 1977 Ph. D. Numerical Analysis Kent State University

Professional Experience

1977-78 Instructor, California Institute of Technology 1978-83 Assistant Professor, Texas Tech University 1983-88 Assistant Professor, Kent State University 1988-1995 Associate Professor, Kent State University 1996-Professor, Kent State University

Arden Ruttan is an active researcher in the area of scientific computing, scientific visualization, computational steering, parallel and distributed algorithms, cluster computing, liquid crystal modeling, highly ill-conditioned mathematical computations, and *a priori* algorithm selection and *a postiori* error analysis for numerical routines. His work is currently supported by the State of Ohio, and he has received NSF ITR, ASC, and infrastructure grants. He is a founder and editor of of the electronic journal, Electronic Transactions on Numerical Analysis which is a peer reviewed journal currently in its 16th year of publication.

Recent Publications:

1. Dan Bennett, Paul Farrell, and Arden Ruttan, *An Extensible Parameter Viewer for CUMULVS*, Proceedings of PACISE 20O9, 10–15, (2009)
2. Paul A. Farrell, Hong Ong, Arden Ruttan, Yang-Ming Zhu, *A Visualization Environment for Nematic Liquid Crystal Materials*, Proceeding of IADIS Multi Conference on Computer Science and Information Systems, Lisbon, Part III -Computer Graphics and Visualization, pp. 83–91, 2007.
3. N. Malhis, and A. Ruttan, *Detecting Gene Regulation Relations from Microarray Time Series Data*, Proceedings of the 2006 International Conference on Machine Learning; Models, Technologies & Applications MLMTA'06, June 26-29, 2006, Las Vegas, Nevada, USA.
4. Cara Stein, Daniel Bennett, Paul A. Farrell, Arden Ruttan, *A Steering and Visualization Toolkit for Distributed Applications*, The Proceedings of the 2006 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'06), pp 451–457.
5. N. Malhis, and A. Ruttan, *Predicting Gene Regulatory Networks from Micro Array Time Series Data by Elimination*, Proceedings of the Second Annual BioTechnology and BioInformatics Symposium BIOT-05, October 20-21, 2005, Colorado Spring, Colorado, USA, pp. 37-42.
6. N. Malhis, A. Ruttan, and H. H. Refai *An Efficient Approach for Candidate Set Generation*, Journal of Information & Knowledge Management, Vol. 4, No. 4 (2005) 287-291.

Recent Research Funding:

2004 J.Blank, P.Farrell, and A.Ruttan, *A Parallel Environment for Interactive Real Time Analysis, Quantification, and Display of 3D Confocal Microscopy Images*, Ohio Board of Regents, $60,000,7/01/04-06/31/06.

2005 K. Khan, L.Reichel, A.Ruttan,*Image Restoration in Biology and Material Sciences*, Ohio Board of Regents, $50,000,06/01/06-05/31/07.

2007 Paul Farrell, Austin Melton, Arden Ruttan, Robin Selinger, Wright Center of Innovation for Advanced Data Management and Analysis, Ohio Department of Development, Subcontract,$400,000, 4/10/07-6/30/08.

2008 P. Farrell, J. Li, R. Reichel, A. Ruttan, X. Zheng, SCREMS: High Performance Scientific Computing Environment. $113,522, 9/1/08-8/31/10.

Committees Served:

System Committee, Chair 2004-2007 OBR Committee, member 2005 Faculty Search Committee, Chair 2005 Provost's Promotion Advisory Board 2007 Qualifying Exam Committee, January 2007 OBR Committee, member 2007 Graduate Studies, member 2008 Lead Faculty Advisor 2009

Professional Activities:
Editorial Board, Electronic Transactions on Numerical Analysis. 1993-Present.

Graduate Students:
Dan Bennett (Ph.D. Expected Spring 2010) Kamal Tejwani M.S. October 2008 Brad Bellomo M.S. July 2008
Dr. Nawar Malhis, Ph.D., December 2006 Manyu Tang, M.S. August 2005 Rohit Chhabra M. A. June 2005

L. Gwenn Volkert

Associate Professor

Education

Degree: Institution: Date of Graduation:
Ph.D. (Computer Science) Wayne State University May 2001
Defended 01/22/01 Computer Science Department
Detroit, Michigan
M.S. (Computer Science) Wayne State University
Computer Science Department May 1995
Detroit, Michigan
B.S. (Computer Science) Central Michigan University May 1983
Department of Computer Science
Mt. Pleasant, Michigan

Academic Appointments

Associate Professor**,** Kent State University, Ohio 2007 - present
Department of Computer Science
Assistant Professor, Kent State University, Ohio 2001 - 2007
Department of Computer Science
Visiting Faculty, The University of Memphis, Tennessee 1999 - 2000
Department of Mathematical Sciences, Division of Computer Science

Publications and Scholarly Work

Refereed Publications

1. Amin Assareh and L. Gwenn Volkert (2009) "Fusing Fuzzy Rule Base Classifiers for Improving Protein Mass Spectra Based Ovarian Cancer Diagnosis ", Proceedings of the IEEE 2009 Symposium on Computational Bioinformatics and Computational Biology, CIBCB'09 Nashville, TN, USA, March 30 - April 2, 2009, IEEE Press, Piscataway, NJ
2. Deborah A. Stoffer and L. Gwenn Volkert, “Evolving Chaos Automata for Fractal Based Visualizations of Protein Sequence Data”, in proceedings of IEEE 2008 Symposium on Computational Bioinformatics and Computational Biology, Sept. 15-17, 2008 Sun Valley, Idaho, IEEE Press, Piscataway, NJ.
3. Amin Assareh, Mohammad Hassan Moradi, L. Gwenn Volkert (2008) “A Hybrid Random Subspace Classifier Fusioin Approach for Protein Mass Spectra Classification”, 6th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics, Springer Lecture Notes in Computer Science Vol 4973.
4. L. Gwenn Volkert (2006) “Investigating EA Based Training of HMM using a Sequential Parameter Optimization Approach”, Proceedings of the IEEE 2006 Congress on Evolutionary Computation, Vancouver, BC, Canada, IEEE Press, Piscataway, NJ.
5. Derek G. Dimcheff, L. Gwenn Volkert, Ying Li, Angelo L. Delucia and William P. Lynch (2006) “Gene Expression Profiling of Microglia Infected by a Highly Neurovirulent Murine Leukemia Virus: Implications for Neuropathogenesis” *Retrovirology*, 3:26 pp. 1—19.
6. L. Gwenn Volkert (2006) “Abstract: Evolving HMMs for Disordered Protein Analysis”, Ohio Collaborative Conference Bioinformatics (OCCBIO-2006), June 28-30, 206, Athens, OH.
7. Deborah A. Stoffer and L. Gwenn Volkert (2005) "A Neural Network for Predicting Protein Disorder using Amino Acid Hydropathy Values”, Proceedings of the IEEE 2005 Symposium on Computational Bioinformatics and Computational Biology, Nov. 12 and 13, 2005, La Jolla, California, IEEE Press, Piscataway, NJ. pp. 482—489.
8. Andrew Sutton, Huzefa Kagdi, Jonathan I. Maletic, L. Gwenn Volkert (2005) "Hybridizing Evolutionary Algorithms and Clustering Algorithms to Find Source-Code Clones”, In proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2004), June 21-24, 2004, Washington D.C., pp. 1079—1080

Teaching and Student Advising/Mentoring

Doctoral Students Currently Directing

- Amin Assereh, Project: “Mining Machine Learning Implementations for Successful Generalization to New Problems” (recruited from top Iranian University)
- Deborah Stoffer, Project: “Investigation of the Range of Behaviors of Chaos Automata”. Matriculated from M.S. program (M.S. Thesis Defended Sept. 2005)

Master’s Students Currently Directing

- Craig Stewert, Project area - Data Discovery with a focus on SNP data
- Kaveh Noorbakhsh, Project area - Medical Informatics (for improving emergency room outcomes)

Dissertation Committee Member (current and completed)

- Andrew Sutton – Computer Science, *Reverse Engineering To Support Generic Programming,* PhD Prospectus Spring 2008, Advisor: J Maletic
- Eric Chapman – Biological Sciences, Dissertation Title “*Investigation of the evolution of feeding strategies in the family Sciomyzidae*”, PhD Advisor: Randy Hough, Defense April 2, 2008
- Bonita Sharif, “*Empirical Assessment of UML Class Diagram Layouts based on Architectural Importance*”, PhD Prospectus Fall 2007, Advisor: J Maletic
- Julie A. Morris – Biological Sciences, Dissertation Title “*A Systematic Study of the Genus Lythrum and the Evolution of Heterostyly”* PhD Advisor: Andrea Schwarzbach, Defense May 17, 2007
- Jennifer Watson, Biological Sciences, “*Evolution of mtDNA in a freshwater bivalve lineage with doubly uniparental inheritance of mtDNA*”, PhD Advisor: Randy Hough, *in progress*
- Elena Khrosinkova – Physics, Dissertation Title “Elastic Electron Scattering from the Helium Isotopes at Large Momentum Transfers” [In Progress]
- Julie A. Morris – Biological Sciences, Dissertation Title “A Systematic Study of the Genus *Lythrum* and the Evolution of Heterostyly” [passed prospectus June 05, Defended Dissertation Summer 07]
- Ting Wu – Biological Sciences, Dissertation Title “Phylogenetic Systematics of Wolf Spider Genus Geolycosa (Araneae, Lycosidae) Based on Multiple Mitochondrial and Nuclear Gene Sequences” [passed prospectus Spring 05]
- John Gale – BMS/Cleveland Clinic Joint Ph.D. progam – (Ph.D. granted Spring 2004- currently a post-doctoral research fellow in the Eskandar lab at Harvard University Medical School)
- Eric Chapman – Biological Sciences, Dissertation Title “Investigation of the evolution of feeding strategies in the family Sciomyzidae” [passed prospectus Fall 03, Dissertation Defense Spring 08]

Theses Directed

- Deborah Stoffer “Alignment of Protein Secondary Structure Sequences“ M.S. Thesis, Kent State University, Kent, Ohio, (Degree granted Fall 2005 - currently working towards Ph.D. at Kent State University)
- Mahesh Tamboli, “*Computer Aided Image Analysis of Mobile Microscopic Objects*”, M.S. Thesis, Kent State University, Kent, Ohio, (Degree granted Fall 2004 – currently employed at Private Bioinformatics Research Institute in Wisconsin)

ROBERT A. WALKER

Professor & Department Chair

Computer Science Department
Kent State University
Kent, OH 44242
http://www.cs.kent.edu/~walker
walker@cs.kent.edu

PROFESSIONAL EDUCATION:

August 1988	**Doctor of Philosophy in Electrical and Computer Engineering**, Carnegie Mellon University, Pittsburgh, Pennsylvania
May 1983	**Master of Science in Electrical Engineering (Computer Engineering)**, Carnegie Mellon University, Pittsburgh, Pennsylvania
June 1981	**Bachelor of Science in Electrical Engineering**, Magna Cum Laude, Tennessee Technological University, Cookeville, Tennessee

MAJOR APPOINTMENTS (all at Kent State University)

July 2006 – present	**Department Chair** (4 year term, through June 2010)
July 2005 – June 2006	**Interim Department Chair**
August 2002 – June 2005	**Assistant Department Chair**
August 2002 – June 2005	**OBR Budget Coordinator**

AWARDS AND HONORS:

October 2006	Recognized as a *Distinguished Member* of the Association for Computing Machinery (ACM) "in recognition of significant accomplishments in, and impact on, the computing field"
June 2008	Received the *Outstanding Contribution to ACM Award* from the Association for Computing Machinery (ACM) "*for a sustained record of dedicated and conscientious leadership within the ACM Special Interest Groups, including service as Chair of the SIG Governing Board, Chair of SIGDA, SGB Representative to Council, as well as leadership in ACM conference organization*"
June 2006	Received the *Distinguished Service Award* from the Association for Computing Machinery Special Interest Group on Design Automation (ACM / SIGDA) "for dedicated service as SIGDA Chair (2001-2005) and over a decade of service to SIGDA, DAC, and the EDA profession"

PROFESIONAL SOCIETY SERVICE:

July 2005 – present	Past Chair, ACM / SIGDA
July 2001 – June 2005	Chairman, ACM / SIGDA
July 2006 – June 2008	Past Chair, SIG Governing Board (SGB)
July 2006 – present	SIG Governing Board (SGB) Representative to ACM Council
July 2006 – present	Member, ACM Council (as SGB Council Representative)
January 2006 – present	Chair, ACM Senior Member Committee
January 2006 – present	Member, ACM Distinguished Member Committee

PUBLICATIONS:

Journal Articles

"Using Hardware Multithreading to Overcome Broadcast/Reduction Latency in an Associative SIMD Processor", Kevin Schaffer and Robert A. Walker, *Parallel Processing Letters*, 18(4):491–509, December 2008.

Conference / Workshop Papers

"Multithreading in an Associative SIMD Processor", Kevin Schaffer and Robert A. Walker, in *Proc. of the 22nd International Parallel and Distributed Processing Symposium (Workshop on Large-Scale Parallel Processing)*, full text on CDROM. Miami, Florida, April 2008.

"Dynamic Round-Robin Task Scheduling to Reduce Cache Misses for Embedded Systems", Ken W. Batcher and Robert A. Walker, in *Proc. of the 2008 Design, Automation and Test in Europe Conference*, pp. 260–263. Munich, Germany, March 2008.

"A Prototype Multithreaded Associative SIMD Processor", Kevin Schaffer and Robert A. Walker, in *Proc. of the 21st International Parallel and Distributed Processing Symposium (Workshop on Advances in Parallel and Distributed Computing Models*, abstract on page 228, full text on CDROM. Long Beach, California, March 2007.

"Implementing a Multiple-Instruction Stream Associative MASC Processor", Hong Wang and Robert A. Walker, in *Proc. of the 18th International Conference on Parallel and Distributed Computing and Systems*, pages 460–465. Dallas, Texas, November 2006.

"Solving the Longest Common Subsequence (LCS) Problem using the Associative ASC Processor with Reconfigurable 2D Mesh", Sabegh Singh Virdi, Hong Wang, and Robert A. Walker, in *Proc. of the 18th International Conference on Parallel and Distributed Computing and Systems*, pages 454–459. Dallas, Texas, November 2006.

"Interrupt Triggered Software Prefetching for Embedded CPU Instruction Caches", Ken W. Batcher and Robert A. Walker, in *12th Real-Time and Embedded Technology and Applications Symposium*, pp. 91–100. San Jose, California, April 2006.

"A Scalable Pipelined Associative SIMD Array with Reconfigurable PE Interconnection Network for Embedded Applications", Hong Wang and Robert A. Walker, in *Proc. of the 17th International Conference on Parallel and Distributed Computing and Systems*, pp. 667-673. Phoenix, Arizona, November 2005.

"Cluster Miss Prediction with Prefetch on Miss for Embedded CPU Instruction Caches", Ken Batcher and Robert A. Walker, in *Proc. of the 7th International Conference on Compilers, Architecture, and Synthesis for Embedded Systems*, pp. 24-34. Washington DC, September 2004.

GRADUATE STUDENTS ADVISED:

Current Doctoral Advisees

Kenneth Atchinson	*Design and Implementation of an Associative SIMD Processor for Virus Detection*,
Kevin Schaffer	*Design and Implementation of a Multithreaded Associative SIMD Processor*

Doctoral Students Advised to Completion

Kenneth Batcher	*Cluster Miss Based Enhancement Techniques for Embedded CPI Instruction Caches*, defended April 2008 (currently employed by the Cisco)
Hong Wang	*Design and Implementation of an FPGA-Based Scalable Pipelined Associative SIMD Processor Array with Specialized Variations for Sequence Comparison and MSIMD Operation*, defended November 2006 (currently employed by the University of Toledo)

Masters Students Advised to Completion

Ping Xu	*Implementing Three VLDC String-Matching Algorithms on an FPGA-Based Associative SIMD Processor*, defended May 2006 (currently in KSU's Nursing program)
Sabegh Singh Virdi	*Solving the Longest Common Subsequence (LCS) Problem using the Associative ASC Processor with Reconfigurable 2D Mesh*, defended March 2006 (currently employed as a consultant)
Jalpesh Chitalia	*Efficient Representation of Data Structures on Associative Processors*, defended September 2004 (currently employed by PayPal)
Lei Xie	*Implementing a PE Interconnection Network for a FPGA-Based Associative Processor*, defended March 2004

Name: Paul S. Wnag
Current title: Professor and ICM Research Director
Educational background:
Ph.D., August 1971, Massachusetts Institute of Technology,
Cambridge, Massachusetts, USA

Brief employment history:
1986- Present: Director of Research, ICM, Kent State U. Kent, Ohio
1981- Present: Professor of Computer Science, Kent State U. Kent, Ohio
1977-81: Associate Professor, Kent State U. Kent, Ohio
1970-77: Research Member of Laboratory for Computer Science,
MIT, Cambridge, Massachusetts
1974-77: Assistant Professor, MIT; 1972-74 Instructor,
MIT, Cambridge, Massachusetts

List of representative publications, presentations, showings, honors, prizes, and awards.
Grants:
- National Science Foundation (NSF), Grant No. CCR-0201772, Internet Accessible Mathematical Computation (IAMC) and Web-based Mathematics Education (WME), Amt: $255,000, Date: August 1, 2002 through July 31, 2008
- National Science Foundation (NSF) Supplement to grant CCR 021772, IAMC 2005 Conference, Beijing China. Amt: $6,000, August 1, 2005.
- NSF REU (jointly with Professor Weidong Liao, Shepherd University, West Virginia) Web-based Mathematics Education, Amt: $6,000.00, 06/1/2005 to 05/30 2006.
- OBR Research Challenge, (jointly with Prof. Mikusa of the School of Education), Web-based Mathematics Education, Amt: $50,304, Jan. 2004 to Jan. 2005.
- National Science Foundation (NSF) Supplement to grant CCR 021772 (Internet Accessible Mathematical Computation). Amt: $4,800, August 1, 2003.
- NSF Grant No. CCR-9721343, Amt: $177,778, Title: Parallel/Distributed Symbolic Computation, Date: July 1998 to July 2003

Textbooks Published:
- Java with Object-Oriented and Generic Programming, E-book published by sofpower.com, ISBN 978-1-4276-3452-8, Sept. 2008
- Web Design and Programming, (with Sanda Katila) Brooks/Cole, 10-2003
- Java with Object-Oriented Programming, Brooks/Cole, 7-2002
- Standard C++ with Object-Oriented Programming, Brooks/Cole, 7-2000
- Java with OOP and Web Applications, Brooks/Cole, 9-1998
- An Introduction to UNIX with X and the Internet, PWS, 7-1996
- C++ with Object-oriented Programming, PWS, 1-1994
- An Introduction to ANSI C on UNIX, Wadsworth, 9-1991
- An Introduction to Berkeley UNIX, Wadsworth, 2-1988

Publications (articles):
- "DMAS: A Web-based Distributed Mathematics Assessment System" (with Saleh Al-shomrani) International Conference on Learning 2008, University of Illinois at Chicago, USA, June 3-6, 2008
- "WME: a Web-based Mathematics Education System for Teaching and Learning." (with M. Mikusa, S. Al-shomrani, X. Lai, X. Zou, D. Zeller) ICME 11 TSG 22 Theme 3 the 11the International Congress on Mathematical Education, Monterrey, Mexico, July 6 - 13, 2008.
- "An On-line MathML Editing Tool for Web Applications." (with Wei Su and Lian Li) Proceedings, 2nd International Multi-Symposiums on Computer and Computational Sciences, 2007, pp. 458-464.
- "GeoSVG: A Web-based Interactive Plane Geometry System for Mathematics Education." (with Xun Lai) Proceedings, the 2nd IASTED International Conference on EDUCATION AND TECHNOLOGY (ICET), July 17-19, 2006, pp. 5-10.

- "Design and Implementation of an Assessment Database for Mathematics Education." (with Saleh Alshomrani) Proceedings, the 2nd IASTED International Conference on EDUCATION AND TECHNOLOGY (ICET), July 17-19, 2006, pp. 173-179.
- "MEML: Supporting Structured, Interoperable and Dynamic Web-based Mathematics Education." (with Xiao Zou) Proceedings, the 2nd IASTED International Conference on EDUCATION AND TECHNOLOGY (ICET), July 17-19, 2006, pp. 113-120.
- "An Approach for Interoperable and Customizable Web-based Mathematics Education." (with David Chiu) Proceedings, The Fifth IASTED International Conference on Web-based Education, January 23-25, 2006 Puerto Vallarta, Mexico, pp. 80-87.
- "Lesson Page Structure and Customization in WME." (with Wei Su and Lian Li) IAMC 2005 Workshop, Chinese Academy of Sciences, July 24 2005, Beijing, China.
- "An SVG Based Tool for Plane Geometry and Mathematics Education," (with Xun Lai) IAMC 2005 Workshop, Chinese Academy of Sciences, July 24 2005, Beijing, China.
- "Building DMAD: A Distributed Mathematics Assessment Database for WME," (with Saleh Al-shomrani) Proceedings, IEEE SoutheastCon, Fort Lauderdale, Florida, April 2005, pp. 630-635.
- "Features and Advantages of WME: A Web-based Mathematics Education System," (with M. Mikusa, S. Al-shomrani, D. Chiu, X. Lai, and X. Zou) Proceedings, IEEE SoutheastCon, Fort Lauderdale, Florida, April 2005, pp. 621-629.

Masters and Dissertation students supervised

Ph.D--Saleh Al-shomrani (2008), Weidong Liao (2003), Simon Gray (1998), Iyad A. Ajwa (1998), Olaf Bachmann (1996), Yaser Doleh (1995), Mohamed Rayes (1995), Ken Weber (1994), Naveen Sharma (1992), Vilmar Travisan (1991), Hui Tan (1986).

Joint-Ph.D. with Lanzhou University, PRC--Su Wei

Masters-- Maja Anderson, Ashish Bhargava, Dan Bennett, David Chrin, David Chiu, Yaser Doleh, Adnan Eshaque, Barbara Gates, Robert Hall, Chia-Kai Hsu, Chao-Jen Hsu, Chokchai Leangsuksun, Sam Lin, Carl Powell, Mohamed Rayes, Naveen Sharma, Cora Stackelberg, Trevor Tan, Linlin Tong, Vilmar Trevisan, Sanjiva, Weerawarana, Carl Williams, David Wu, Douglas Young, Pei Young, Rick Zhang.

List of committees served and role, and professional service activities.

Committees:

- Kent State Patent Board Member
- CAC rep, College of Arts and Sciences
- Chair and memvber, graduate studies committee, CS Department
- Member, Curriculum committee, CS Department
- Member, Industrial Liason committee, CS Department

Professional Activities:

- Invited keynote speaker, East Cost Computer Algebra Day, Shepherd University, Shepherdstown, West Virginia, Saturday, May 10, 2008.
- WME Tutorial Workshop by P. Wang and the WME Team Shepherd University, Shepherdstown, West Virginia, May 9, 2008.
- Improvements and new implementation for MathPASS, a Web-based drill-and-practice system for remedial mathematics, by P. Wang, Su Wei and Dean Zeller in cooperation with the Department of Mathematical Sciences, started Spring 2008.
- External Examiner, Ph.D. Committee, University of Western Ontario, London, Ontario, Canada, August 2007.
- Member Ohio Mathematics and Science Coalition (OMSC), since 2006.
- WME Tutorial Workshop by P. Wang and X. Lai at the 4th International Conference on Web-based Learning (ICWL 2005), Hong Kong SAR, China, July 31, 2005.
- Program committee, International Symposium on Symbolic and Algebraic Computation, Beijing China, July 2005.

Ye Zhao — Biosketch

Education

August 2006	**Ph.D., Computer Science** Stony Brook University, Stony Brook, New York
July 2000	**M.Sc., Computer Science** Tsinghua University, Beijing, China
July 1997	**B.Eng., Computer Science** Tsinghua University, Beijing, China

Employment

Assistant Professor August 2006-Present
Department of Computer Science, Kent State University

Publications

Journal Papers

1. Zhe Fan, Yu-chuan Kuo, Ye Zhao, Feng Qiu, Arie Kaufman, Bill Arcieri, **Visual Simulation of Thermal Shock in a Pressurized Water Reactor**, *The Visual Computer, International Journal of Computer Graphics*, Springer, To appear, 2009.
2. Ye Zhao, **Lattice Boltzmann based PDE solver on the GPU**, *The Visual Computer, International Journal of Computer Graphics*, volume 24, number 5, pages 323-333, May, 2008.
3. Ye Zhao, Yiping Han, Zhe Fan, Feng Qiu, Yuchuan Kuo, Arie Kaufman and Klaus Mueller, **Visual Simulation of Heat Shimmering and Mirage**, *IEEE Transactions on Visualization and Computer Graphics, Vol 13, No 1, pages 179-189*, January/February, 2007
4. Ye Zhao, Lujing Wang, Feng Qiu, Arie Kaufman and Klaus Mueller, **Melting and Flowing in Multiphase Environment**, *Computers&Graphics, Vol 30, No 4, pages 519-528*, August, 2006.

Conference Papers

1. Jamal Alsakran, Ye Zhao and Xinlei Zhao, **Tile-based parallel coordinates and its application in financial visualization**, *Proceedings of the Visual and Data Analysis conference at IS&T/SPIE Electronic Imaging 2010*, To appear, San Jose, CA, Jan, 2010, SPIE.
2. Aaron Hagan, Ye Zhao, **Parallel 3D Image Segmentation of Large Data Sets on a GPU Cluster**, *Proceedings of the 5th International Symposium on Visual Computing, Lecture Notes in Computer Science*, To appear, Dec, 2009, Springer.
3. Fan Chen, Ye Zhao, **Distance Field Transform with an Adaptive Iteration Method**, *Proceedings of IEEE International Conference on Shape Modeling and Applications (SMI)*, pages 111-118, Beijing, China, June, 2009, IEEE.
4. Jamal Alsakran, Ye Zhao and Xinlei Zhao, **Visual Analysis of Mutual Fund Performance**, *Proceedings of the 13th International Conference on Information Visualization*, pages 252-259, Barcelona, Spain, July, 2009, IEEE Computer Society.
5. Ye Zhao, **GPU-Accelerated Surface Denoising and Morphing with Lattice Boltzmann Scheme**, *Proceedings of IEEE International Conference on Shape Modeling and Applications (SMI)*, pages 19-28, Stony Brook, NY, June, 2008.
6. Ye Zhao, Feng Qiu, Zhe Fan and Arie Kaufman, **Flow Simulation with Locally-Refined LBM**, In *Proceedings of ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 181-188, Seattle, WA, May, 2007.

Book Chapters

1. Arie Kaufman, Ye Zhao, **Visual Simulation of Flow**, In *Scientific Visualization: Challenges for the Future, Dagstuhl Research Seminar Series*, Springer, To appear, 2009.

Tutorials and Courses

1. Ye Zhao, Arie Kaufman, Klaus Mueller, Ulrich Rüde, Nils Thuerey, **Interactive Lattice-Based Flow Simulation and Visualization**, Organizer, In *IEEE Visualization Conference*, Columbus, OH, October, 2008

Colloquium and Invited Talks

KSU spring 2009 Science Symposium
Introduction to visual simulation, analysis and acceleration, Kent, July 2009

Colloquium Talk, School of Software, Tsinghua University
Title: "Lattice Boltzmann Method and Its Usage in Computer Graphics and Visualization", Beijing, China, June, 2008

Invited Talk, First Visualization Workshop, Peking University
Title: "Modeling Natural Phenomena with Lattice Boltzmann Method", Beijing, China, June, 2008

Conference Paper Presentations

The 13th International Conference on Information Visualization
Title: "Visual Analysis of Mutual Fund Performance", Barcelona, Spain, July 2009

IEEE International Conference on Shape Modeling and Applications
Title: "Distance Field Transform with an Adaptive Iteration Method", Beijing, China, June 2009

IEEE International Conference on Shape Modeling and Applications
Title: "GPU-accelerated surface denoising and morphing with lattice Boltzmann scheme", Stony Brook, NY, June 2008

ACM Symposium on Interactive 3D Graphics and Games
Title: "Flow Simulation with Locally-Refined LBM", Seattle, WA, May 2007

Awards

NSF Grant, Title: *HCC:Small:FlowBase: A Realtime Simulation System of Turbulent Fluids Driven by Flow Pattern Database*, Single PI: Ye Zhao, Agency: NSF IIS-0916131, Duration: 3 Years, Date: 07/15/09 - 07/14/12, Amount: $391,689.
Kent State University, Academic Year Research and Creative Activity Appointment, 2008-2009

Professional Service

NSF Proposal Panelist, June, 2009
IEEE International Conference on Shape Modeling and Application, Program Committee, 2009
Center for Material Informatics, Kent State, Fellow, 2009
Visualization Workshop at Peking University, Program Committee, 2008
IADIS International Conference on computer Graphics and Visualization, Program Committee, 2007, 2008

Advisees

PhD Student: Jamal Alsakran, Fan Chen, Aaron Hagan, Zakia Tamimi, Zhi Yuan
Masters Student: Hiroki Morimoto, Ryan O'Neill
Undergraduate Student: Nick Dragan
Visiting Scholar: Nan Geng
Master committee: Brad Bellomo, Xiaoxi Du, Aaron Hagan, Xin Li, Fan Chen

Section C8:
FACULTY NAME - last name, first name
PROP - number of proposals submitted (both funded and unfunded)
AWARD - number of awards
AWARD $ - total award amount (direct plus indirect for the entire funding period)
STUDENT YEARS SUPPORTED- number of full academic years of graduate student support stipends from external awards.

FACULTY NAME	PROPOSALS	AWARDS	AWARD $	CS Portion of Award	CS STUDENT YEARS SUPPORTED
Baker, Johnnie	2	0	0	0	0
Bansal, Arvind	1	1	69,000	37,066	.5 RA
Batcher, Kenneth	1	1	10,000	6,000	0
Breitbart, Yrui	2	0	0	0	0
Dragan, Feodor	4	0	0	0	0
Farrell, Paul	4	3	573,522	142,504	.5 RA 1 Sem
Guercio, Angela	1	0	0	0	0
Jin, Ruoming	14	1	50,000	25,000	1 RA for 1 semester; 2 Student Workers for 11 wks
Khan, Javed	5	0	0	0	0
Lu, Cheng-Chang	1	1	69,000	31,934	.5 RA
Maletic, Jonathan	15	4	852,425	452,484	28, 1 year ugrad tuition scholarships; 1 RA for 2 years
Melton, Austin	8	1	400,000	100,000	0
Nesterenko, Mikhail	6	1	0	0	0
Peyravi, Hassan	5	0	0	0	0
Ruttan, Arden	4	3	573,522	142,504	.5 RA 1 Sem
Volkert, L. Gwenn	5	1	60,000	19,800	0
Walker, Robert	6	5	2,314,815	2,308,815	* See Below
Wang, Paul	11	1	50,304	25,152	0
Zhao, Ye	7	1	391,689	391,689	2 RAs for 3 years

*21 Tuition & Stipends AY; 7 Tuition only AY; 7 Tuition & Stipends 1 semester; 4 Tuition only 1 semester; 11% of 5 Tuition & Stipends AY; 11% of 3 Tuition and Stipends 1 semester

Appendix E

Section E1:

A represenative listing of student publications

1. Feodor F. Dragan, Irina Lomonosov: On compact and efficient routing in certain graph classes. Discrete Applied Mathematics 155(11): 1458-1470 (2007)
2. Feodor F. Dragan, Chenyu Yan, Irina Lomonosov: Collective tree spanners of graphs. SIAM J. Discrete Math. 20(1): 241-260 (2006)
3. Feodor F. Dragan, Irina Lomonosov: On Compact and Efficient Routing in Certain Graph Classes. ISAAC 2004: 402-414
4. C. Yan, Y. Xiang and F.F. Dragan, Compact and Low Delay Routing Labeling Scheme for Unit Disk Graphs, *The Algorithms And Data Structures Symposium (WADS 2009)*, Banff Conference Centre, Banff, Alberta, Canada, 21-23 August 2009, Springer, *Lecture Notes in Computer Science*, 5664, pp. 566–577.
5. F. F. Dragan, Y. Xiang, "How to use spanning trees to navigate in graphs", Proc. of 34st International Symposium on Mathematical Foundations of Computer Science (MFCS 2009), High Tatras, Slovakia, August 24-28, 2009, pp. 282-294.
6. Dan Bennett, Paul A. Farrell, Cara Stein, A Chromium Based Viewer for CUMULVS , The Proceedings of the 2006 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'06), pp 472--477, (2006)
7. Dan Bennett, Paul Farrell, and Arden Ruttan, An Extensible Parameter Viewer for CUMULVS , Proceedings of PACISE 20O9, 10--15, (2009)
8. Cartesian Contour: A Concise Representation for a Collection of Frequent Sets, Ruoming Jin, Yang Xiang, and Lin Liu, in the Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'09), pages 417-426 (Full Paper, Long Presentation, Acceptance rate: 10%) .
9. Migration Motif: A spatial-temporal pattern mining approach for financial market, Xiaoxi Du, Ruoming Jin, Ling Ding, Victor Lee, and John Thornton, in the Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'09), pages1135-1144 (Industry/Application Track, Full Paper, Short Presentation, Acceptance rate: 27%).
10. Mingming Lu, Qiyu Zhang, Wayne Cheng, Cheng-Chang Lu, "Retrieval of Multimedia Objects using Color Segmentation and Dimension Reduction Features," International Conference on Image and Graphics, Xi'an, China, Sep 2009.
11. Wayne Cheng and Cheng-Chang Lu, "Acceleration of Medical Image Registration using GPU in Computing Normalized Mutual Information," International Conference on Image and Graphics, Xi'an, China, Sep 2009.
12. Sutton, A., Holeman, R., Maletic, J. I., (2009), "Abstracting the Template Instantiation Relation in C++", in the Proceedings of the IEEE International Conference on Software Maintenance (ICSM'09), Edmonton, Canada, September 20-26, pp. 559-562. (*21% acceptance of full, 39% acceptance of full & short papers*)
13. Dragan, N., Collard, M.L., Maletic, J. I., (2009), "Using Method Stereotype Distribution as a Signature Descriptor for Software Systems", in the Proceedings of the IEEE International Conference on Software Maintenance (ICSM'09), Edmonton, Canada, September 20-26, pp. 567-570. (*21% acceptance of full, 39% acceptance of full & short papers*)
14. P. N. Gedela, P. Kosaraju, and A. Melton, "Measurement Theory Principles in Software Measurement", Proceedings of the 16th International Conference on Software Engineering and Data Mining, M. Al-Mubaid and M. Garbey -- editors, ISBN 978-1-880843-63-5, Las Vegas, Nevada, 2007, pages 344-348.
15. Vikash Sananda and Austin Melton, "Decision and Implementation Models for Outsourcing", The Global Studies Journal
16. Kehinde O. Jolayemi and Austin Melton, "Software Measurement Needs Its Own Theory", Proceedings of The 2008 International Conference on Software Engineering Research and Practice, SERP 2008, Volume II, Hamid R.
17. R. Nor, M. Nesterenko and P. Lavrentyev "Oxybuoy: Constructing a Real-Time Inexpensive Hypoxia Monitoring Platform", *International Workshop on Advanced Sensor Integration Technology (ASIT 2009)*, to appear;
18. T. Clouser, M. Miyashita, M. Nesterenko "Fast Geometric Routing with Concurrent Face Traversal", *12th International Conference on Principles of Distributed Systems (OPODIS 2008)*, pp. 346-362, Luxor, Egypt, December 2008;

19. Y Drabu and H. Peyravi. An adaptive bandwidth control algorithm for IP routers. In *International Conference on Communications in Computing (CIC)*, pages 311–317, Las Vegas, USA, June 2004.
20. Y. Drabu and H. Peyravi. Fault tolerant routing in tri-sector wireless cellular mesh networks. In *International Conference Parallel and Distributed Computing Systems*, pages 209–216, San Francisco, California, USA, September 2006.
21. Y. Drabu and H. Peyravi. Gateway placement with QoS constraints in wireless mesh networks. In *IEEE International Conference on Networking*, pages 46–51, Cancun, Mexico, April 2008. IEEE Computer Society.
22. Y. Drabu and H. Peyravi. Fault recovery with qos provisioning in wireless mesh networks. *International Conference on Wireless Networks (ICWN'09)*, pages 1–8, July 2009.
23. Saleh Al-shomrani and Paul Wang. ``Design and Implementation of an Assessment Database for Mathematics Education", Proceedings, the 2nd IASTED International Conference on EDUCATION AND TECHNOLOGY (ICET), July 17-19, 2006, pp. 173-179.
24. S. Al-shomrani and P. S. Wang. ``Building DMAD: A Distributed Mathematics Assessment Database for WME", Proceedings of the IEEE Southeast Conference, IEEE, Ft. Lauderdale, FL, April 8-10 2005, pp. 630-635.
25. Wei Su, Paul S. Wang, Lian Li An On-line MathML Editing Tool for Web Applications 2nd International Multi-Symposiums on Computer and Computational Sciences 2007 (IMSCCS|07), The University of Iowa, Iowa City, Iowa, USA, pp. 458-464.
26. Xun Lai and Paul Wang. ``GeoSVG: A Web-based Interactive Plane Geometry System for Mathematics Education", Proceedings, the 2nd IASTED International Conference on EDUCATION AND TECHNOLOGY (ICET), July 17-19, 2006, pp. 5-10.
27. Xiao Zou and Paul Wang. ``MEML: Supporting Structured, Interoperable and Dynamic Web-based Mathematics Education", Proceedings, the 2nd IASTED International Conference on EDUCATION AND TECHNOLOGY (ICET), July 17-19, 2006, pp. 113-120.
28. David Chiu and Paul Wang. ``An Approach for Interoperable and Customizable Web-based Mathematics Education", Proceedings, the Fifth IASTED International Conference on Web-based Education, Jan. 23-25, 2006, Puerto Vallarta, Mexico, pp. 80-87.
29. Jamal Alsakran, Ye Zhao and Xinlei Zhao, Tile-based parallel coordinates and its application in financial visualization, *Proceedings of the Visual and Data Analysis conferenceat IS&T/SPIE Electronic Imaging 2010*, To appear, San Jose, CA, Jan, 2010, SPIE.
30. Aaron Hagan, Ye Zhao, Parallel 3D Image Segmentation of Large Data Sets on a GPU Cluster, *Proceedings of the 5th International Symposium on Visual Computing, Lecture Notes in Computer Science*, To appear, Dec, 2009, Springer.
31. Fan Chen, Ye Zhao, Distance Field Transform with an Adaptive Iteration Method, *Proceedings of IEEE International Conference on Shape Modeling and Applications (SMI)*, pages 111-118, Beijing, China, June, 2009, IEEE.
32. Jamal Alsakran, Ye Zhao and Xinlei Zhao, Visual Analysis of Mutual Fund Performance, *Proceedings of the 13th International Conference on Information Visualization*, pages 252-259, Barcelona, Spain, July, 2009, IEEE Computer Society.
33. Vibha Tripathi,YuriBreitbart Replica Management in Content Distribution Networks IADIS International Journal on WWW/Internet, Vol 4, No. 1, pp. 43-54, 2006
34. Yuri Breitbart and Hassan Gobjuka, Characterization of Layer 2 Unique Topologies,Information Proccessing Letters, 2008.
35. Ruoming Jin, Yuri Breitbart and Chibuike Muoh, Data Discretization Unification, Knowledge and Information Systems International Journal,(invitedpaper),SpringerVerlag,29pages, vol. 19,No. 1,April
36. Yuri Breitbart, Feodor Dragan, Hassan Gobjuka Effective Monitor Placement in Internet Networks, Journal Of Networks,(to appear) 2009.
37. Y. Breitbart, Deepakraj Shathilal "Relational Network Manager for IP Networks", Proceedings of International Conference on Internet Computing, 2004 Las Vegas, USA
38. Y.Breitbart,F.Dragan,H.Gobjuka"EffectiveNetworkMonitoring", Proceedings of the 13th International Conference on Computer Communications and Networks (ICCCN'04), Chicago, October 2004.
39. Fuat Akal, Yuri Breitbart, Torsten Grabs, Hans-Jorg Schek, Can Turker, Lourens Veen "Fine-Grained Replication, and Scheduling with Freshness and Correctness Guarantees",Proceedings of the 31th International Conference on VLDB, Trondheim, Norway, August 2005
40. Yuri Breitbart, Vibha Tripathi "Replica Management in Content Distribution Networks" Proceedings of IADIS Conference, Lisbon, Portugal, 2005

41. Hassan Gobjuka,Yuri Breitbart ”Characterization of Layer-2 Unique Topologies in Multisubnet Local Area Networks”, Proceedings of LCN Conference,Tampa,Florida,2006(poster paper)
42. Yuri Breitbart,Hassan Gobjuka ”Ethernet Topology for Networks with Incomplete Information” Proceedings of theInternational Conference on Computer Communication(ICCCN),2007,Honolulu,Hawaii
43. Hassan Gobjuka,Yuri Breitbart, ”Discovering Network Topology of large Multisubnet Ethernet Networks”, Proceedings of the 32d Annual IEEE Conference on Local Computer Networks, 2007, Dublin, Ireland.
44. Ruoming Jin, Yuri Breitbart, Chibuke Muoh, ”Data Discretization Unification”, Proceedings of International Conference on Data Mining, 2007, Omaha NE.
45. Ruoming Jin, Scott McCallen, Yuri Breitbart, David Fuhry, Dong Wang ”Estimating the Number of Frequent Itemsets in a Large Database”, Proceedings of EDBT 2009, Saint Petersburg, Russia.
46. L. Voicu, H. Schuldt, Y. Breitbart, and H.-J. Schek ”Replicated Data Management in the GRID: The ReGRIDiT Approach”, Proceedings of International Workshop on Data Grids for e-Science, 2009
47. L. Voicu, Fuat Akal, Y. Breitbart, H. Schuldt, and H.-J. Schek ”Re:GRIDiT -Coordinated Distributed Update Transactions on Replicated Data in the Grid”, Proceedings of GRID 2009 Conference, Banff, Canada 2009
48. Ruoming Jin, Yuri Breitbart, and Rong Li ”A Tree-based Framework for Differences Summarization” Proceedings of ICDM 2009 Conference, Miami, Florida 2009
49. “Using Hardware Multithreading to Overcome Broadcast/Reduction Latency in an Associative SIMD Processor”, Kevin Schaffer and Robert A. Walker, *Parallel Processing Letters*, 18(4):491–509, December 2008.
50. “Multithreading in an Associative SIMD Processor”, Kevin Schaffer and Robert A. Walker, in *Proc. of the 22nd International Parallel and Distributed Processing Symposium (Workshop on Large-Scale Parallel Processing)*, full text on CDROM. Miami, Florida, April 2008.
51. “Dynamic Round-Robin Task Scheduling to Reduce Cache Misses for Embedded Systems”, Ken W. Batcher and Robert A. Walker, in *Proc. of the 2008 Design, Automation and Test in Europe Conference*, pp. 260–263. Munich, Germany, March 2008.
52. “A Prototype Multithreaded Associative SIMD Processor”, Kevin Schaffer and Robert A. Walker, in *Proc. of the 21st International Parallel and Distributed Processing Symposium (Workshop on Advances in Parallel and Distributed Computing Models*, abstract on page 228, full text on CDROM. Long Beach, California, March 2007.
53. “Implementing a Multiple-Instruction Stream Associative MASC Processor”, Hong Wang and Robert A. Walker, in *Proc. of the 18th International Conference on Parallel and Distributed Computing and Systems*, pages 460–465. Dallas, Texas, November 2006.
54. “Solving the Longest Common Subsequence (LCS) Problem using the Associative ASC Processor with Reconfigurable 2D Mesh”, Sabegh Singh Virdi, Hong Wang, and Robert A. Walker, in *Proc. of the 18th International Conference on Parallel and Distributed Computing and Systems*, pages 454–459. Dallas, Texas, November 2006.
55. “Interrupt Triggered Software Prefetching for Embedded CPU Instruction Caches”, Ken W. Batcher and Robert A. Walker, in *12th Real-Time and Embedded Technology and Applications Symposium*, pp. 91–100. San Jose, California, April 2006.
56. “A Scalable Pipelined Associative SIMD Array with Reconfigurable PE Interconnection Network for Embedded Applications”, Hong Wang and Robert A. Walker, in *Proc. of the 17th International Conference on Parallel and Distributed Computing and Systems*, pp. 667-673. Phoenix, Arizona, November 2005.
57. “Cluster Miss Prediction with Prefetch on Miss for Embedded CPU Instruction Caches”, Ken Batcher and Robert A. Walker, in *Proc. of the 7th International Conference on Compilers, Architecture, and Synthesis for Embedded Systems*, pp. 24-34. Washington DC, September 2004.

Section E2:

A representative listing of student presentations

Irina Lomonosov:

- *12/04* 15th Annual International Symposium on Algorithms and Computation (ISAAC 2004), HKUST, Hong Kong

Chenyu Yan:

- *06/06* SIAM Conference on Discrete Mathematics, Victoria, BC, Canada
- 05/07 MIGHTY, Dayton, Ohio

Yang Xiang:

- 08/09 WADS 2009, Banff, Alberta, Canada
- 07/09 ACM SIGMOD Conference, Providence, Rhode Island
- 08/08 ACM SIGKDD Conference, Las Vegas, Nevada
- 05/07 MIGHTY, Dayton, Ohio

Dan Bennett:

- PACISE'O6
- PDPTA'06
- PACISE 20O9

Sutton, A.:

- “Abstracting the Template Instantiation Relation in C++”, at the IEEE International Conference on Software Maintenance (ICSM 2009), Edmonton, Canada, September 20-26.
- "How we Manage Portability and Configuration with the C Preprocessor", at the IEEE 23rd International Conference on Software Maintenance (ICSM 2007), Paris, France, Oct. 2-4.
- "Mappings for Accurately Reverse Engineering UML Class Models from C++ ", at the 12th IEEE Working Conference on Reverse Engineering (WCRE 2005), Pittsburgh, PA USA, Nov. 7-11.

Hammad, M.:

- “Automatically Identifying Changes that Impact Code-to-Design Traceability”, at the 16th IEEE International Conference on Program Comprehension (ICPC 2009), Vancouver, BC, Canada, May 17-19.

Kagdi, H.:

- "An Approach to Mining Call-Usage Patterns with Syntactic Context", at ACM/IEEE International Conference on Automated Software Engineering (ASE 2007), Atlanta, GA, Nov 5-7.
- "Combining Single-Version and Evolutionary Dependencies for Software-Change Prediction", at the 4th ACM International Workshop on Mining Software Repositories (MSR 2007), Minneapolis, MN, May 19-20.
- "Comparing Approaches to Mining Source Code for Call-Usage Patterns", at the 4th ACM International Workshop on Mining Software Repositories (MSR 2007), Minneapolis, MN, USA, May 19-20.
- "Software Repositories: A Source for Traceability Links", at the 4th ACM International Workshop on Traceability in Emerging Forms of Software Engineering (GCT/TEFSE 2007), Lexington, KY, Mar. 22-23.
- "Software-Change Prediction: Estimated+Actual", at the 2nd International IEEE Workshop on Software Evolvability, Philadelphia (SE 2006), Pennsylvania USA, September 24.
- "Mining for Co-Changes in the Context of Web Localization", at the 8th IEEE International Symposium on Web Site Evolution (WSE 2006), Philadelphia, Pennsylvania USA, September 23-24.

Sharif, B.:

- "Using Fine-Grained Differencing to Evolve Traceability Links", at the 4th ACM International Workshop on Traceability in Emerging Forms of Software Engineering (GCT/TEFSE 2007), Lexington, KY, Mar. 22-23.

Yusuf, S.:

- "Mining Sequences of Changed-files from Version Histories", at the ACM International Workshop on Mining Software Repositories (MSR 2006), Shanghai China, May 22-23.

Collard, M. L.:

- "Supporting Source Code Difference Analysis", at the 20th IEEE International Conference on Software Maintenance (ICSM 2004), Chicago, Illinois, September 11-17.

Drabu, Y.:

- An adaptive bandwidth control algorithm for IP routers. In *International Conference on Communications in Computing (CIC)*, pages 311–317, Las Vegas, USA, June 2004.

- Fault tolerant routing in tri-sector wireless cellular mesh networks. In *International Conference Parallel and Distributed Computing Systems*, pages 209–216, San Francisco, California, USA, September 2006.
- Gateway placement with QoS constraints in wireless mesh networks. In *IEEE International Conference on Networking*, pages 46–51, Cancun, Mexico, April 2008. IEEE Computer Society.
- Fault recovery with Qos provisioning in wireless mesh networks. *International Conference on Wireless Networks (ICWN'09)*, pages 1–8, July 2009.

Jamal Alsakran:

- The 13th International Conference on Information Visualization Title: Visual Analysis of Mutual Fund Performance", Barcelona, Spain, July 2009

Fan Chen:

- IEEE International Conference on Shape Modeling and Applications Title: Distance Field Transform with an Adaptive Iteration Method", Beijing, China, June 2009

Kevin Schaffer:

- "Multithreading in an Associative SIMD Processor", in *Proc. of the 22nd International Parallel and Distributed Processing Symposium (Workshop on Large-Scale Parallel Processing)*, full text on CDROM. Miami, Florida, April 2008.
- "A Prototype Multithreaded Associative SIMD Processor", in *Proc. of the 21st International Parallel and Distributed Processing Symposium (Workshop on Advances in Parallel and Distributed Computing Models*, abstract on page 228, full text on CDROM. Long Beach, California, March 2007.

Ken W. Batcher:

- "Interrupt Triggered Software Prefetching for Embedded CPU Instruction Caches", in *12th Real-Time and Embedded Technology and Applications Symposium*, pp. 91–100. San Jose, California, April 2006.
- "Dynamic Round-Robin Task Scheduling to Reduce Cache Misses for Embedded Systems", in *Proc. of the 2008 Design, Automation and Test in Europe Conference*, pp. 260–263. Munich, Germany, March 2008.
- "Cluster Miss Prediction with Prefetch on Miss for Embedded CPU Instruction Caches", in *Proc. of the 7th International Conference on Compilers, Architecture, and Synthesis for Embedded Systems*, pp. 24-34. Washington DC, September 2004.

Hong Wang:

- "Implementing a Multiple-Instruction Stream Associative MASC Processor", in *Proc. of the 18th International Conference on Parallel and Distributed Computing and Systems*, pages 460–465. Dallas, Texas, November 2006.
- "A Scalable Pipelined Associative SIMD Array with Reconfigurable PE Interconnection Network for Embedded Applications", in *Proc. of the 17th International Conference on Parallel and Distributed Computing and Systems*, pp. 667-673. Phoenix, Arizona, November 2005.

Sabegh Singh Virdi:

- "Solving the Longest Common Subsequence (LCS) Problem using the Associative ASC Processor with Reconfigurable 2D Mesh", in *Proc. of the 18th International Conference on Parallel and Distributed Computing and Systems*, pages 454–459. Dallas, Texas, November 2006.

Section E3:
A representative listing of externally – awarded student prizes and awards

Sutton, A.:

- User Friendly Graphs Their Measures. Google Summer of Code, **$5,000**, May – Aug, 2007.
- Next Generation Generic Graph Library. Google Summer of Code, **$5,000**, May – Aug, 2007.

Wagner, C.:

- Enhancing Boost Graph Library. Google Summer of Code, **$5,000**, May – Aug, 2009.

Lopez, M.:

- Enhancing Boost Graph Library. Google Summer of Code, **$5,000**, May – Aug, 2009.

Zakia Tamimi:

- Fulbright Grant, 2008, 2009, the United States Department of State.