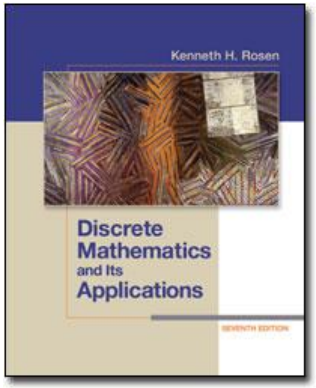
Discrete Structures for Computer Science

Feodor F. Dragan
Spring 2012

Textbook

Discrete Mathematics and Its Applications
By Kenneth H. Rosen, McGraw Hill (7th ed.)



Use lecture notes as study guide.

Course Requirements

- Attendance 5%
- Quizzes 20%
- Homework 20%
- Midterm Exam 25%
- Extra Credit Problem 2-5%
- Final Exam 30%

Why Discrete Math?

Design efficient computer systems.

- How did Google manage to build a fast search engine?
- What is the foundation of internet security?

algorithms, data structures, database,
parallel computing, distributed systems,
cryptography, computer networks…

Logic, sets/functions, counting, graph theory…

What is discrete mathematics?

Logic: artificial intelligence (AI), database, circuit design

Counting: probability, analysis of algorithm

Graph theory: computer network, data structures

Number theory: cryptography, coding theory

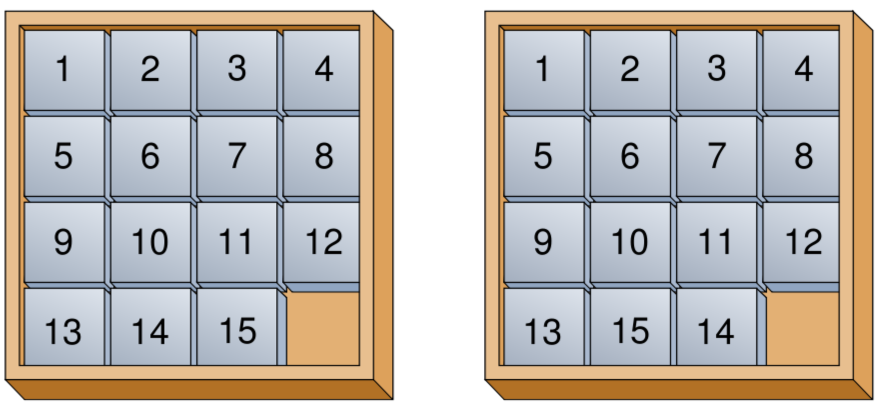
logic, sets, functions, relations, etc

Topic 1: Logic and Proofs

How do computers think?

Logic: propositional logic, first order logic

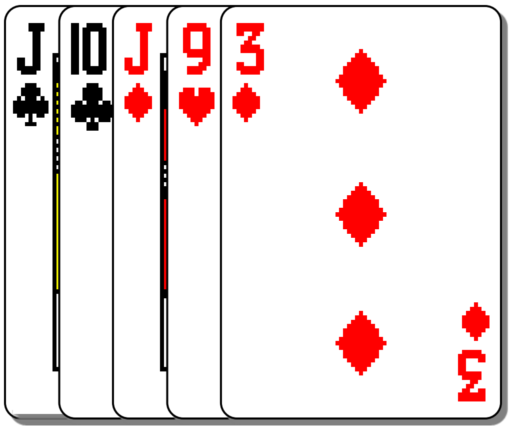
Proof: induction, contradiction



Artificial intelligence, database, circuit, algorithms

Topic 2: Counting

- Sets
- Combinations, Permutations, Binomial theorem
- Functions
- Counting by mapping, pigeonhole principle
- Recursions, generating functions



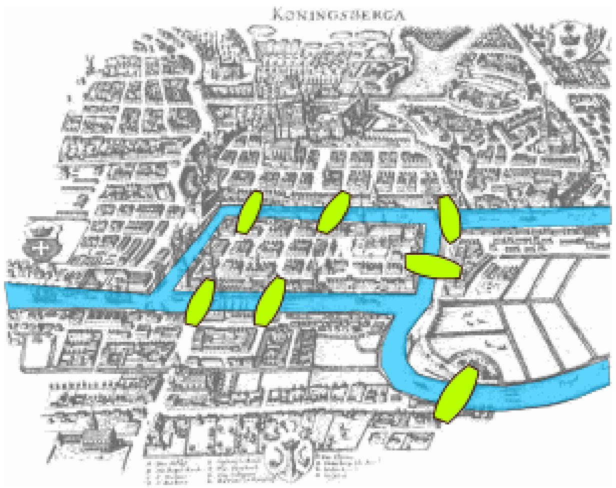
Probability, algorithms, data structures

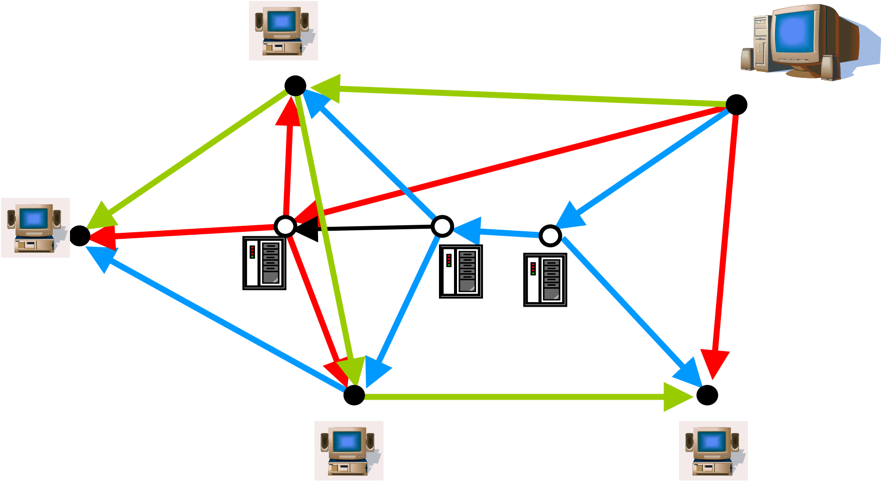
Topic 2: Counting

How many steps are needed to sort n numbers?

Topic 3: Graph Theory

- Relations, graphs
- Degree sequence, isomorphism, Eulerian graphs
- Trees

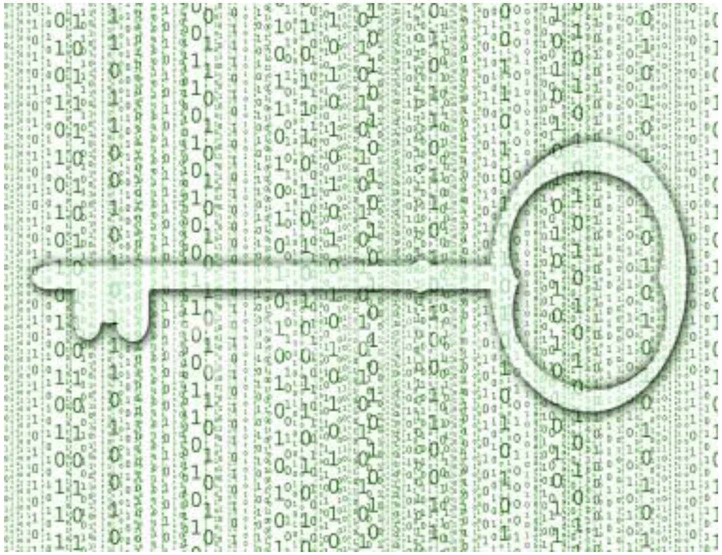




Computer networks, circuit design, data structures

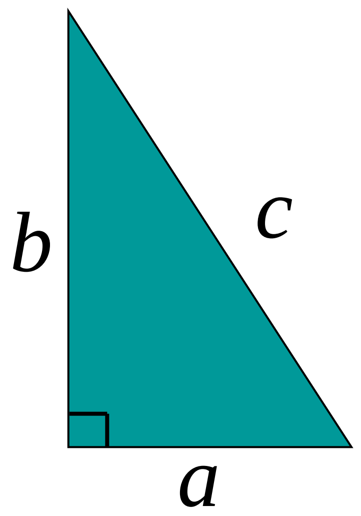
Topic 4: Number Theory

- Number sequence
- Euclidean algorithm
- Prime number
- Modular arithmetic



Cryptography, coding theory, data structures

Pythagorean theorem

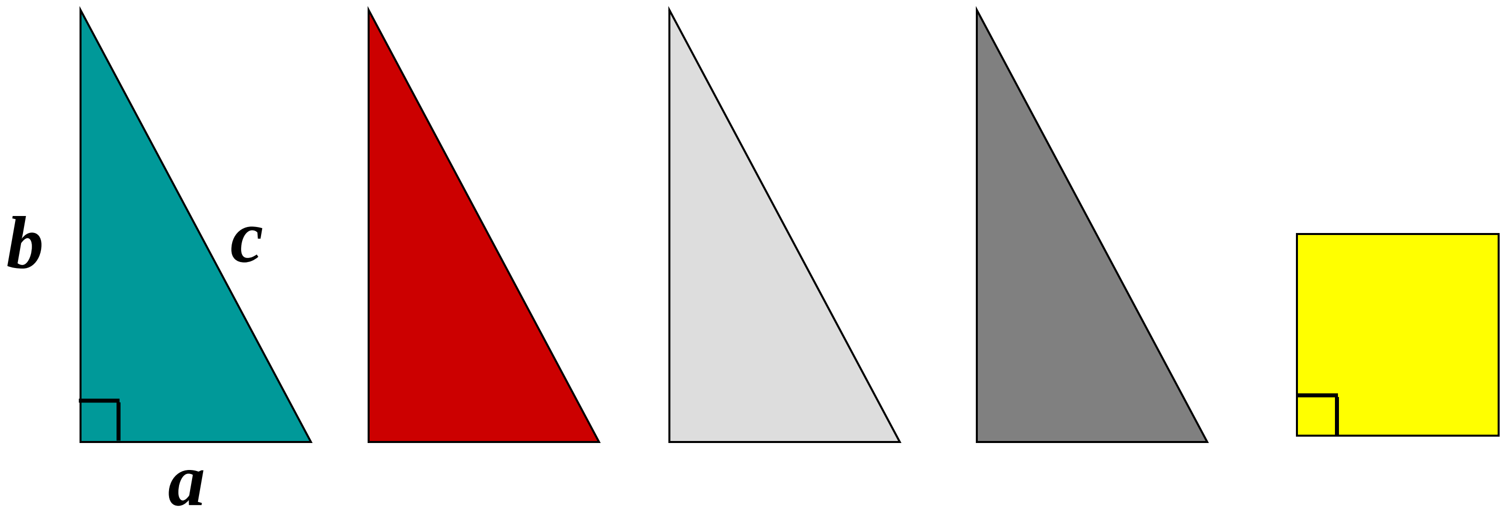


$$a^2 + b^2 = c^2$$

Familiar?

Obvious?

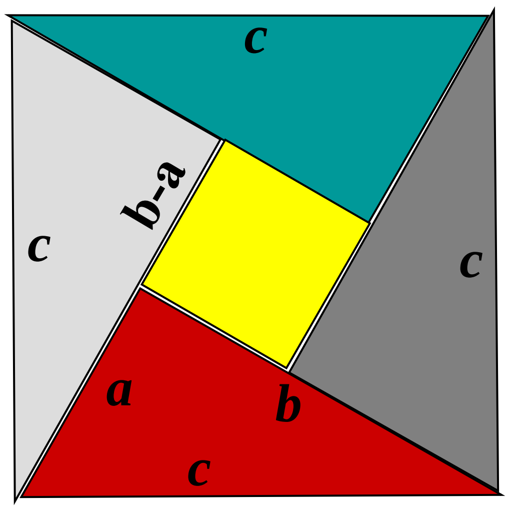
Good Proof



Rearrange into: (i) a c×c square, and then

(ii) an a×a & a b×b square

Good Proof



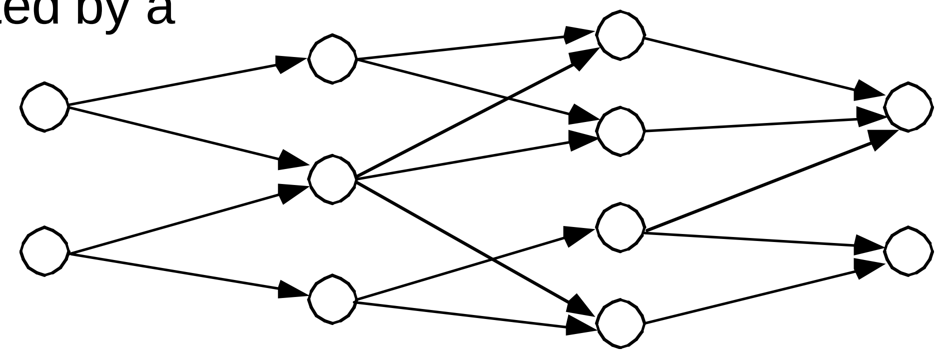
81 proofs in http://www.cut-the-knot.org/pythagoras/index.shtml

Acknowledgement

- Next slides are adapted from ones created by Professor Bart Selman at Cornell University.

Graphs and Networks

•Many problems can be represented by a graphical network representation.



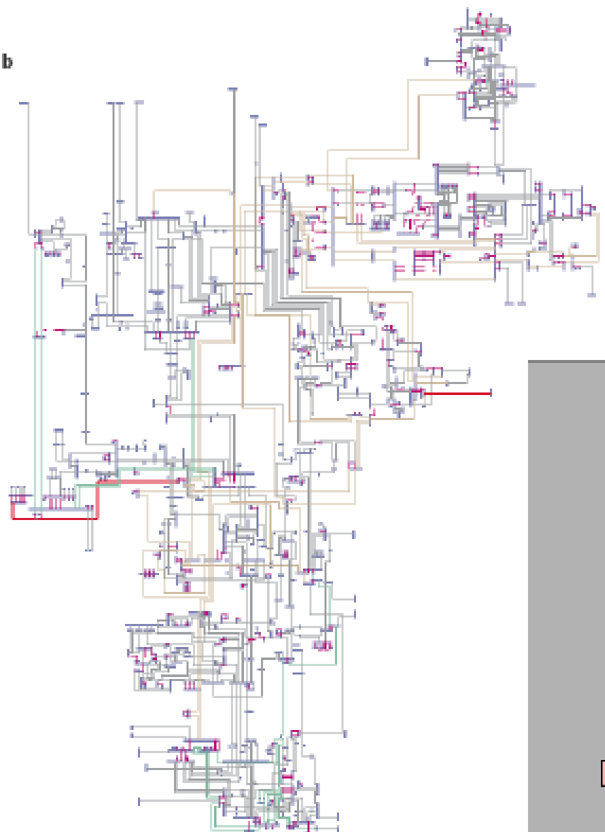
•Examples:

- Distribution problems
- Routing problems
- Maximum flow problems
- Designing computer / phone / road networks
- Equipment replacement
- And of course the Internet

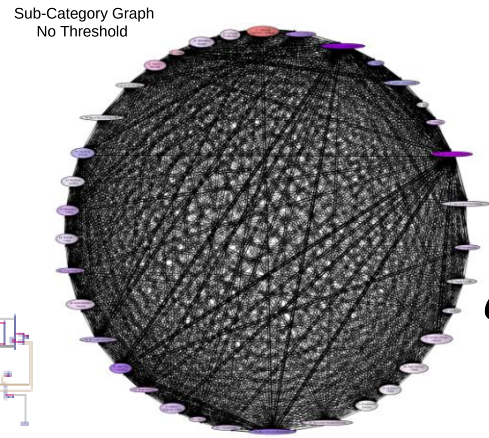
<u>Aside</u>: finding the right problem representation is one of the key issues.

New Science of Networks

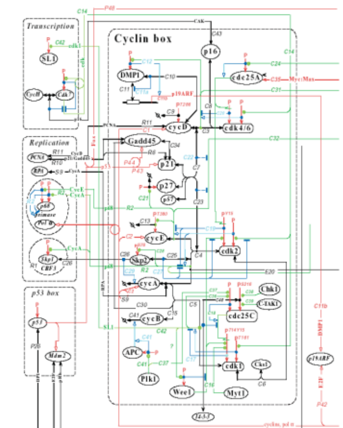
Networks are pervasive



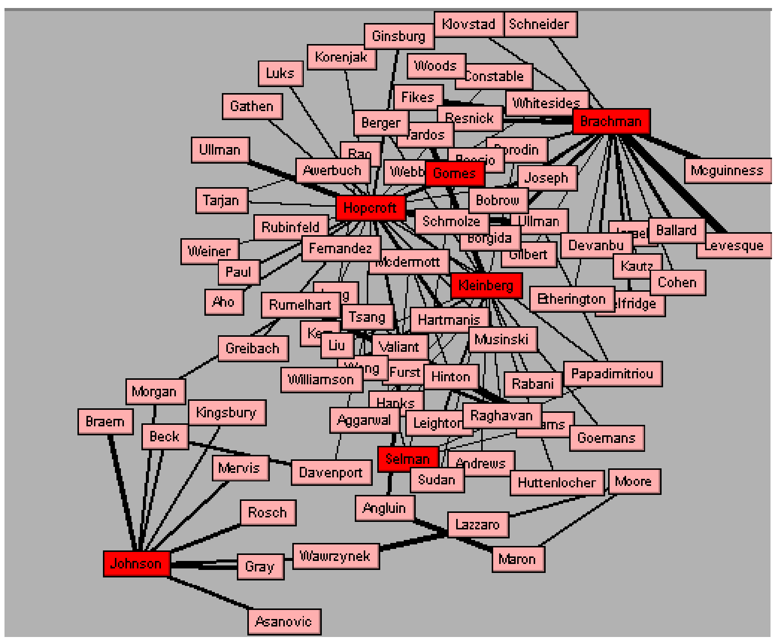
NYS Electric Power Grid (Thorp,Strogatz,Watts)



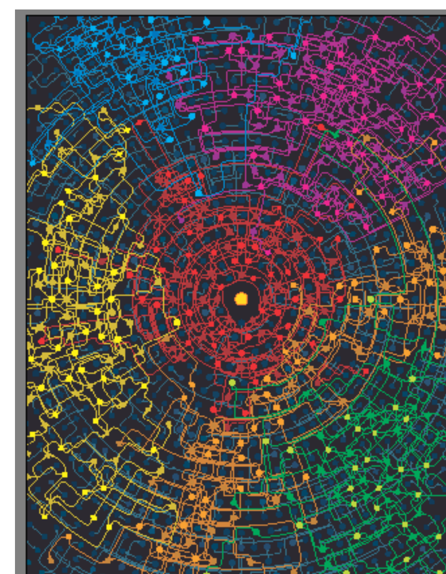
Utility Patent network 1972-1999 (3 Million patents) Gomes,Hopcroft,Lesser,Selman



Neural network of the nematode worm C- elegans (Strogatz, Watts)

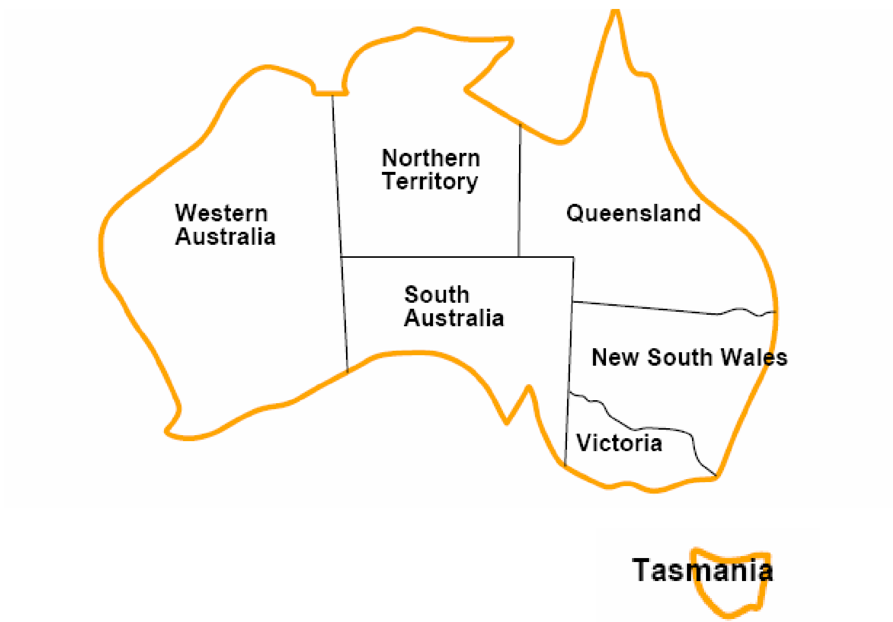


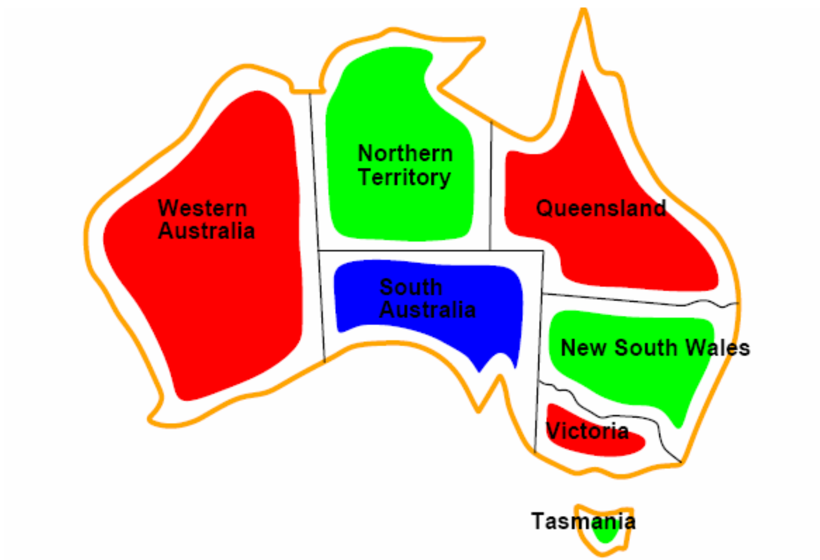
Network of computer scientists ReferralWeb System (Kautz and Selman)



Cybercommunities (Automatically discovered) Kleinberg et al

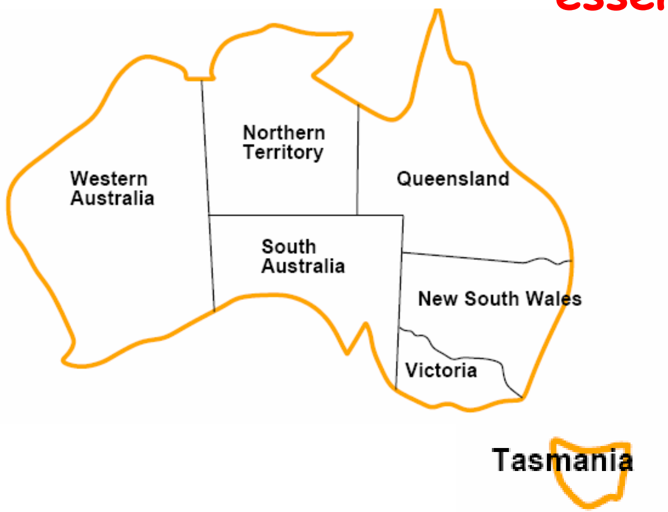
Example: Coloring a Map



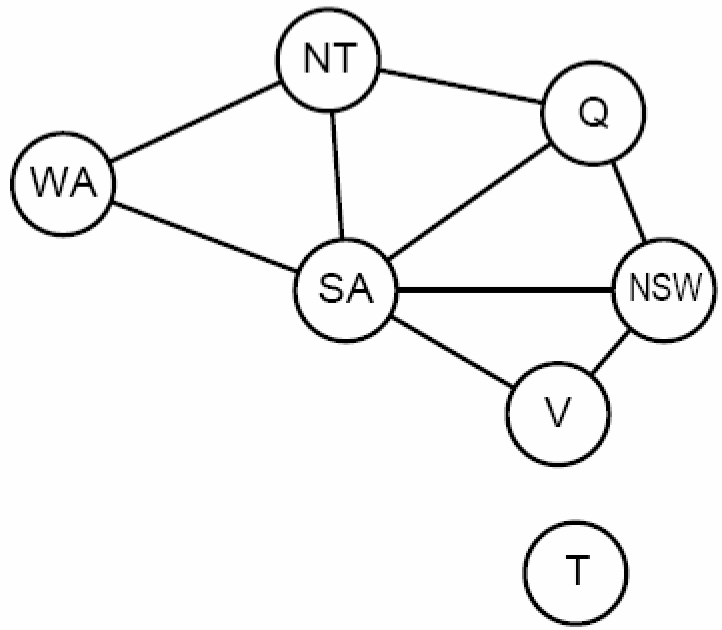


How to color this map so that no two adjacent regions have the same color?

Graph representation

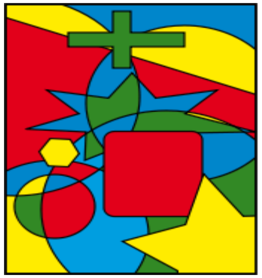


Abstract the essential info:



Coloring the nodes of the graph:
What's the minimum number of colors such that any two nodes connected by an edge have different colors?

Four Color Theorem



Four color map.

- The chromatic number of a graph is the least number of colors that are required to color a graph.

- The Four Color Theorem – ***the chromatic number of a planar graph is no greater than four. (quite surprising!)***

- Proof by Appel and Haken 1976;
- careful case analysis performed by computer;
- Proof reduced the infinitude of possible maps to 1,936 reducible configurations (later reduced to 1,476) which had to be checked one by one by computer.
- The computer program ran for hundreds of hours. The first significant *computer-assisted* mathematical proof. *Write-up was hundreds of pages including code!*

Examples of Applications of Graph Coloring

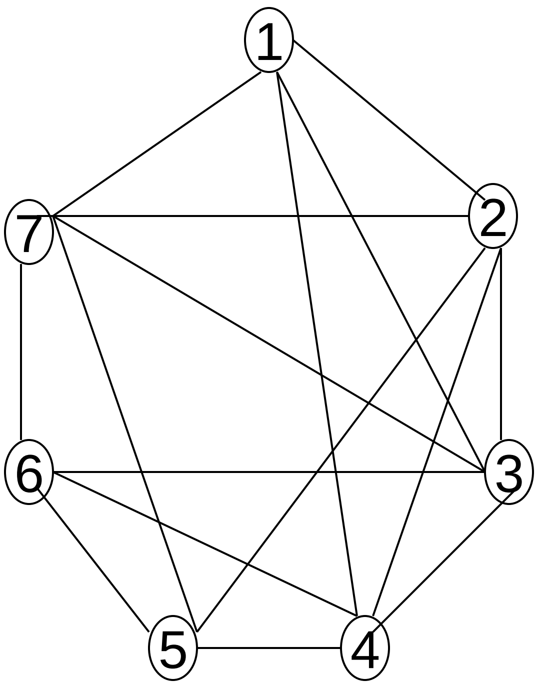
Scheduling of Final Exams

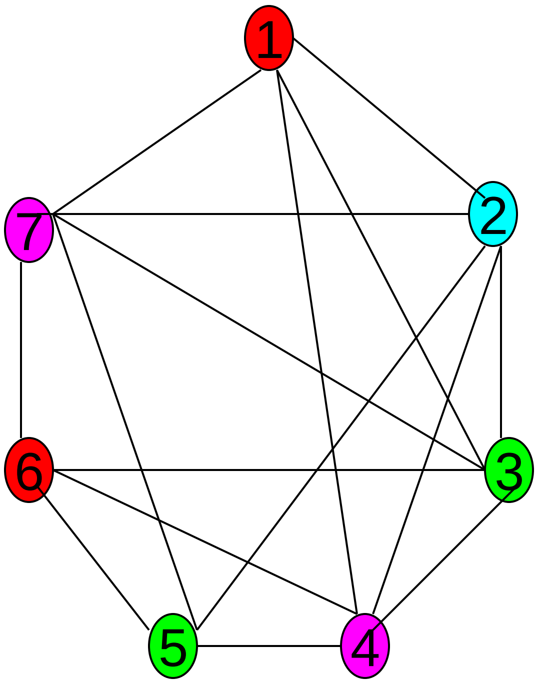
- How can the final exams at Kent State be scheduled so that no student has two exams at the same time? ***(Note not obvious this has anything to do with graphs or graph coloring!)***

Graph:
A vertex correspond to a course.
An edge between two vertices denotes that there is at least one common student in the courses they represent.
Each time slot for a final exam is represented by a different color.

A coloring of the graph corresponds to a valid schedule of the exams.

Scheduling of Final Exams





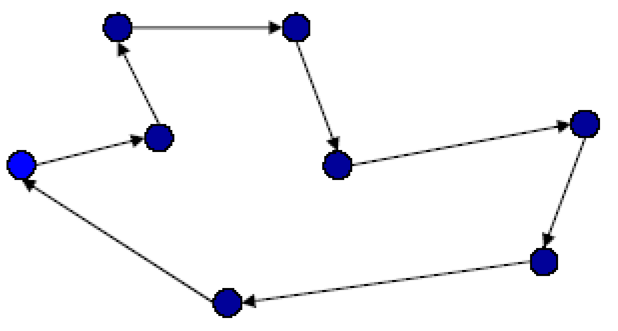
Time Period	Courses
I	1,6
II	2
III	3,5
IV	4,7

Why is mimimum number of colors useful?

What are the constraints between courses?
Find a valid coloring

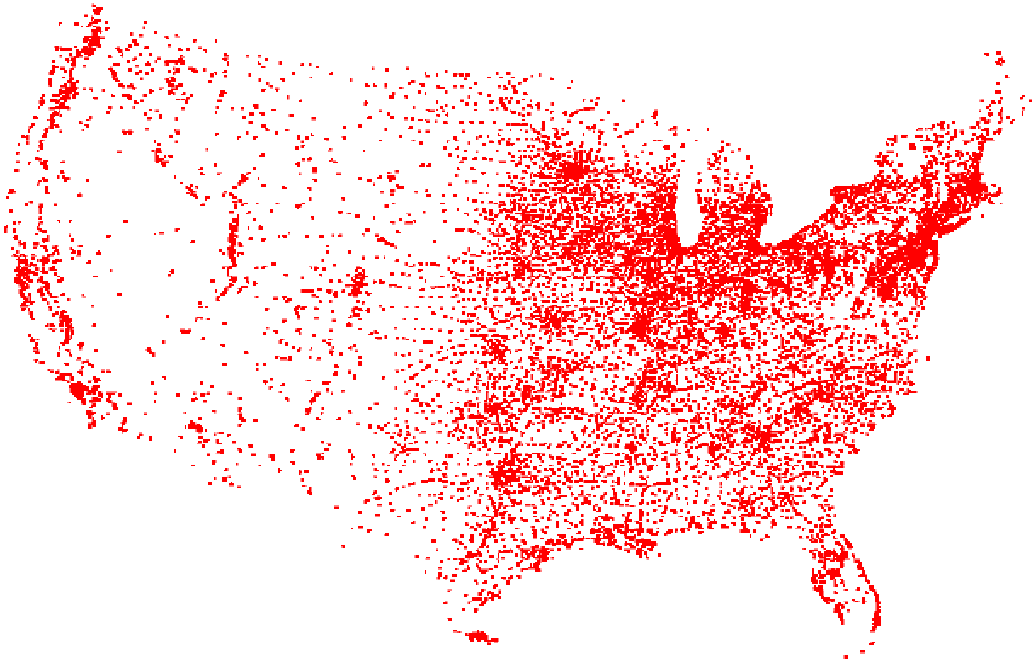
Example 2: Traveling Salesman

Find a closed tour of minimum length visiting all the cities.



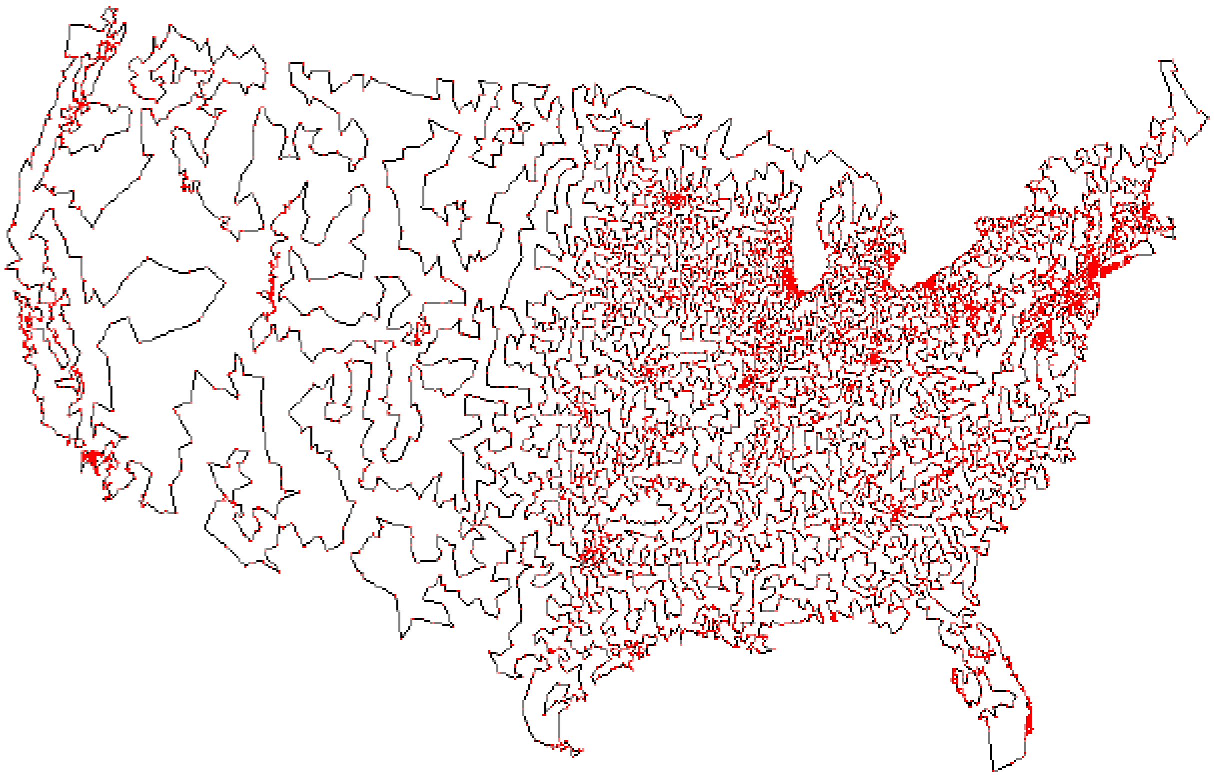
TSP → lots of applications:
Transportation related: scheduling deliveries
Many others: e.g., Scheduling of a machine to drill holes in a circuit board ;
Genome sequencing; etc

13,509 cities in the US



13508!= 1.4759774188460148199751342753208e+49936

13509 cities in the USA



(Applegate, Bixby, Chvatal and Cook, 1998)

The optimal tour!