Design and Analysis of Algorithms

This Class

Website and Contact

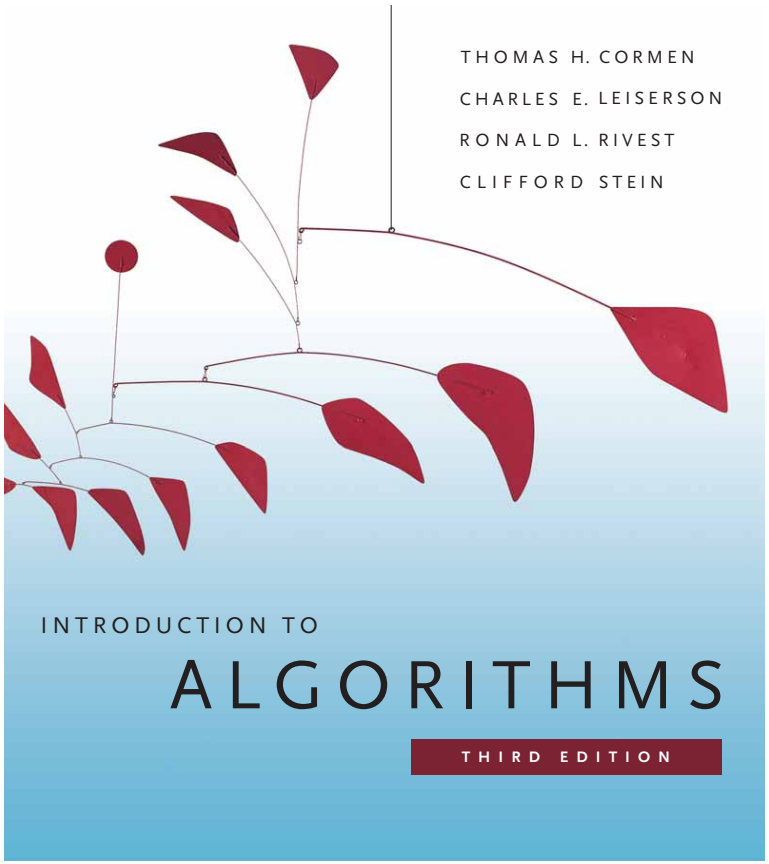
Website

- `www.cs.kent.edu/~aleitert/daa/`
- Important information
- Slides
- Announcements

Email

- `aleitert@cs.kent.edu`

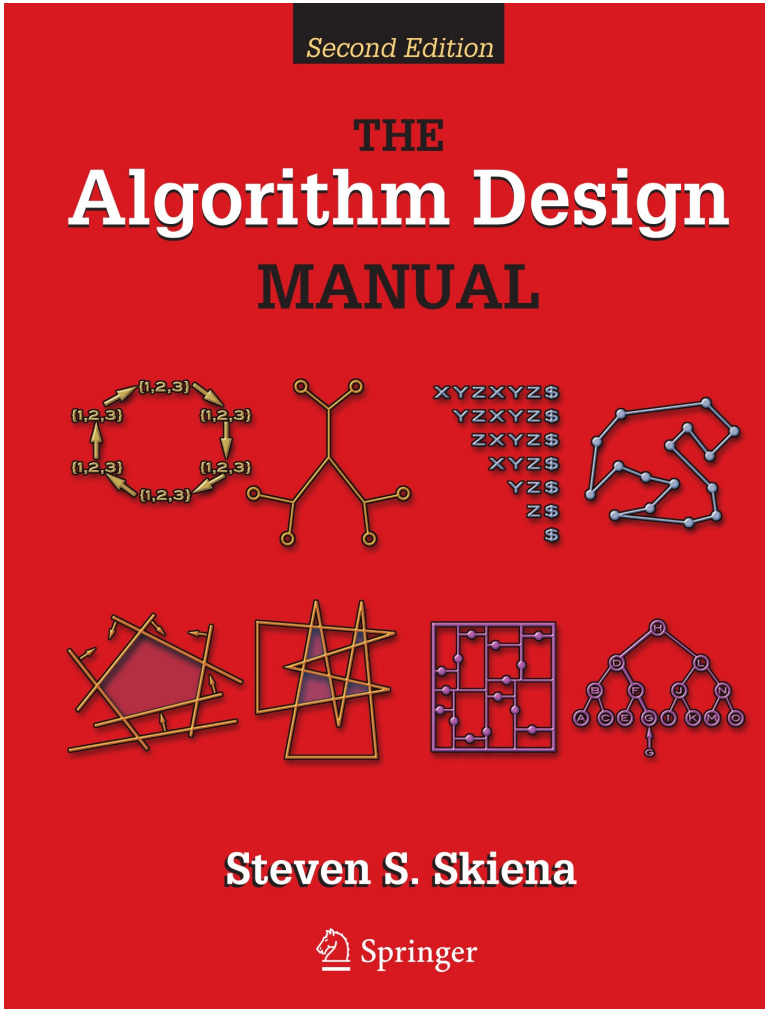
Primary Textbook



Introduction to Algorithms,
by Cormen et al.
3rd edition, MIT Press, 2009

Primary source for this class.

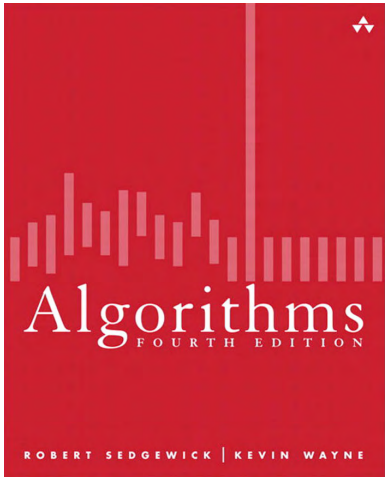
Other Textbooks



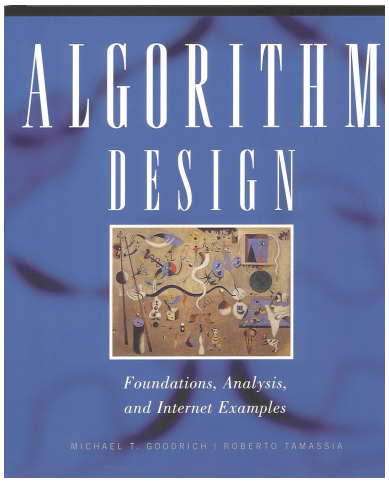
The Algorithm Design Manual,
by Steven S. Skiena
2nd edition, Springer, 2008

PDF-Version available for free at
Springer Link

Other Textbooks



Algorithms, 4th Edition,
by Robert Sedgewick and Kevin Wayne
4th edition, Addison-Wesley Professional, 2011



Algorithm Design: Foundations, Analysis, and Internet Examples,
by Michael T. Goodrich and Roberto Tamassia, 1st edition, Wiley, 2001

Clarification

Clarification

You do not need (to buy) a textbook. These are recommendations if you are looking for a textbook to study.

Course Requirements

Exam 1	33.3 %	June 30, during class
Exam 2	33.3 %	July 19, during class
Exam 3	33.3 %	Aug 4, during class

(Dates may change.)

Exams

- closed book examination
- one handwritten sheet (*one side*) allowed

Homework

- Will not be graded.
- Good preparation for exams.

Office Hours

On appointment
Room 352, Math and CS Building

Send me an email if you want to meet.

Academic Presence Verification

Due to federal rules, instructors “must verify that students begin attendance in each course for which they are registered.”

Required to receive federal financial aid.

Verification of *Your* Attendance

- Sign attendance sheet *once*.

Algorithms

Algorithm

Question

What is an *algorithm*?

Algorithm

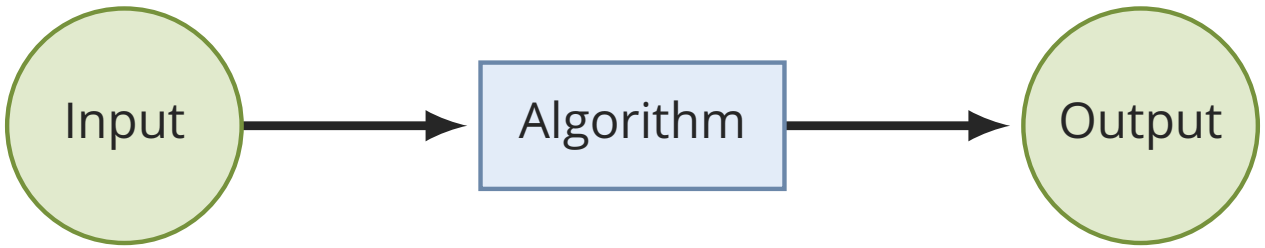
Question

What is an *algorithm*?

Wikipedia

> *An algorithm is a self-contained step-by-step set of operations to be performed. [...] An algorithm is an effective method that can be expressed within a finite amount of space and time [...] for calculating a function. Starting from an initial state and initial input, the instructions describe a computation that [...] proceeds through a finite number of well-defined successive states, eventually producing "output" and terminating at a final ending state.*

Algorithm



Properties of Algorithms

Correctness

- Will it produce the desired output?
- We will prove that our algorithms are correct.

Efficiency

- How fast is the algorithm?
- How much resources does it need?
- Is there a faster algorithm?

Having one of both properties is (usually) easy. However, having both is the goal.

Example

Problem

Finding doubles

You are given two integer arrays A and B. Is there an integer i which is in both arrays?

Algorithm 1

```
1 For Each a ∈ A
2     For Each b ∈ B
3         If a = b Then
4             Return "Yes"
5 Return "No"
```

Algorithm 2

```
1  Sort A and B.
2  Set i := 0 and j := 0.
3  While i < |A| and j < |B|
4      If A[i] = B[j] Then
5          Return "Yes"
6      Else If A[i] < B[j] Then
7          Set i := i + 1.
8      Else If A[i] > B[j] Then
9          Set j := j + 1.
10 Return "No"
```

Question

Question

Which algorithm is better and why?