Advanced Algorithms

Master's degree in Computer Science

University of Padova

Motivation

Look around you. Computers and networks are everywhere, enabling an intricate web of complex human activities: education, commerce, entertainment, research, manufacturing, health management, human communication, even war. Of the two main technological underpinnings of this amazing proliferation, one is obvious: the breathtaking pace with which advances in microelectronics and chip design have been bringing us faster and faster hardware.

This book tells the story of the other intellectual enterprise that is crucially fueling the computer revolution: efficient algorithms. It is a fascinating story.

—Dasgupta, Papadimitriou, and Vazirani: *Algorithms*

Motivation (continued)

Not so long ago, anyone who had heard the word "algorithm" was almost certainly a computer scientist or mathematician. With computers having become prevalent in our modern lives, however, the term is no longer esoteric. If you look around your home, you'll find algorithms running in the most mundane places: your microwave oven, your washing machine, and, of course, your computer. [...] The word "algorithm" appears somewhere in the news seemingly every day.

Therefore, it behooves you to understand algorithms not just as a student or practitioner of computer science, but as a citizen of the world.

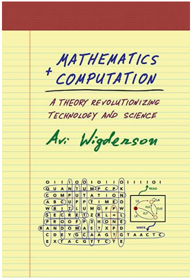
—Cormen, Leiserson, Rivest, and Stein: *Introduction to Algorithms*

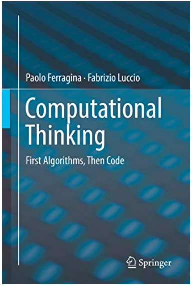
Motivation (continued)

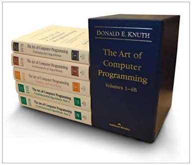
Algorithms are the heart and soul of computer science. Their applications range from network routing and computational genomics to public-key cryptography and machine learning. Studying algorithms can make you a better programmer, a clearer thinker, and a master of technical interviews.

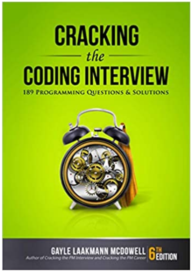
—Tim Roughgarden, Columbia University

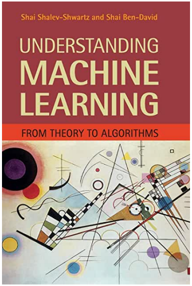
Motivation (continued)

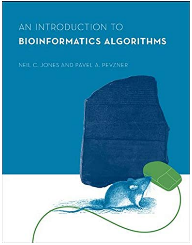




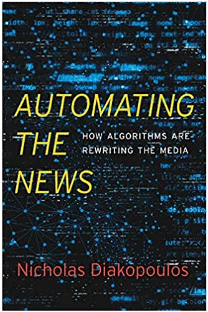


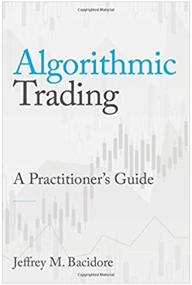


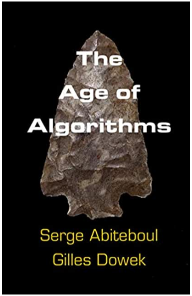


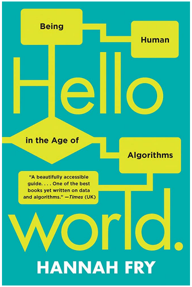


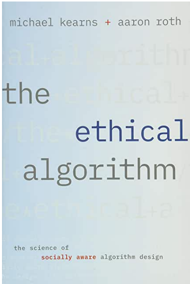
Motivation (continued)

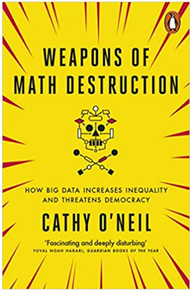












Motivation (continued)

…and that's why Advanced Algorithms is **mandatory**. Hence, you should give it the highest priority in this semester!

Contents

1. Graph algorithms
   - Graph search and its applications, minimum spanning trees, shortest paths, maximum flows

Contents

1. Graph algorithms
   - Graph search and its applications, minimum spanning trees, shortest paths, maximum flows
2. Approximation algorithms
   - Intractable problems: NP-hardness and reductions between problems
   - Approximation algorithms for intractable problems such as vertex cover, set cover, and the traveling salesperson problem

Contents

1. Graph algorithms
   - Graph search and its applications, minimum spanning trees, shortest paths, maximum flows
2. Approximation algorithms
   - Intractable problems: NP-hardness and reductions between problems
   - Approximation algorithms for intractable problems such as vertex cover, set cover, and the traveling salesperson problem
3. Randomized algorithms
   - Main techniques (e.g., Chernoff bounds) and applications to problems such as sorting and minimum cuts

Goals

You'll learn how to:

- Design and analyze algorithms for complex domains such as graphs
- Recognize NP-hard problems and address them using approximation algorithms
- Use the power of randomness to design fast algorithms for complex problems, and analyze them
- Analyze algorithms with respect to correctness and efficiency
- Formulate real-life problems as algorithmic problems and solve them

Prerequisites

Although there are no formal prerequisites, an undergraduate course in algorithms and a good knowledge of (discrete) probability are assumed. Specifically, you should be familiar with:

- Algorithm design techniques: divide and conquer, greedy, dynamic programming
- Data structures: lists, stacks, queues, binary trees, search trees, heaps
- Probability: basic notions, discrete random variables

Course Style

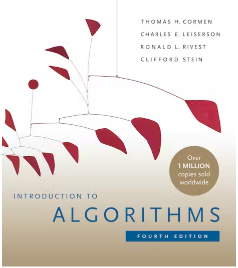
We'll discuss the intuition behind formulating an algorithmic solution and distill the core ideas that make the algorithm work. However, we'll never sacrifice rigorousness.

> *When you write an algorithm you need to have a proof that it's correct; an algorithm without a proof is a conjecture, it's not a theorem. And if you are proving things, well, that means mathematics.*[1]
>
> —Laslie Lamport, Turing Award 2013

[1] https://www.youtube.com/watch?v=rkZzg7Vowao

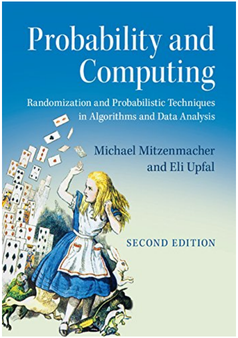
Course Style (continued)

Textbooks



Cormen, Leiserson, Rivest, and Stein
Introduction to Algorithms
4th edition
MIT Press, 2022

- The leading algorithms textbook
- 1.000.000+ copies sold worldwide
- Translated into more than a dozen languages



Mitzenmacher and Upfal
Probability and Computing
2nd edition
Cambridge University Press, 2017

- Standard reference for randomized algorithms and probabilistic analysis

Lecture Time & Place

24 lectures:

- Mondays 12:30–14:10, room 1A150
- Thursdays 12:30–14:10, room 1A150

Moodle

The only source of course information. You'll find:

- Announcements
- Log of lectures
- Lecture notes
- Examples of exam exercises
- Further readings

Exam

Written test, 2 hours. It comprises:

- 2-3 theory questions: questions on the topics presented in class
- 2 problems: exercises aimed at verifying the student's ability to use notions learned during the course to solve new problems

Instructor

Michele Scquizzato

- https://www.math.unipd.it/~scquizza/
- scquizza@math.unipd.it
- Room 4BC2, 4th floor
- Office hours: by appointment
- Research area: algorithms