Algorithms

FOURTH EDITION

ROBERT SEDGEWICK | KEVIN WAYNE

http://algs4.cs.princeton.edu

ALGORITHMS, PARTS I AND II

▸ *overview*

▸ *why study algorithms?*

▸ *resources*

Course overview

What is this course?

- Intermediate-level survey course.

- Programming and problem solving, with applications.

- Algorithm: method for solving a problem.

- Data structure: method to store information.

topic	data structures and algorithms	
data types	stack, queue, bag, union-find, priority queue	
sorting	quicksort, mergesort, heapsort	part 1
searching	BST, red-black BST, hash table	
graphs	BFS, DFS, Prim, Kruskal, Dijkstra	
strings	radix sorts, tries, KMP, regexps, data compression	part 2
advanced	B-tree, suffix array, maxflow	

Why study algorithms?

Their impact is broad and far-reaching.

Internet. Web search, packet routing, distributed file sharing, ...

Biology. Human genome project, protein folding, ...

Computers. Circuit layout, file system, compilers, ...

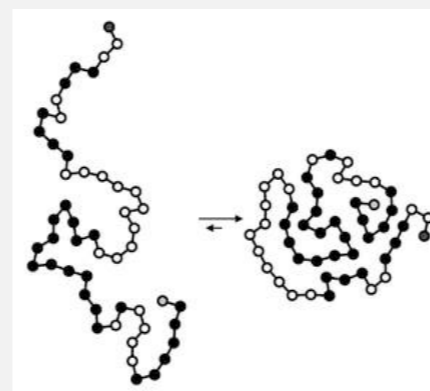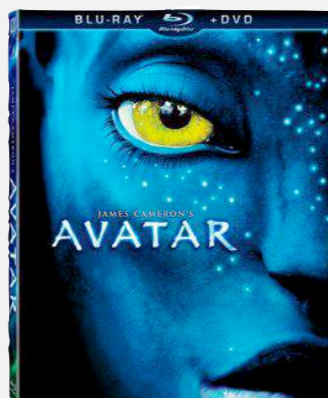Computer graphics. Movies, video games, virtual reality, ...

Security. Cell phones, e-commerce, voting machines, ...

Multimedia. MP3, JPG, DivX, HDTV, face recognition, ...

Social networks. Recommendations, news feeds, advertisements, ...

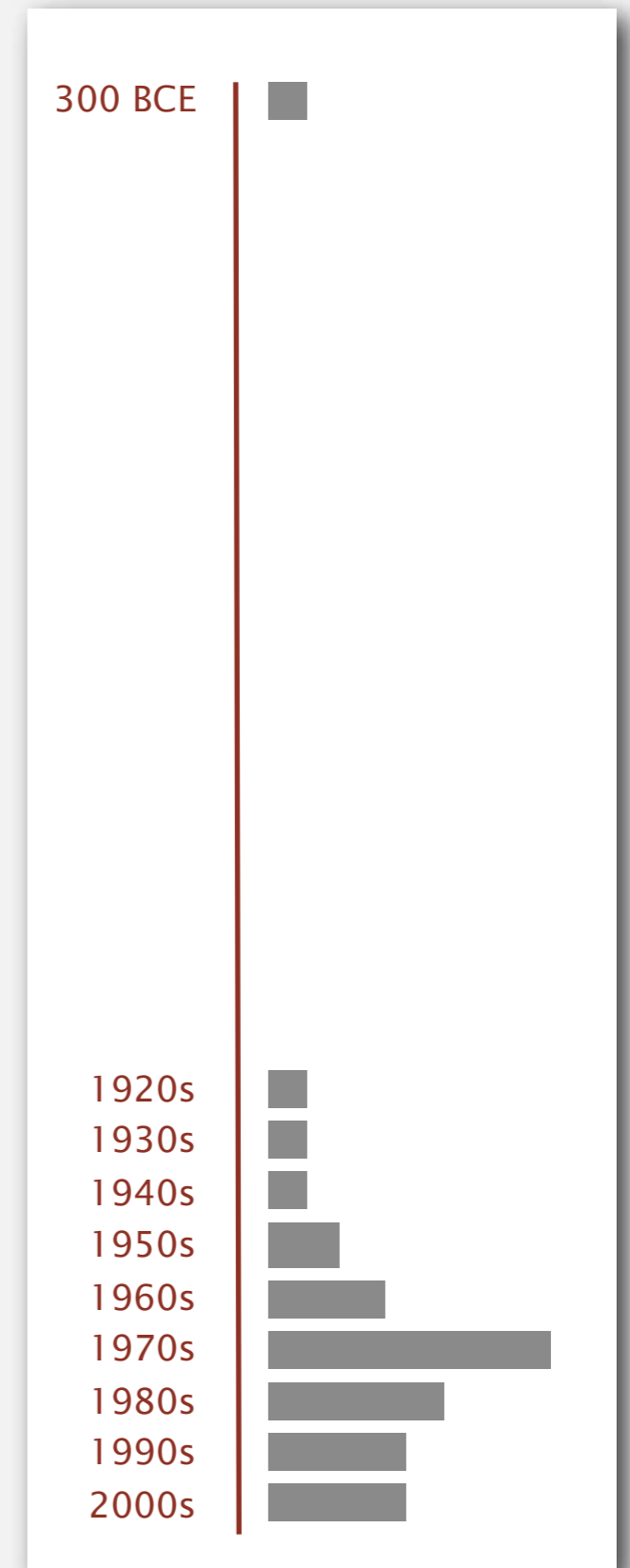Physics. N-body simulation, particle collision simulation, ...

⋮
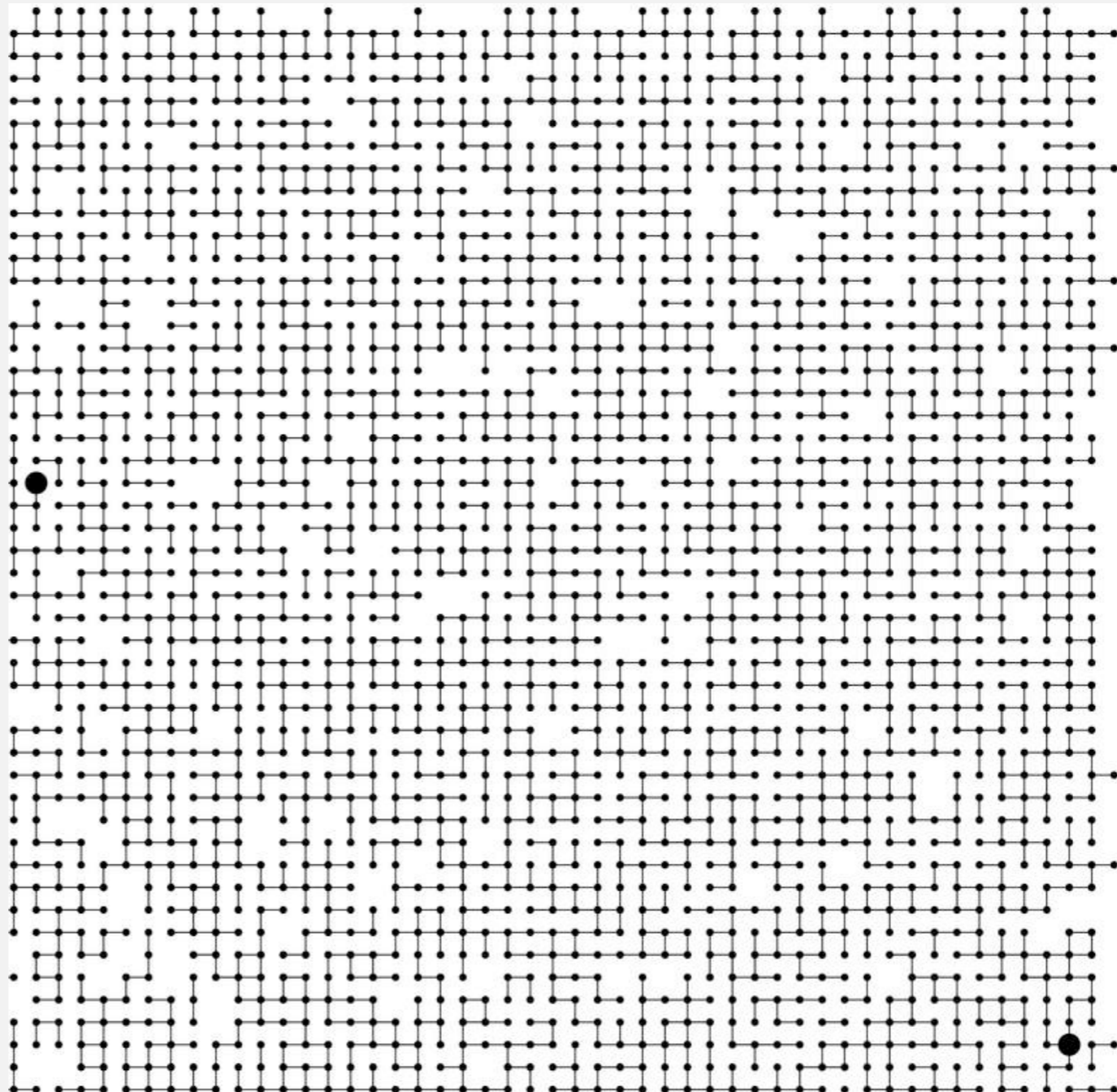
Why study algorithms?

Old roots, new opportunities.

- Study of algorithms dates at least to Euclid.
- Formalized by Church and Turing in 1930s.
- Some important algorithms were discovered by undergraduates in a course like this!

Why study algorithms?

To solve problems that could not otherwise be addressed.

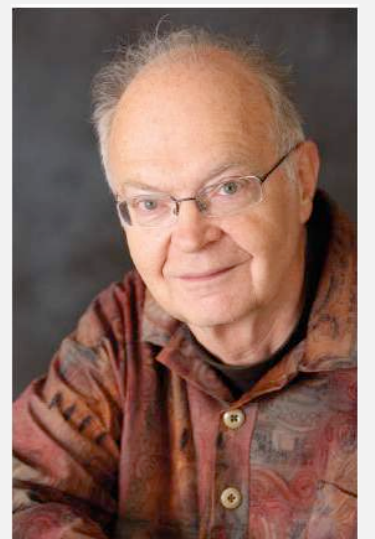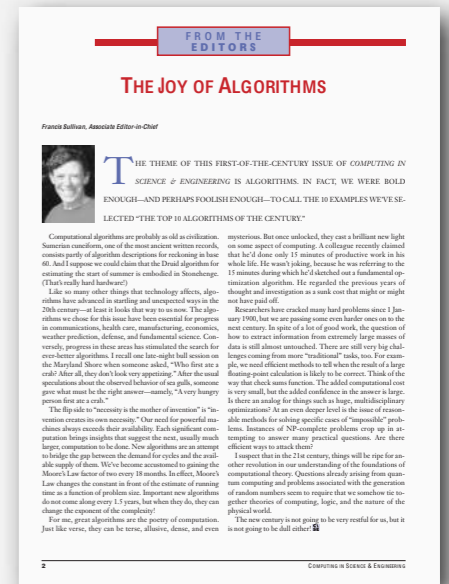Ex. Network connectivity. [stay tuned]

Why study algorithms?

For intellectual stimulation.

" For me, great algorithms are the poetry of computation. Just like verse, they can be terse, allusive, dense, and even mysterious. But once unlocked, they cast a brilliant new light on some aspect of computing. " — *Francis Sullivan*

" An algorithm must be seen to be believed. " — *Donald Knuth*
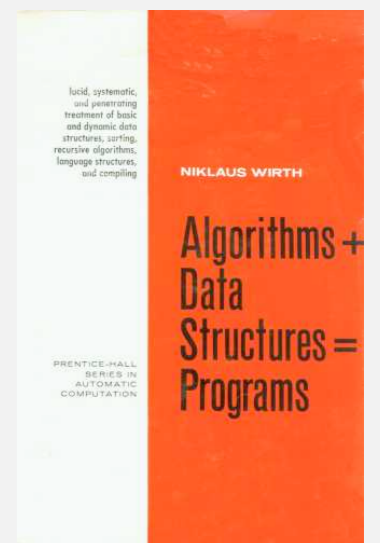
Why study algorithms?

To become a proficient programmer.

" I will, in fact, claim that the difference between a bad programmer and a good one is whether he considers his code or his data structures more important. Bad programmers worry about the code. Good programmers worry about data structures and their relationships. "

— Linus Torvalds (creator of Linux)

" Algorithms + Data Structures = Programs. "   — Niklaus Wirth

Why study algorithms?

They may unlock the secrets of life and of the universe.

Computational models are replacing math models in scientific inquiry.

$$E = mc^2$$

$$F = ma$$

$$F = \frac{Gm_1m_2}{r^2}$$

$$\left[-\frac{\hbar^2}{2m}\nabla^2 + V(r)\right]\Psi(r) = E\,\Psi(r)$$

20th century science
(formula based)

```
for (double t = 0.0; true; t = t + dt)
    for (int i = 0; i < N; i++)
    {
        bodies[i].resetForce();
        for (int j = 0; j < N; j++)
            if (i != j)
                bodies[i].addForce(bodies[j]);
    }
```

21st century science
(algorithm based)

" *Algorithms: a common language for nature, human, and computer.* " — *Avi Wigderson*
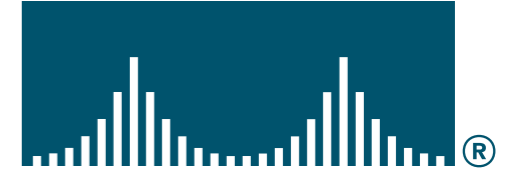
Why study algorithms?

For fun and profit.

Why study algorithms?

- Their impact is broad and far-reaching.
- Old roots, new opportunities.
- To solve problems that could not otherwise be addressed.
- For intellectual stimulation.
- To become a proficient programmer.
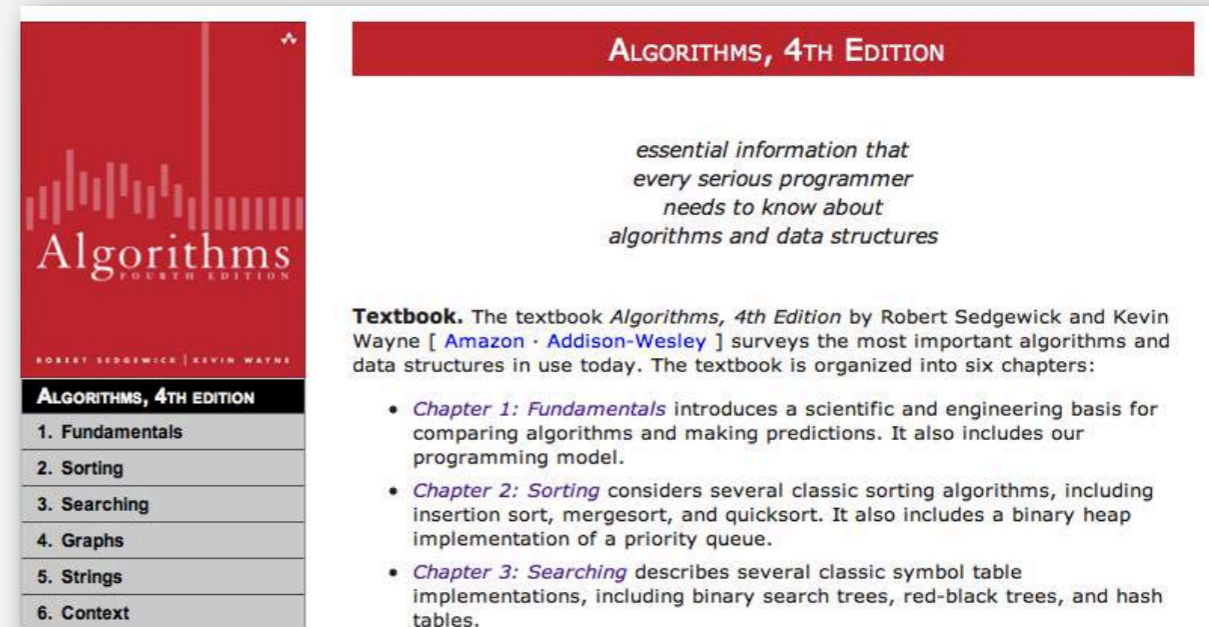- They may unlock the secrets of life and of the universe.
- For fun and profit.

Why study anything else?

Resources

Booksite.

- Lecture slides.
- Download code.
- Summary of content.

http://algs4.cs.princeton.edu

Textbook (optional).

- *Algorithms, 4th edition* by Sedgewick and Wayne.
- More extensive coverage of topics.
- More topics.

ISBN 0-321-57351-X

Prerequisites

Prerequisites.

- Programming: loops, arrays, functions, objects, recursion.
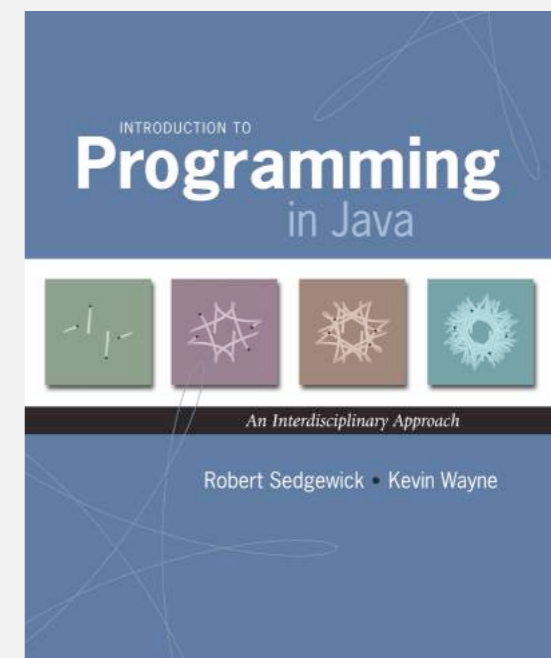- Java: we use as expository language.
- Mathematics: high-school algebra.

Review of prerequisite material.

- Quick: Sections 1.1 and 1.2 of *Algorithms, 4th edition.*
- In-depth: *An Introduction to programming in Java: an interdisciplinary approach* by Sedgewick and Wayne.

Programming environment.

- Use your own, e.g., Eclipse.
- Download ours (see instructions on web).

Quick exercise. Write a Java program.

ISBN 0–321–49805–4
http://introcs.cs.princeton.edu