

COS 226, FALL 2019

ALGORITHMS and DATA STRUCTURES

KEVIN WAYNE · MAIA GINSBURG · IBRAHIM ALBLUWI



PRINCETON
UNIVERSITY



Algorithms

ROBERT SEDGEWICK | KEVIN WAYNE

<https://algs4.cs.princeton.edu>

INTRO TO COS 226

- ▶ *motivation*
- ▶ *course structure*
- ▶ *assessments*
- ▶ *resources*

COS 226 course overview

What is COS 226?

- Intermediate-level survey course.
- Programming and problem solving, with applications.
- **Algorithm:** sequence of instructions for solving a problem.
- **Data structure:** method to organize data in a computer.

topic	data structures and algorithms
data types	stack, queue, union–find, priority queue
sorting	quicksort, mergesort, heapsort, radix sorts
searching	BST, red–black BST, hash table
graphs	BFS, DFS, Prim, Kruskal, Dijkstra
strings	KMP, regular expressions, tries, data compression
advanced	k-d tree, suffix array, maxflow

Why study algorithms and data structures?

Their impact is broad and far-reaching.

The image is a collage of news snippets from different publications, all centered around the theme of algorithms and their impact on society. At the top left, a Wall Street Journal article discusses how companies like Wal-Mart and Credit Suisse use algorithms to predict employee turnover. Another snippet from The Verge talks about prisons using algorithms for parole decisions. A third snippet from eHarmony questions if mathematical algorithms can find true love. Other articles mention Google's algorithm changes, the rise of Bitcoin, and how UPS uses algorithms to optimize delivery routes. The snippets are presented as if they are torn pieces of paper, creating a layered, overlapping effect.

MANAGEMENT

Algorithm That Tells the Boss Who Might Quit

Wal-Mart, Credit Suisse Crunch Data to See Which Workers Are Likely to Leave or Stay

Prisons turn to computer algorithms for deciding who to parole

By Jacob Kastrenakes on October 14, 2013 10:06 am Email @jake_k

DON'T MISS STORIES FOLLOW THE VERGE

POPULAR ON WSJ 18 COMMENTS Will world Hillary Clinton, Not

CIO JOURNAL.

Algorithms Will Drive Future Health Gains, Dean of Stanford Predicts

Med ALGORITHMS TAKE CONTROL OF WALL STREET

Can maths find you love? eHarmony's love algorithm

maths find you love? The dating harmony, who claim to have responsible for a on marriages, U se. ov id

This Algorithm Knows You Better Than Your Facebook Friends Do

The Algorithm Economy Heads To Amazon

Posted Nov 30, 2014 by Danny Crichton (@DannyCrichton)

1,796 SHARES

THE SATURDAY ESSAY

Bitcoin and the Digital-Currency Revolution

For all bitcoin's growing pains, it represents money and global finance.

THE WALL STREET JOURNAL. TECH

At UPS, the Algorithm Is the Driver

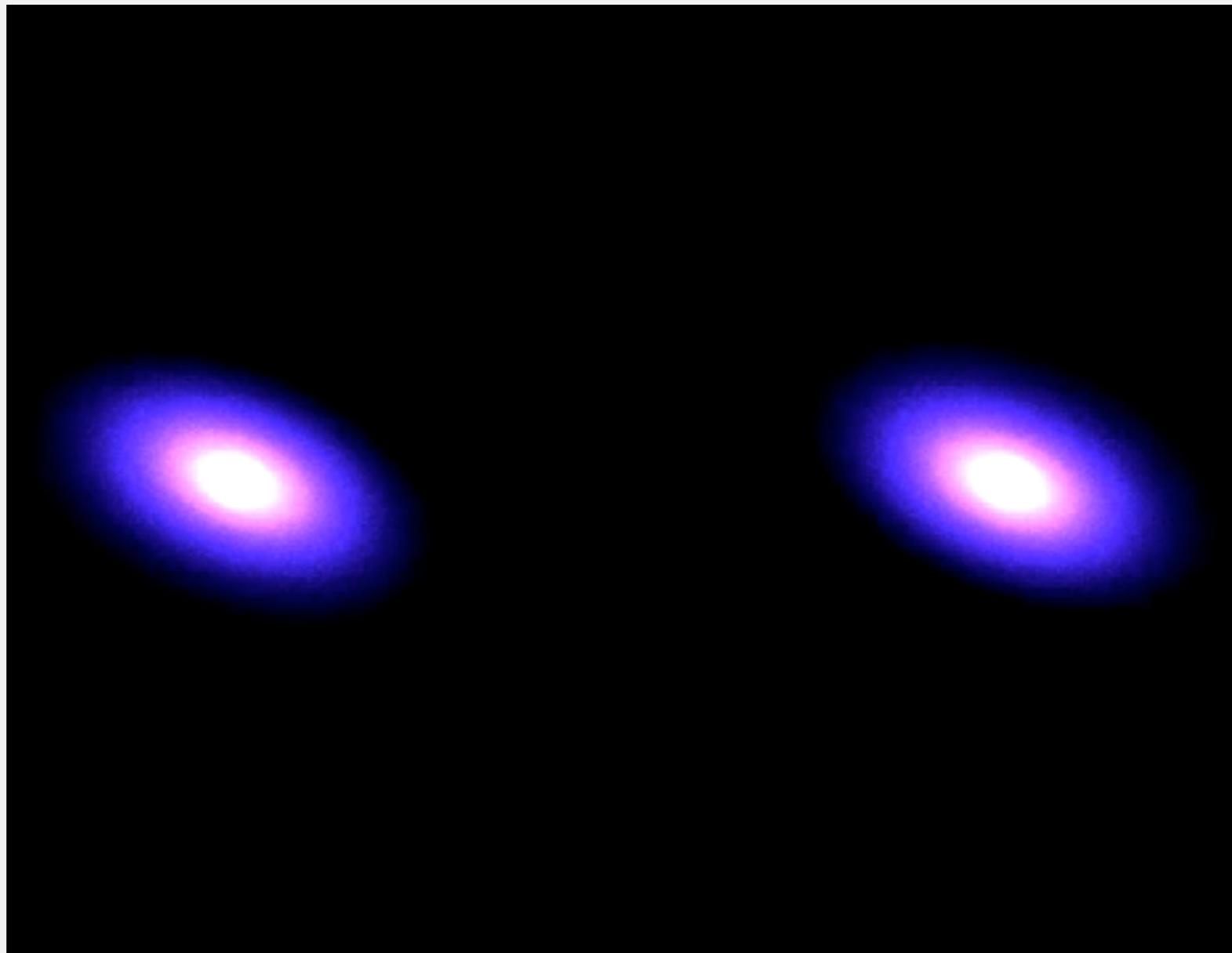
Turn right, turn left, turn right: inside Orion, the 10-year effort to squeeze every penny from

By STEVEN ROSENBUCH and LAURA STEVENS

Feb. 16, 2015 8:28 p.m. ET

Why study algorithms and data structures?

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

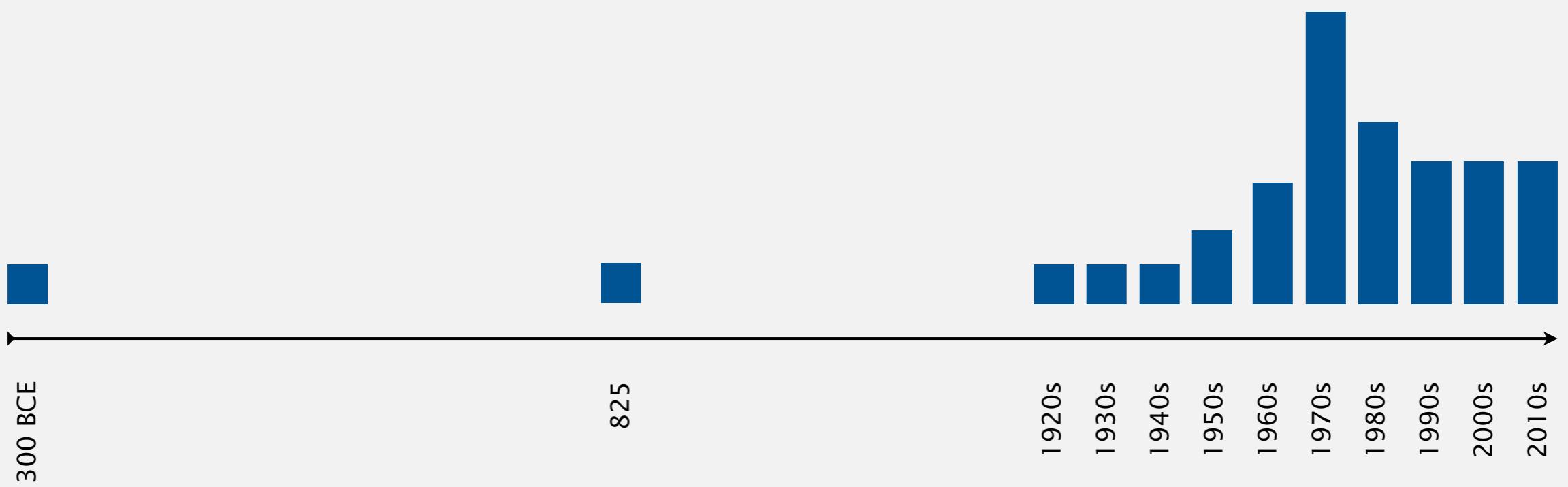


http://www.youtube.com/watch?v=ua7YIN4eL_w

Why study algorithms and data structures?

Old roots, new opportunities.

- Study of algorithms dates at least to Euclid.
- Named after Muḥammad ibn Mūsā al-Khwārizmī.
- Formalized by Church and Turing in 1930s.
- Some important algorithms were discovered by undergrads in a course like this!



Why study algorithms and data structures?

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 (architect of Linux and git)



Why study algorithms and data structures?

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

FROM THE EDITORS

THE JOY OF ALGORITHMS

Francis Sullivan, Associate Editor-in-Chief



THE THEME OF THIS FIRST-OF-THE-CENTURY ISSUE OF COMPUTING IN SCIENCE & ENGINEERING IS ALGORITHMS. IN FACT, WE WERE BOLD ENOUGH—AND PERHAPS FOOLISH ENOUGH—to call the 10 examples we've selected “THE TOP 10 ALGORITHMS OF THE CENTURY.”

Computational algorithms are probably as old as civilization. Sumerian cuneiform, one of the most ancient written records, contains partly of algorithmic descriptions of reckoning in base 60. And we could cite the work of Donald Knuth, who wrote the 15 minutes during which he sketched out a fundamental operation of the simplex algorithm. He regards the 100 years of thought and investigation that went into it as might as might not have paid off.

Like so many other things that technology affects, algorithms have moved in startling and unexpected ways in the last decade. The 10 algorithms we chose for this issue have been essential for progress in communications, health care, manufacturing, economics, and more. The search for better ways to store and process data is still almost untouched. There are still very big challenges coming from more “traditional” tasks, too. For example, I recall one late-night bad session on the simplex algorithm. “It’s taking forever,” I said. “Is there a bug? After all, they don’t look very appealing.” After the usual speculations about observed behavior of set paths, someone gave up and asked, “What would happen if we swapped the person first and a crabs?”

The flip side to “necessity is the mother of invention” is “the inventiveness of necessity.” Our need for powerful machines always exceeds their availability. Each significant computational insight that begins the trend, usually much larger computers, leads to a desire for algorithms that will help to bridge the gap between the demand for cycles and the available ones. One of the most remarkable accounts of Moore’s Law factor of two every 18 months. In effect, Moore’s Law changes the constant in front of the estimate of running time. It does not change the exponent. If you run a program that does not come along every 1.5 years, but when they do, they can change the exponent of the complexity!

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. A colleague recently claimed that he’d done only 15 minutes of productive work in his office. After a brief lecture, however, we were up to the 15 minutes during which he sketched out a fundamental operation of the simplex algorithm. He regards the 100 years of thought and investigation that went into it as might as might not have paid off.

Computational algorithms have cracked many hard problems since January 1950, but we are posing some even harder ones on the next century. In spite of a lot of good work, the question of how to store and process data is still almost untouched. There are still very big challenges coming from more “traditional” tasks, too. For example, I recall one late-night bad session on the simplex algorithm. “It’s taking forever,” I said. “Is there a bug? After all, they don’t look very appealing.” After the usual speculations about observed behavior of set paths, someone gave up and asked, “What would happen if we swapped the person first and a crabs?”

Is there an analog for chips such as logic, multidisciplinary optimization? At an even deeper level is the issue of reasoning that goes beyond the quantum leap that occurred in 1950. Instances of NP-complete problems crop up in attempting to answer many practical questions. Are there

computational algorithms that will be ripe for an even revolution in our understanding of the nature of computation and problems? Quantum leaps arising from quantum computing and problems associated with the generation of random numbers are just a few examples. We may get together theories of computing, logic, and the nature of the physical world.

The new century is not going to be very useful for us, but it is not going to be dull either. ■



Why study algorithms and data structures?

For fun and profit.

Google



facebook

Cisco Systems



Morgan Stanley

IBM

Nintendo®

JANE STREET

NETFLIX

Adobe

RSA
SECURITY™

D E Shaw & Co

ORACLE®

P
PANDORA®

Akamai

YAHOO!®

amazon

Microsoft®

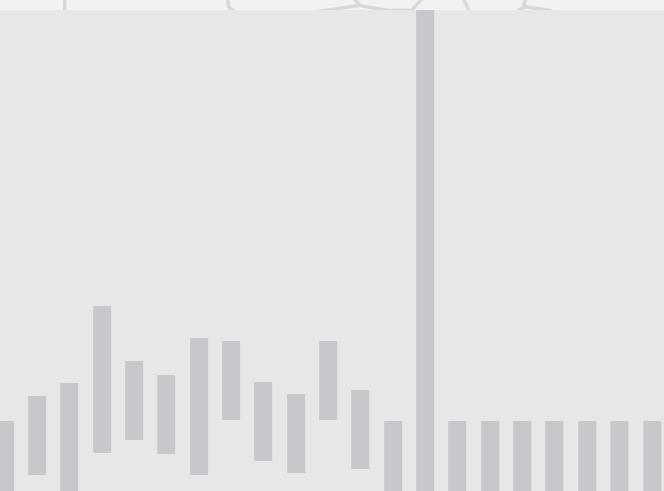
P I X A R
ANIMATION STUDIOS

Why study algorithms and data structures?

- Their impact is broad and far-reaching.
- They may unlock the secrets of life and of the universe.
- Old roots, new opportunities.
- To become a proficient programmer.
- For intellectual stimulation.
- For fun and profit.

Why study anything else?





Algorithms

ROBERT SEDGEWICK | KEVIN WAYNE

<https://algs4.cs.princeton.edu>

INTRO TO COS 226

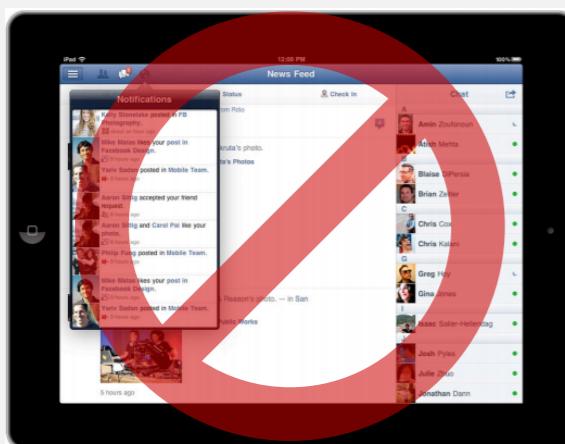
- ▶ *motivation*
- ▶ **course structure**
- ▶ *assessments*
- ▶ *resources*

Lectures

Live lectures. Introduce new material.

What	When	Where	Who	Office Hours
L01	TTh 11-12:20	Friend 101	Kevin Wayne	M 1:30-3:30pm

Electronic devices. Permitted *only* to support lecture
(e.g., viewing slides and taking notes).





Student response system (required).

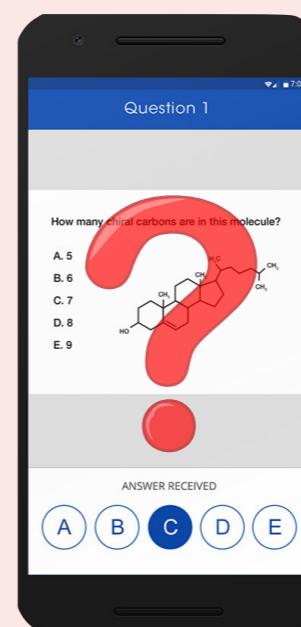
- Any hardware version of iClicker.
(use iClicker Reef at your own risk, WiFi issues?)
- Register your iClicker in Blackboard. 
- Available at Labyrinth Books (\$30).

use only one device
per lecture!



Which model of iClicker are you using?

- A. iClicker.
- B. iClicker+.
- C. iClicker 2.
- D. iClicker Reef.



Precepts

Discussion, problem-solving, assignment prep, ...



Maia Ginsburg

Faculty

Lead Preceptor



Ibrahim Albluwi

Faculty

Lead Preceptor



Bob Tarjan

Faculty

Preceptor



Lisa Jian

Graduate Student

Preceptor



Chris Sciavolino

Graduate Student

Preceptor



Devon Loehr

Graduate Student

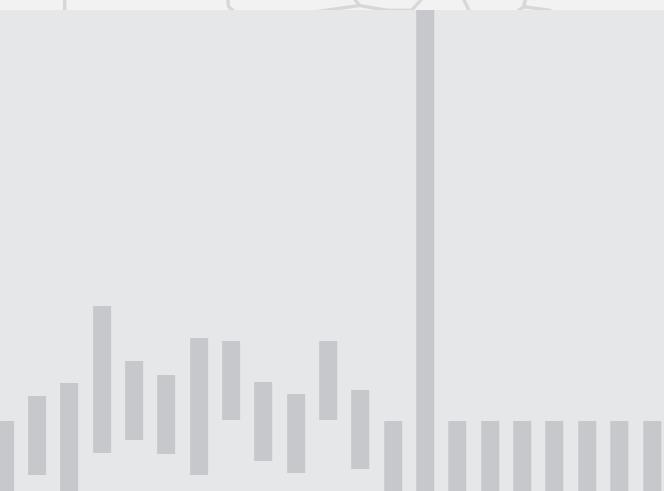
Preceptor

Precepts

Discussion, problem-solving, assignment prep, ...

What	When	Where	Who	
P01	Th 1:30–2:50pm	Friend 016	Maia Ginsburg	
P02	Th 3–4:20pm	Friend 016	Chris Sciavolino	
P04	F 11–12:20pm	Friend 009	Ibrahim Albluwi	
P05	F 11–12:20pm	Friend 111	Lisa Jian	
P07	F 1:30–2:50pm	Friend 009	Devon Loehr	
P08	F 3–4:20pm	Friend 009	Ibrahim Albluwi	
P09	Th 3–4:20pm	Sherred 001	Bob Tarjan	
NEW	P10	Th 3–4:20pm	TBA	Maia Ginsburg





Algorithms

ROBERT SEDGEWICK | KEVIN WAYNE

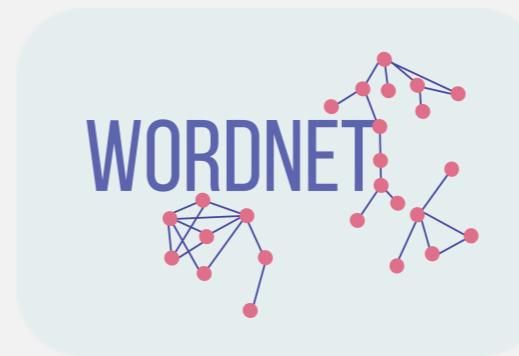
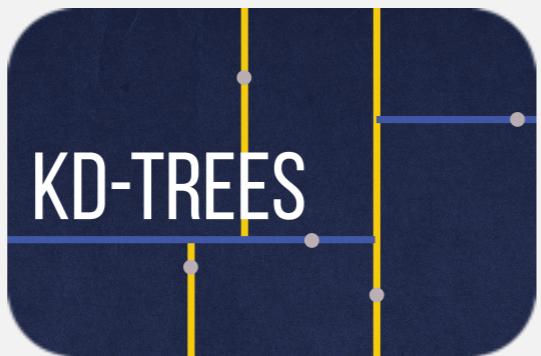
<https://algs4.cs.princeton.edu>

INTRO TO COS 226

- ▶ *motivation*
- ▶ *course structure*
- ▶ ***assessments***
- ▶ *resources*
- ▶ *union-find*

Programming assignments

Implement an efficient algorithm or data structure.



BURROWS WHEELER

0	A	B	R	A	C	A	D	A	B	A	I
1	B	R	A	C	A	D	A	B	A	I	A
2	R	A	C	A	D	A	B	A	I	A	B
3	A	C	A	D	A	B	A	I	A	B	R
4	C	A	D	A	B	A	I	A	B	R	A
5	A	D	A	B	A	I	A	B	R	A	C
6	D	A	B	A	I	A	B	R	A	C	A
7	A	B	A	I	A	B	R	A	C	A	D
8	B	A	I	A	B	R	A	C	A	D	A
9	A	I	A	B	R	A	C	A	D	A	B
10	I	A	B	R	A	C	A	D	A	B	A
11	A	B	R	A	C	A	D	A	B	A	I

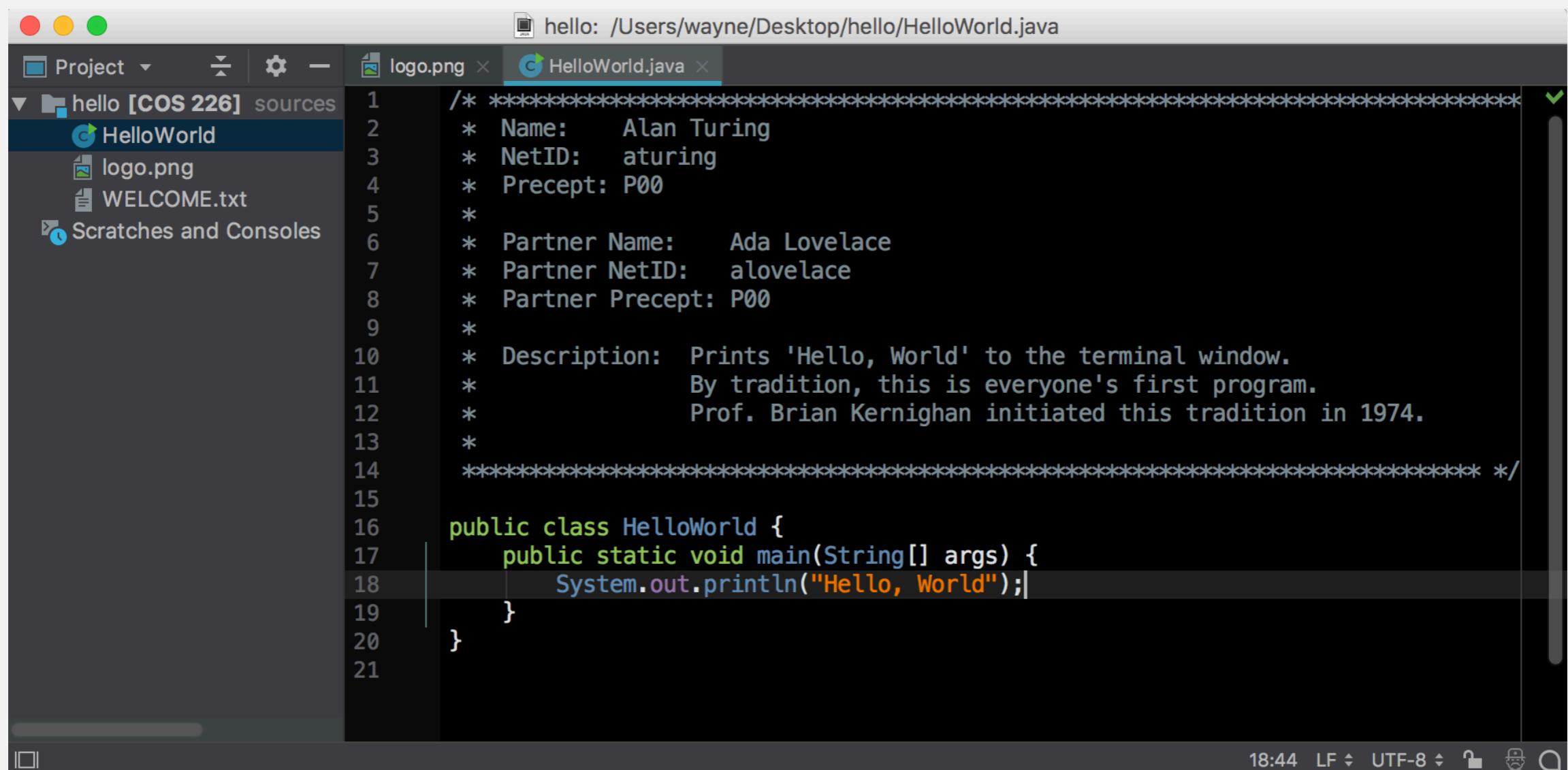
Solve an interesting application using a “textbook” algorithm.



Programming assignments

Recommended IDE. Custom IntelliJ environment (used in COS 126).

- Continuous code inspection; integrated Checkstyle and Spotbugs.
- Autoformat, autoimport, and autocomplete.
- Embedded bash terminal.
- ...



The screenshot shows the IntelliJ IDEA IDE interface. The title bar says "hello: /Users/wayne/Desktop/hello/HelloWorld.java". The left sidebar shows a project structure with a folder "hello [COS 226] sources" containing "HelloWorld", "logo.png", and "WELCOME.txt". Below this is a "Scratches and Consoles" section. The main editor window displays the Java code for "HelloWorld.java". The code includes a multi-line comment at the top with student information and a tradition note. The main method is defined as follows:

```
1  /* ****
2  * Name: Alan Turing
3  * NetID: aturing
4  * Precept: P00
5  *
6  * Partner Name: Ada Lovelace
7  * Partner NetID: alovelace
8  * Partner Precept: P00
9  *
10 * Description: Prints 'Hello, World' to the terminal window.
11 * By tradition, this is everyone's first program.
12 * Prof. Brian Kernighan initiated this tradition in 1974.
13 *
14 **** */
15
16 public class HelloWorld {
17     public static void main(String[] args) {
18         System.out.println("Hello, World");
19     }
20 }
21
```

The status bar at the bottom shows the time as 18:44 and encoding as UTF-8.

Quizzes



- 2–3 short questions per lecture.
- 3 attempts per question.
- Use pencil and paper.



Quizzera

wayne

Logout

Courses / Algorithms and Data Structures / Union Find

Quick Find

Attempts Remaining: 1

Quiz Ends in 2 days.

New Attempt

Attempts ▾

Seed: 50233 (Provider: QuickFindExercise)

Question

Give the id[] array that results from the following sequence of 6 union operations on a set of 10 items using the quick-find algorithm.

5-7 3-2 4-3 1-6 0-7 4-9

Recall: our quick-find convention for the union operation p-q is to change id[p] (and perhaps some other entries) but not id[q].

Answer

Your answer should be a sequence of 10 integers (between 0 and 9), separated by whitespace.

Submit

About

Midterm and final

Written exams.

- Questions drawn from quizzes and lectures.
- Emphasizes **non-programming** material.

COS 226	Algorithms and Data Structures	Fall 2017
Midterm		

This exam has 10 questions (including question 0) worth a total of 55 points. You have 80 minutes. This exam is preprocessed by a computer, so please **write darkly** and **write your answers inside the designated spaces**.

Policies. The exam is closed book, except that you are allowed to use a one page cheatsheet (8.5-by-11 paper, one side, in your own handwriting). No electronic devices are permitted.

Grading

Programming assignments. 45%

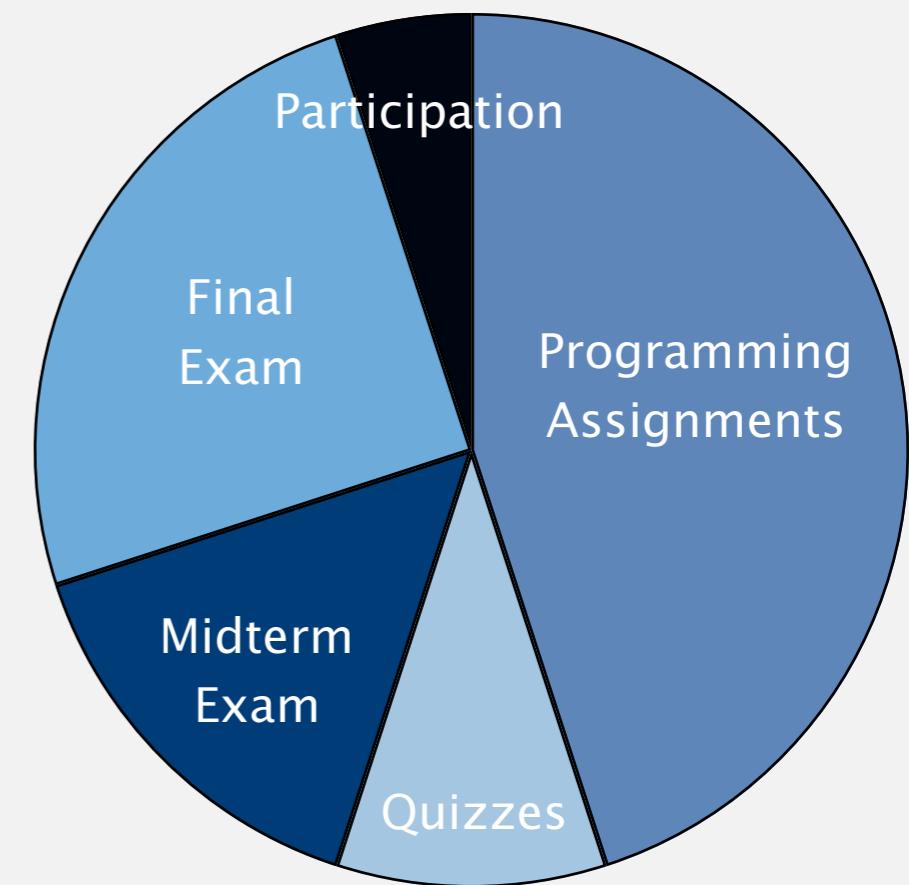
- Due at 11pm on Mondays via TigerFile.
- Collaboration/lateness policies: see web.

Quizzes. 10%

- Due at 11pm on Fridays via Quizzera.
- Collaboration/lateness policies: see web.

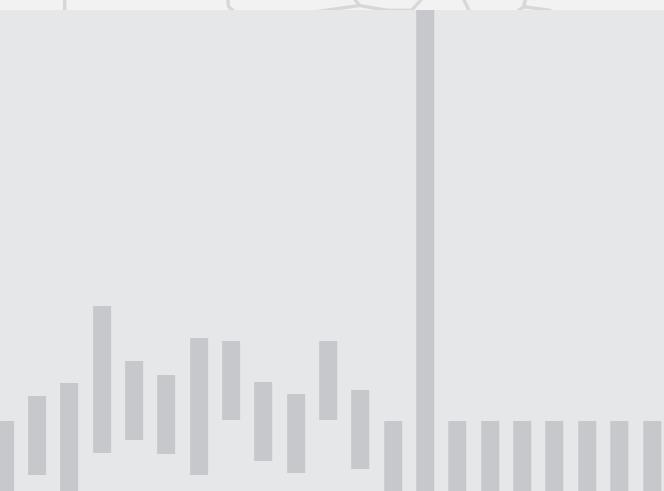
Exams. 15% + 25%

- Midterm (in class on Tuesday, October 22).
- Final (to be scheduled by Registrar).



Active participation. 5%

- Participate in precept/lecture.
(perfect attendance not required to earn 100% of participation points)
- Answer questions on Piazza.



Algorithms

ROBERT SEDGEWICK | KEVIN WAYNE

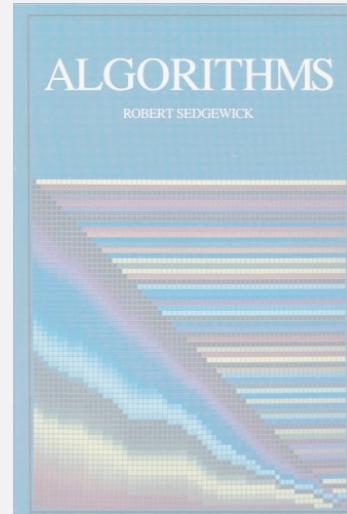
<https://algs4.cs.princeton.edu>

INTRO TO COS 226

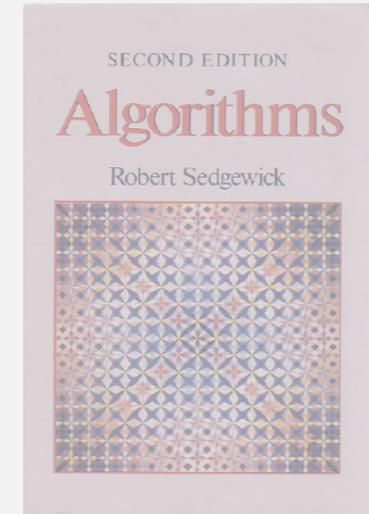
- ▶ *motivation*
- ▶ *course structure*
- ▶ *assessments*
- ▶ **resources**

Resources (textbook)

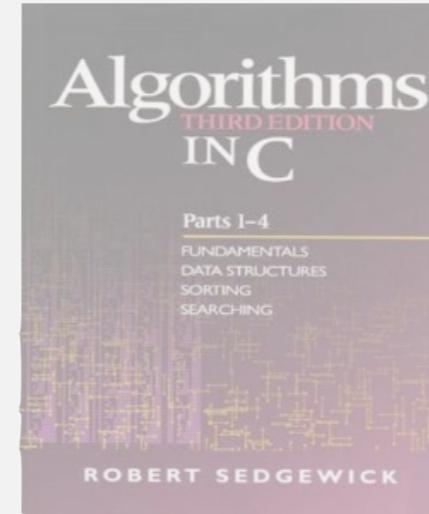
Readings (required). Algorithms 4th edition by R. Sedgewick and K. Wayne, Addison-Wesley Professional, 2011, ISBN 0-321-57351-X.



1st edition (1982)



2nd edition (1988)



3rd edition (1997)



4th edition (2011)

Available from various vendors in hardcover and ebook formats.

- Amazon: \$75 hardcover, \$58 Kindle, ...
- Labyrinth: \$63 hardcover, \$40 rent.
- Engineering library: on reserve.
- Safari Tech Books Online.



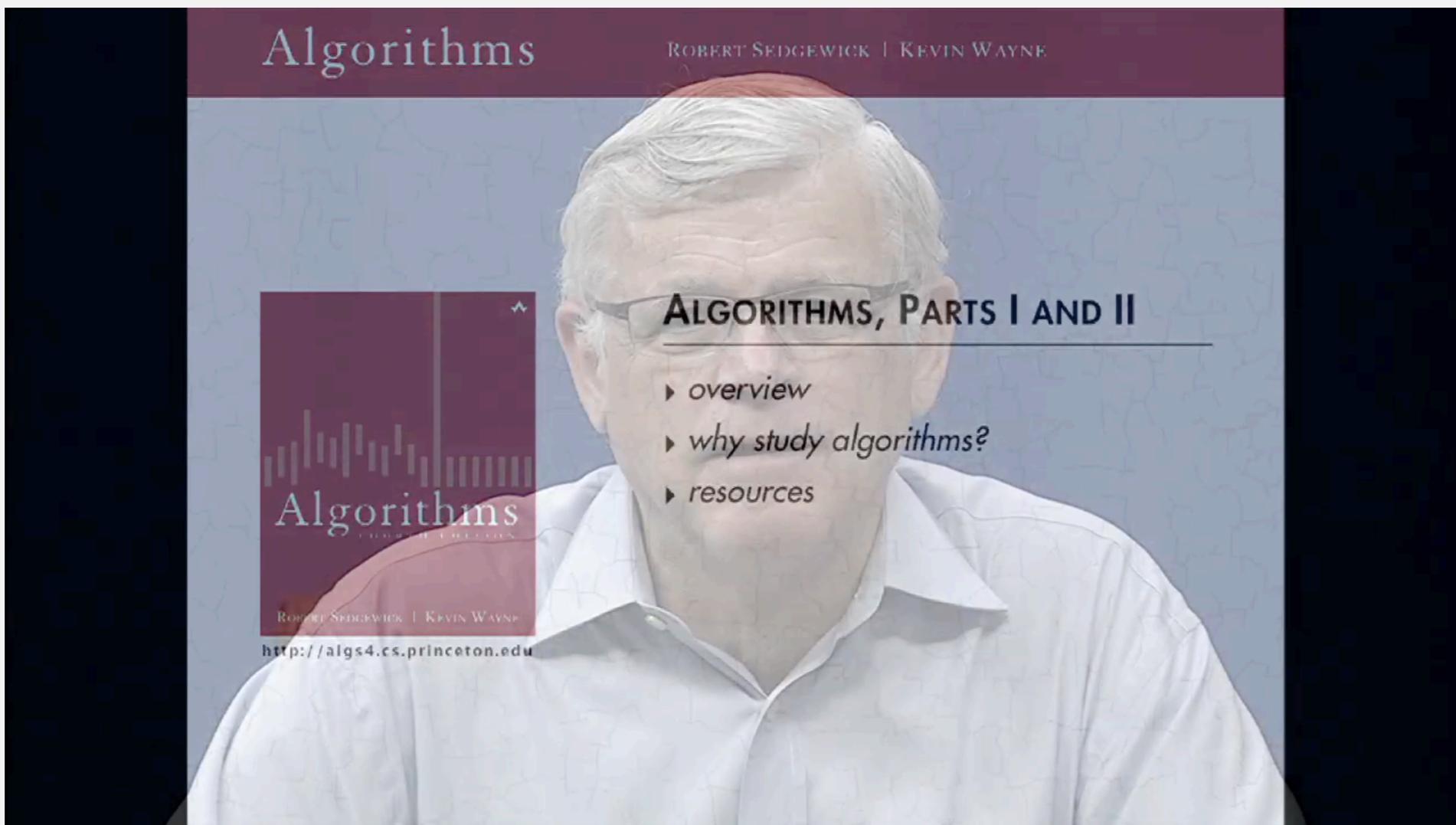
LABYRINTH
BOOKS



Resources (videos)

Lecture videos (optional).

- Missed lecture.
- Review for exams.



Resources (videos)

Lecture videos (optional).

- Missed lecture.
- Review for exams.

The screenshot shows the Acurate platform interface. At the top, there is a teal header bar with the word "Acurate" on the left, a search bar containing "percolation" in the center, and a user profile icon with the text "Hello, cas-princeton-university-wayne" on the right.

Below the header, the main area displays search results for "percolation". On the left, a large button labeled "View your progress" with a progress bar is visible. To its right, the search results are listed:

- Search Results: "percolation"**: This section contains two video thumbnails.
 - 1.E Applications**: Shows a thumbnail of a video titled "Monte Carlo simulation" with a QR code. It has a progress bar at 0% and a timestamp of 0:00.
 - 3.F* Applications**: Shows a thumbnail of a video titled "Arithmetic expression evaluation" with a thumbnail of a person speaking. It has a progress bar at 0% and a timestamp of 0:00.
- Include: everything**: A dropdown menu currently set to "everything".
- 2 Results**: A summary of the total number of results found.

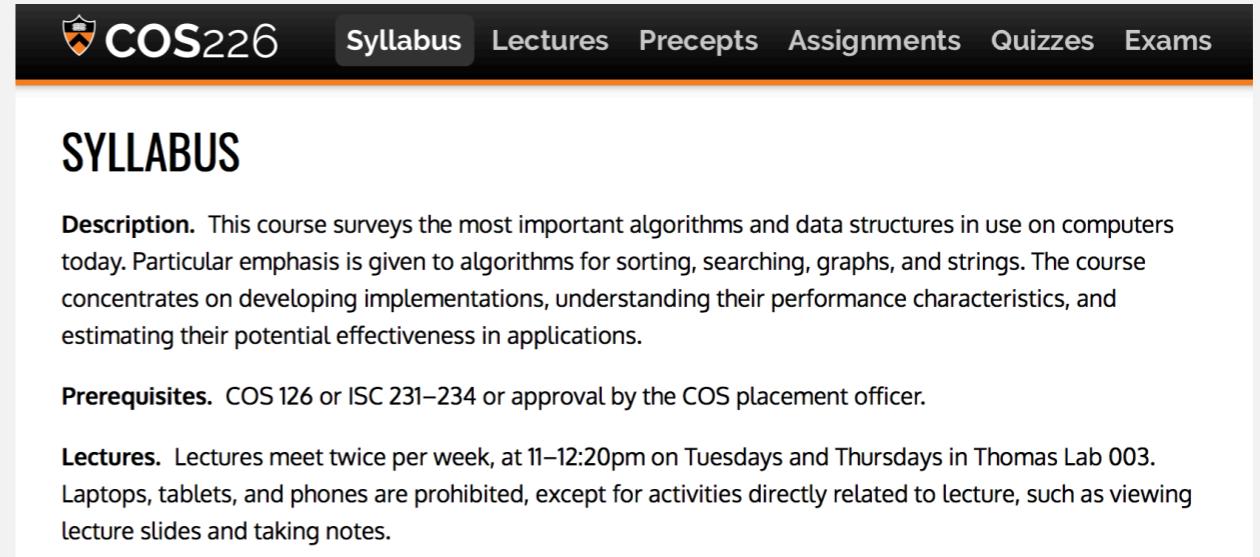
Below the search results, a section titled "1.E Applications" is expanded, showing the following transcript highlights:

- 0:56 So, the one we're going to talk about now is called **percolation**.
- 2:49 That's just a few examples of the **percolation** model.
- 6:17 So the **percolation** model on the left corresponds to the, connection model on the right, according to what we've been doing.
- 7:41 And that's where we get the result that, by running enough simulations for a big-enough n, that this, **percolation** threshold is about.

Resources (web)

Course content.

- Course info.
- Lecture slides.
- Programming assignments.
- Quizzes.
- Exam archive.

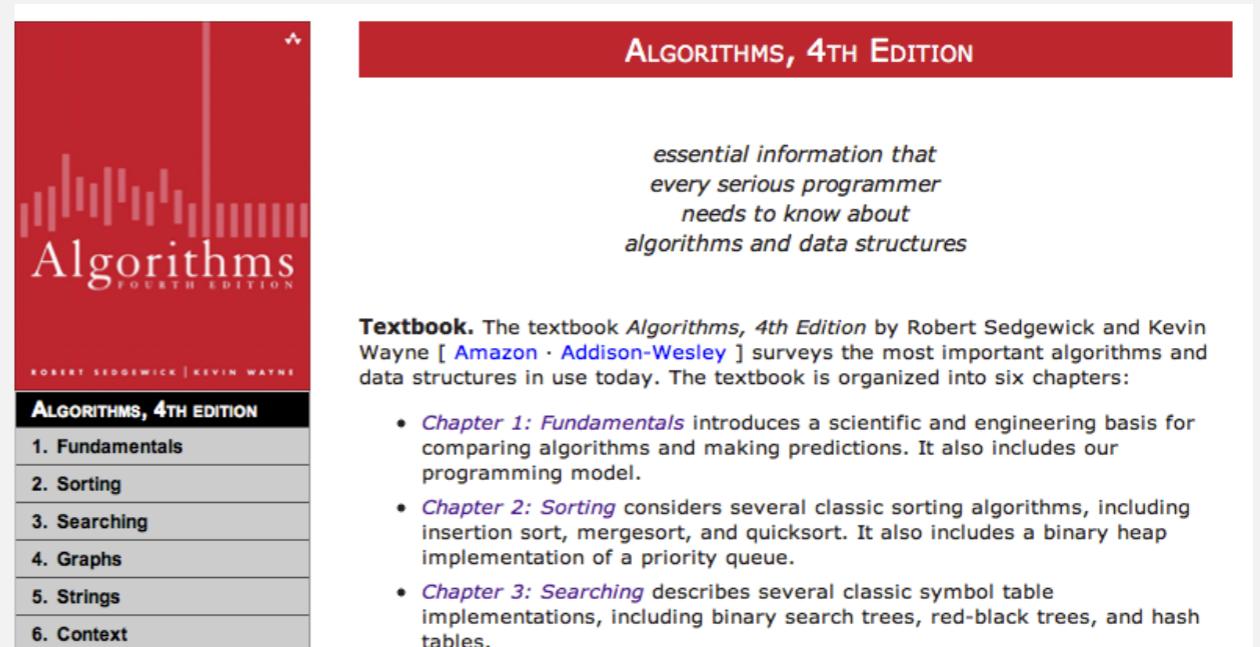


The screenshot shows the COS226 syllabus page. At the top, there is a navigation bar with links for Syllabus, Lectures, Precepts, Assignments, Quizzes, and Exams. The main content area is titled "SYLLABUS". It contains a "Description" section which states: "This course surveys the most important algorithms and data structures in use on computers today. Particular emphasis is given to algorithms for sorting, searching, graphs, and strings. The course concentrates on developing implementations, understanding their performance characteristics, and estimating their potential effectiveness in applications." Below it is a "Prerequisites" section stating: "COS 126 or ISC 231–234 or approval by the COS placement officer." There is also a "Lectures" section describing the meeting times and rules.

<https://www.cs.princeton.edu/~cos226>

Booksite.

- Brief summary of content.
- Download code from book.
- APIs and Javadoc.



The screenshot shows the book page for "Algorithms, 4th Edition". The page features the book's cover image on the left and a summary text on the right. The summary text reads: "essential information that every serious programmer needs to know about algorithms and data structures". Below this is a "Textbook" section which describes the book as surveying important algorithms and data structures. It also lists the six chapters: Fundamentals, Sorting, Searching, Graphs, Strings, and Context. A list of bullet points below the chapters provides a brief overview of each chapter's content.

Textbook. The textbook *Algorithms, 4th Edition* by Robert Sedgewick and Kevin Wayne [[Amazon](#) · [Addison-Wesley](#)] surveys the most important algorithms and data structures in use today. The textbook is organized into six chapters:

- *Chapter 1: Fundamentals* introduces a scientific and engineering basis for comparing algorithms and making predictions. It also includes our programming model.
- *Chapter 2: Sorting* considers several classic sorting algorithms, including insertion sort, mergesort, and quicksort. It also includes a binary heap implementation of a priority queue.
- *Chapter 3: Searching* describes several classic symbol table implementations, including binary search trees, red-black trees, and hash tables.

<https://algs4.cs.princeton.edu>

Resources (people)

Piazza discussion forum.

- Low latency, low bandwidth.
- See Piazza for guidelines.



<https://piazza.com/princeton/fall2019/cos226>

Office hours.

- High bandwidth, high latency.
- See web for schedule.



<https://www.cs.princeton.edu/~cos226>

Computing laboratory.

- Undergrad lab TAs.
- For help with debugging.
- See web for schedule.



<https://labta.cs.princeton.edu>

This week



you are here!

precept starts tomorrow (or today)
read Assignment 1 before precept

Sun	Mon	Tue	Wed	Thu	Fri	Sat
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28

protip: start early

yes, really!

A typical week



Sun	Mon	Tue	Wed	Thu	Fri	Sat
8	9	10	11	12	13	14
15	16	17 Lecture 2 (Analysis)	18	19 Lecture 3 (Stacks)	20 Precept 2 Quiz 2 and 3	21
22	23 Assignment 2 (Deques+RQs)	24	25	26	27	28

content based on
week's material

support lecture material;
assignment prep

content based on
corresponding lectures

Q+A

Not registered? Register ASAP; attend any precept this week.

Change precept? Use TigerHub.

All non-conflicting precepts closed? See Colleen Kenny-McGinley in CS 210.

Haven't taken COS 126? See COS placement officer.

Placed out of COS 126? Review Sections 1.1–1.2 of Algorithms 4/e.

