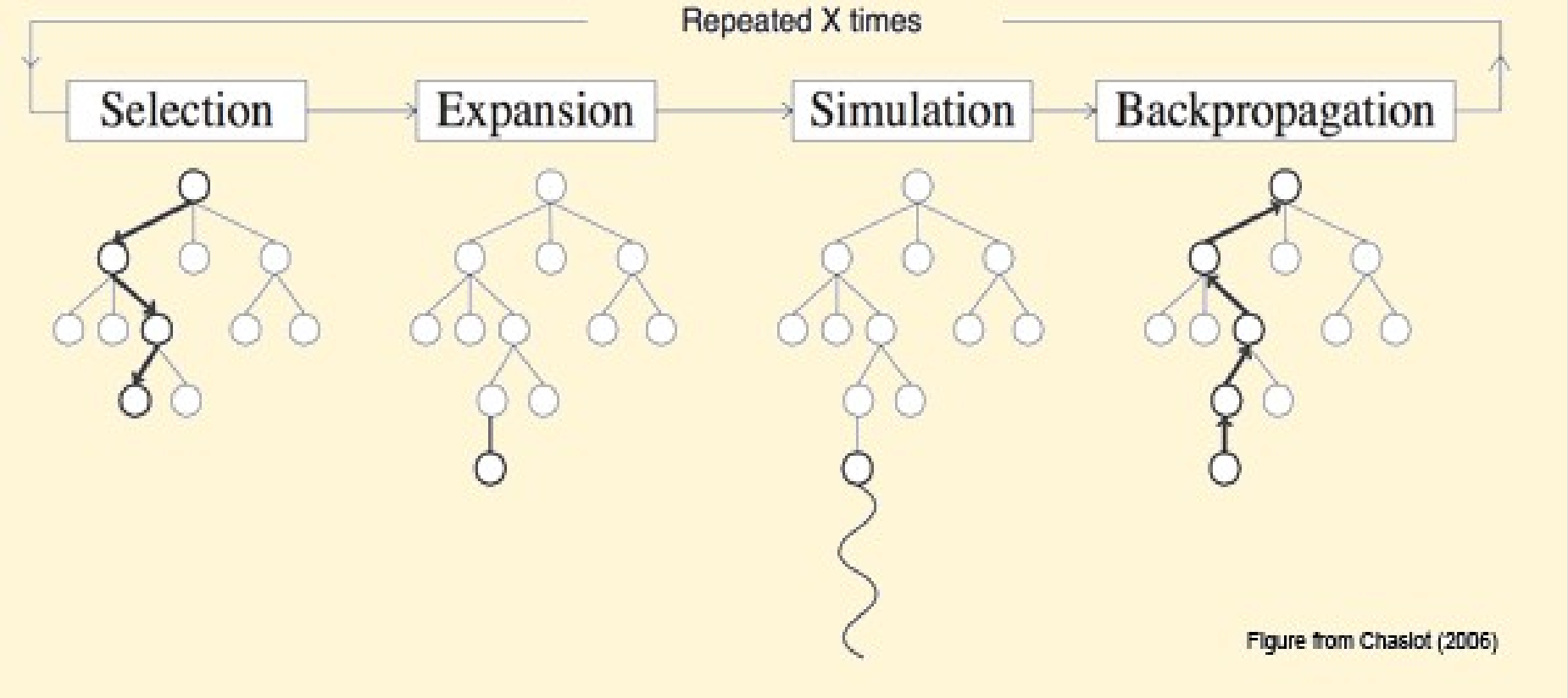
Welcome to CSci 4041

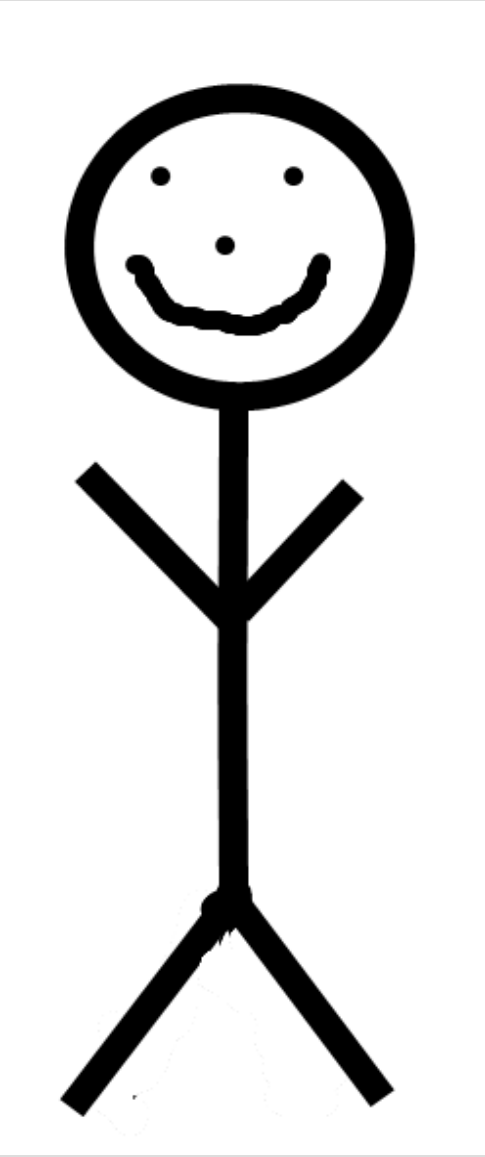
Algorithms and Data Structures



Instructor (me)

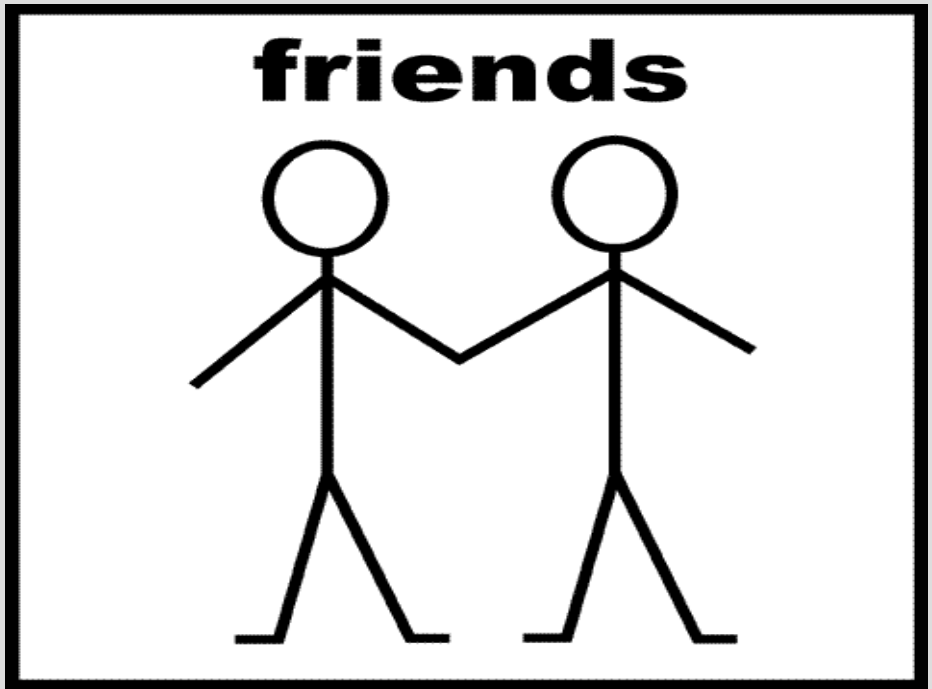
James Parker
Shepherd Labs 391

Primary contact:
jparker@cs.umn.edu



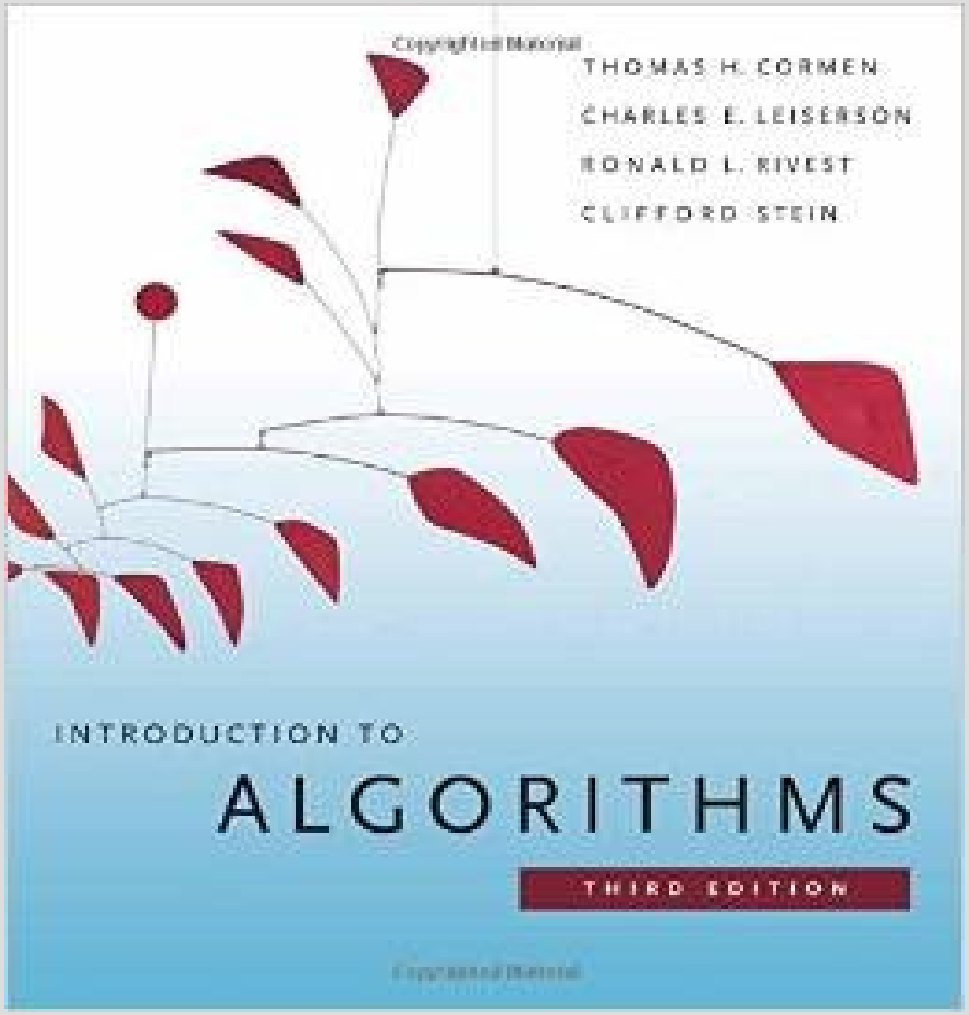
Teaching Assistant

Pariya Babaie, Jayant Gupta,
Song Liu, Anoop Shukla,
Nikolaos Stefas, Kshitij Tayal
Nitin Varyani



Textbook

Introduction to Algorithms, Cormen et al., 3rd edition



Discussion sections

No discussion on Friday
(don't come, no one will be there)

These will typically reinforce the topics of the week (or exam review)

The TAs may do exercises, so bring something to write on

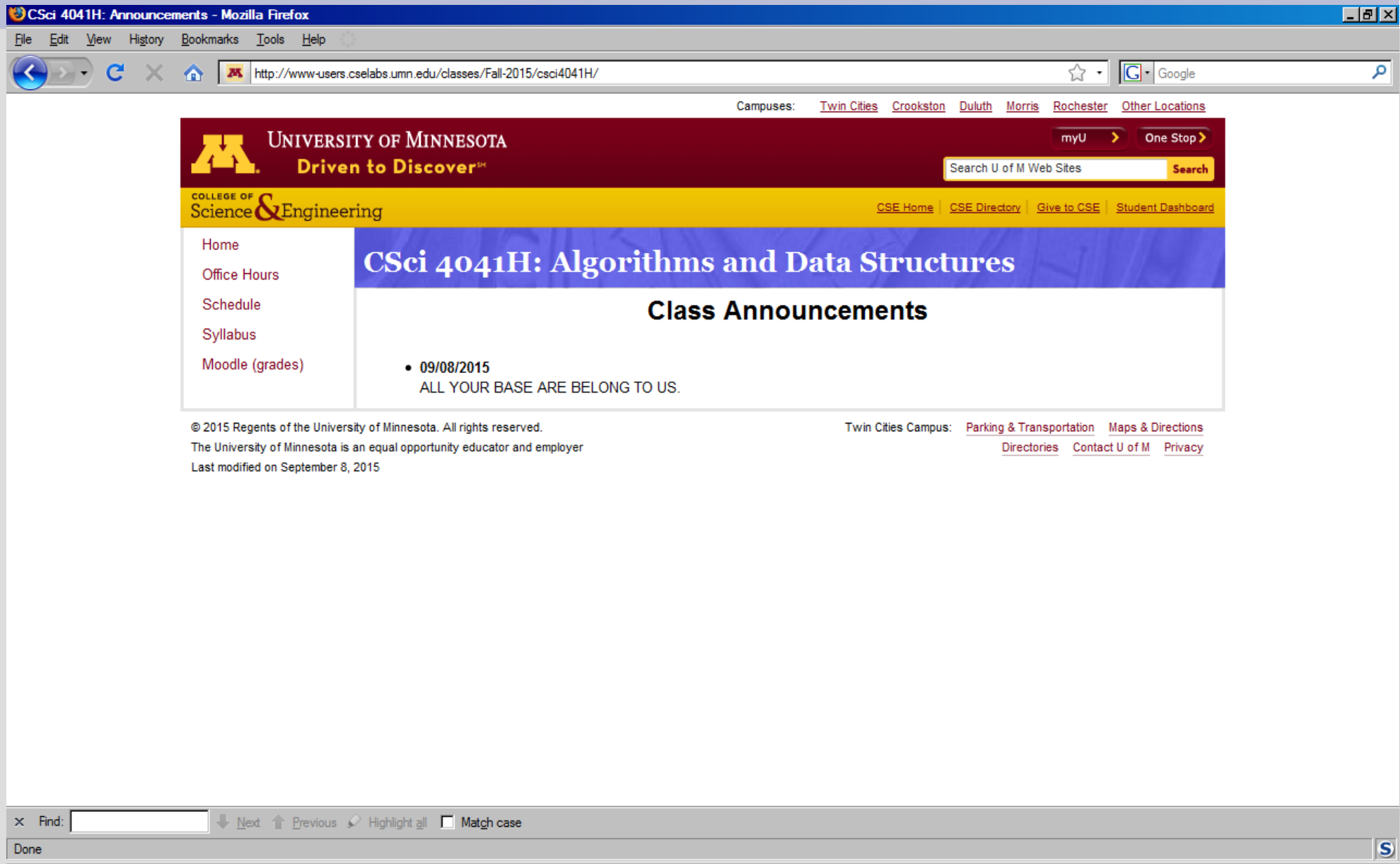
Class website

www.cs.umn.edu/academics/classes
Or google “umn.edu csci class”

Syllabus, schedule, other goodies

Moodle page will have grades and
Possibly homework submission

www.cs.umn.edu

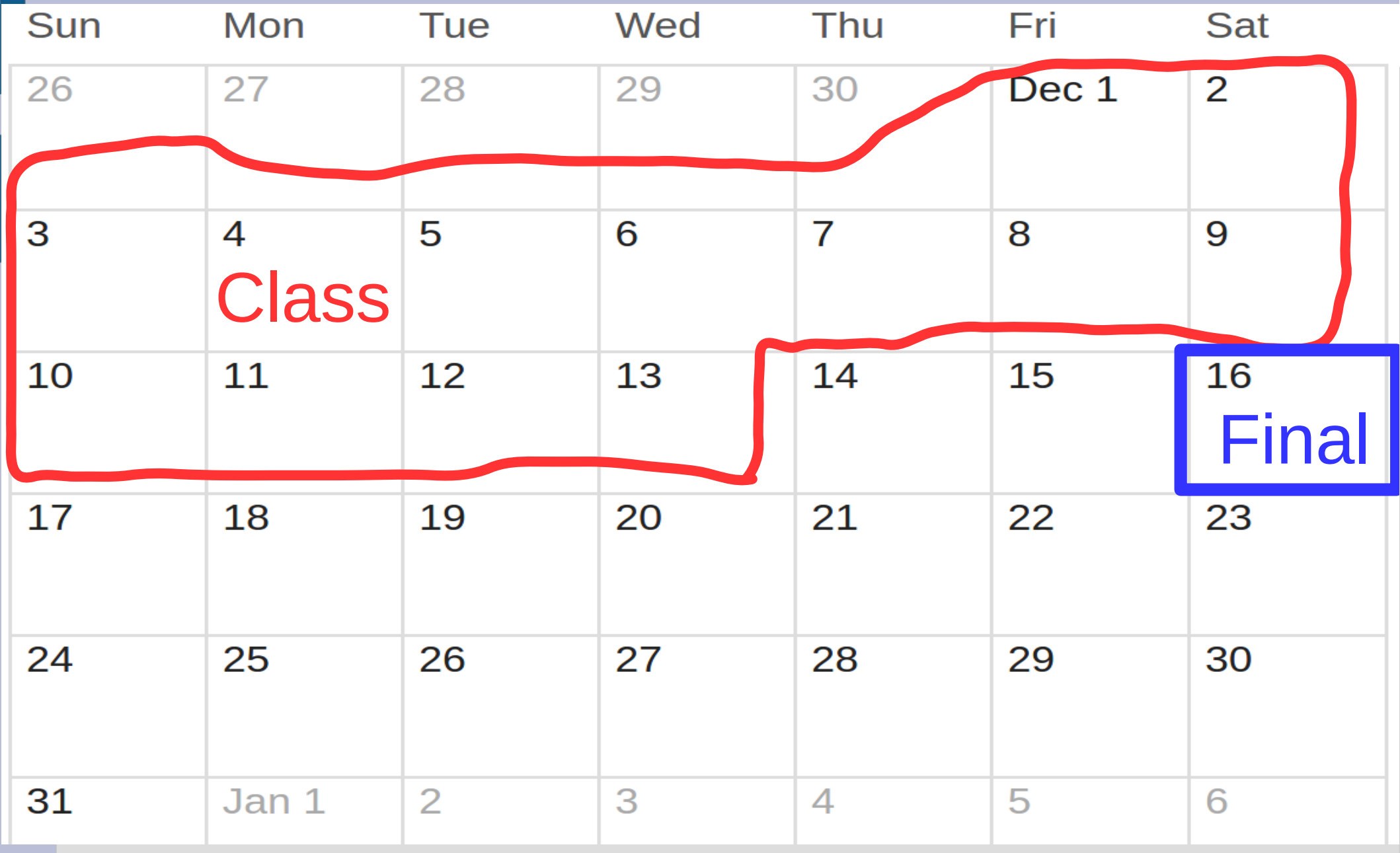


Syllabus

30% Homework
20% Programming assignments
25% Midterm (Oct. 24)
25% Final (Dec. 16)?

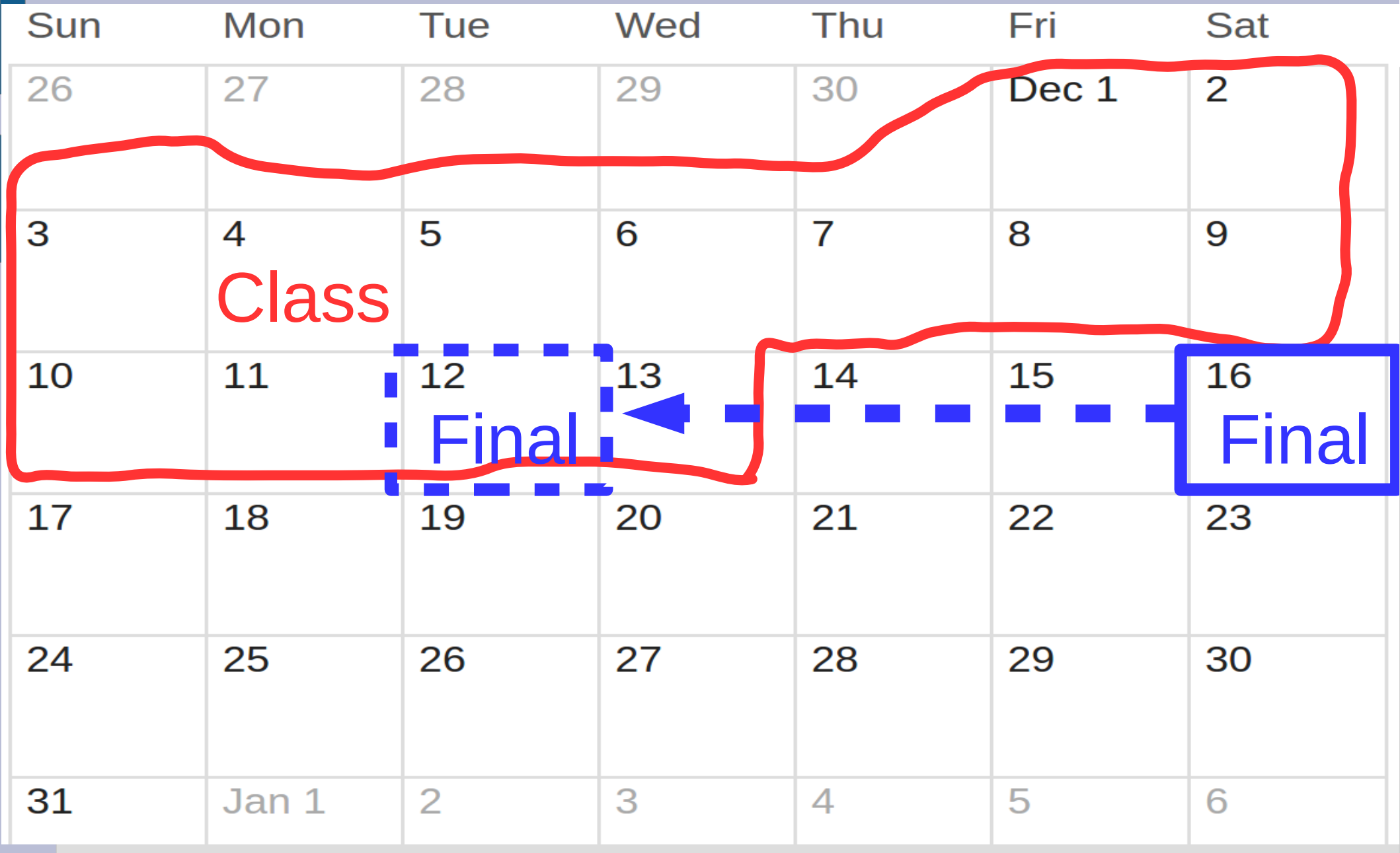
(No late homework; must ask for extension 48hr before deadline)

Syllabus



Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	Dec 1	2
3	4 Class	5	6	7	8	9
10	11	12	13	14	15	16 Final
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	Jan 1	2	3	4	5	6

Syllabus


Sun
Mon
Tue
Wed
Thu
Fri
Sat
26
27
28
29
30
Dec 1
2
3
4
5
6
7
8
9
Class
10
11
12
13
14
15
16
Final
Final
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
Jan 1
2
3
4
5
6

Syllabus

30% Homework
20% Programming assignments
25% Midterm (Oct. 24)
25% Final ~~(Dec. 16)?~~ (Dec. 12)

(No late homework; must ask for extension 48hr before deadline)

Programming vote

C/C++?

Java?

Python?

Syllabus

Grading scale:
93% A
90% A-
87% B+
83% B
80% B-
77% C+
73% C
70% C-
67% D+
60% D
Below F

Schedule

Ch. 1, 2, 3: Introduction
Ch. 2.1, 2.3, 7, 8: Sequences and Sets
Ch. 6, 9, 13, 32: More Sequences and Sets
Ch. 22, 23, 24, 25, 26: Graph Algorithms
Ch. 33: Geometric Algorithms
Ch. 4.2, 30, 31: Algebraic and Numeric Alg.
Ch. 34: NP-Completeness

Syllabus

Any questions?

Course overview

Major topics:

- Learn lots of algorithms
- Decide which algorithm is most appropriate
- Find asymptotic runtime and prove an algorithm works (mathy)

Algorithms

We assume you can program

This class focuses on improving your ability to make code run faster by picking the correct algorithm

This is a crucial skill for large code

Algorithms

We will do a pretty thorough job of sorting algorithms

After that we will touch interesting or important algorithms

The goal is to expose you to a wide range of ways to solve problems

Algorithms

Quite often there is not a single algorithm that always performs best

Most of the time there are trade-offs:
some algorithms are fast,
some use more/less memory,
some take use parallel computing...

Algorithms

A major point of this class is to tell how scalable algorithms are

If you have a 2MB input text file and your program runs in 2 min ... what if you input a 5MB file?

... 20 MB file?

Algorithms

In addition to using math to find the speed of algorithms, we will prove algorithms correctly find the answer

This is called the “correctness” of an algorithm (and often will be proof-by-induction)

Next time...

Thursday will be a “review” of things you should already know (Ch. 3)

We will often have ungraded in-class exercises, so please bring something to jot notes on