

Berkeley Time!

All lectures, sections, etc. will start 10 minutes past the official start time, as is Berkeley tradition! In the meantime, check out this fun word cloud of your classmates' interests and chat.



Lecture 1: Introduction

June 21st, 2021

A photograph of a modern building with a facade made of small, square tiles in various shades of gray and blue, creating a checkered pattern. Large, arched windows are visible on the upper floors. In the foreground, there are palm trees and a wooden pergola structure. The sky is clear and blue.

Welcome to CS 61A!

CS61A

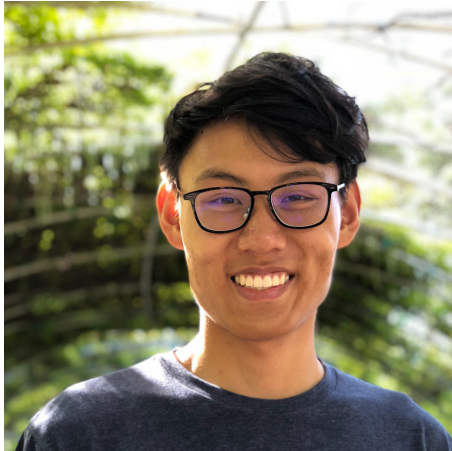
- 580+ students
- 19 timezones



Humans of CS 61A

Instructors

Albert Xu
(he/him/his)
albertxu3@berkeley.edu



Alex Kassil
(he/him/his)
alexkassil@berkeley.edu



Catherine Cang
(she/her/hers)
catherinecang@berkeley.edu



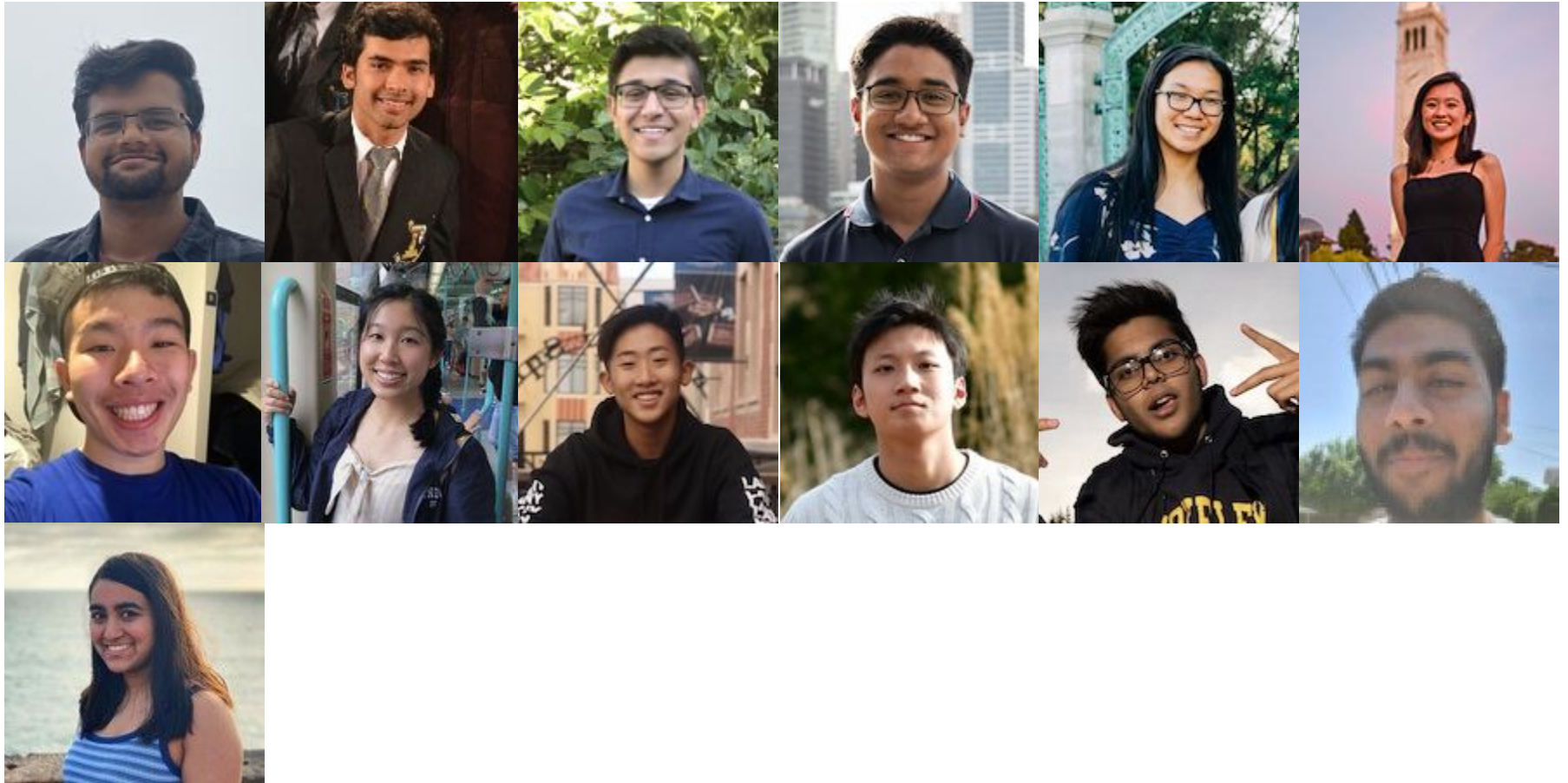
Prof. John DeNero



Teaching Assistants



Tutors



You!

Computer Science

What is Computer Science?

- What problems can be solved using computation?
- How do we solve those problems using computers?
- What techniques lead to effective solutions?

Systems

Artificial Intelligence

Graphics

Security

Networking

Programming Languages

Theory

...



Computer Vision

Natural Language Processing

Robotics

...

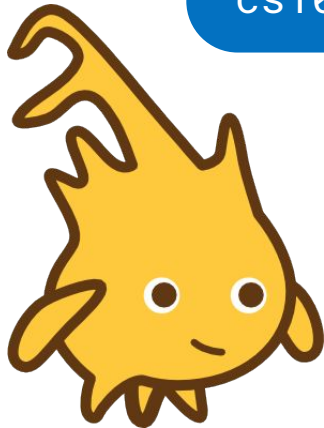
What is CS 61A?

- A course about managing complexity
 - Mastering abstraction
 - Programming paradigms
- An introduction to programming
 - Full understanding of Python fundamentals
 - Combining multiple ideas in large projects
 - How computers interpret programming languages
- A challenging course that will demand a lot of you

Alternatives to CS 61A

CS10: The Beauty and Joy of Computing

An introduction to fundamentals (& Python) that sets students up for success in CS 61A



cs10.org

Data 8 The Foundations of Data Science

Fundamentals of computing, statistical inference, & machine learning applied to real-world data sets

data8.org

<https://edstem.org/us/courses/5163/discussion/491818> Info about CS10 Reserved Spots for Summer 2021 CS61A Students

Course Logistics

Course Format

Lecture	Whenever you want!
Discussion	the most important part of this course
Lab Party	the most important part of this course
Office hours	the most important part of this course
Tutoring	the most important part of this course
Textbook	composingprograms.com

- 8 programming **homeworks**
- 4 programming **projects**
- 1 **diagnostic quiz**, 1 **midterm exam**, and 1 **final exam**
- Lots of course support and a great community

The First Week

- Lab 00 released already!
- Discussion starts tomorrow and Office Hours start tomorrow!
- Lab Party starts Wednesday
- Office Hours Schedule:

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
	8-10 AM	Albert's OH	8-10 AM		
Lecture Alex's OH	Lecture	Lecture	Lecture Alex's OH	11AM - 3PM Catherine's OH	
	4-8 PM	3-6 PM	4-8 PM		4-6 PM
		7-10 PM			

Lecture

- Two options for lecture:
 - pre-recorded lecture videos by Professor John DeNero
 - Short, few-minute videos per topic
 - Live lectures given by us Monday-Thursday 12:30-2PM
 - 1:20 minutes long, will answer student questions live!
 - Missed live lecture but still want to see *us*? We'll also post a recorded live lecture.
- Which option is better for me?
 - If you like asking questions during lecture, come to live lecture!
 - If you like sticking to a pre-defined schedule, come to live lecture!
 - If you can't attend live lecture or prefer to learn at your own pace, watch the pre-recorded lecture
- **Important!** The pre-recorded lectures and live lectures will cover *the same* material. Do not attend both. It is not a good use of your time.

Your Role in Lecture

- Come to lecture!
- Be respectful of other students.
- Talk in Chat
 - The benefit of learning on Zoom is being able to talk with your peers (and us) in real-time!
- Ask questions on Live lecture thread - or answer questions!
 - <https://edstem.org/us/courses/5163/discussion/492271>

Discussion Section

- Only part of the course that gives points for attendance
- 90 minute section twice a week
- Conceptual review + Online worksheet
- Three options:
 - Standard Sections
 - 1.5 hours
 - Limited Experience (LE)
 - 2 hours
 - Covers same material as Standard Section at a slower pace
 - Exam Prep Sections (EP)
 - 1.5 hours
 - No conceptual review
 - Emphasis on problem solving, exam prep, and more difficult problems
- If you filled out the welcome survey, you'll have a discussion assigned this afternoon.

Lab Party

- Lab party happens twice a week. *You can attend any section you choose:*
 - Monday/Wednesday, 4-5PM
 - Monday/Wednesday, 5-6PM
 - Tuesday/Thursday, 10AM-11AM
 - Tuesday/Thursday 11AM-12PM
- 1 hour long each. We encourage you to come in at the top of the hour, but you can also drop-in later if you have specific questions.
- Come to work on lab with other students and get help from staff!
- Feel free to hang out here and listen to others' questions.



Office Hours

- Three formats: OH Queue, Parties, and Instructor
- Office Hours Queue:
 - Queue-based
 - Great for conceptual help, assignment help, and exam prep!
- Parties:
 - 1 hour each
 - 2 types: Lab and Project
 - Work with other students and get help from staff!
- Instructor:
 - Whatever you want to chat about!
 - Great for both content questions and things outside of CS61A, like life, Berkeley, internships, research, etc

Small Group Tutoring Sections [optional]

Recurring small-group sections (5-8 students) centered around a worksheet which reviews content from the corresponding discussion

- Meet twice a week (Wed and Fri) regularly with the same group of students
- If you sign up for a section, you must attend twice every week.
- Please notify your tutor if you need to miss a section, or you might be dropped from the section.
- If you aren't able to snag a section, please check back later. Spots will open!

Sign-ups will open after Tuesday's lecture. Sections will start Wednesday this week!

Tools

Zoom: A platform for video calls

- Can ask questions via voice or text-chat
- Option to ask questions individually in a “breakout” room
- Where lectures, discussions , Lab/Project parties will happen

Ed

- Forum for students to post questions & get announcements from staff

Various pieces of software (like Python, Text Editor, and Terminal)

- Will introduce most of these in Lab 00, so complete this ASAP!

Important Websites

cs61a.org

- Hub for all things related to CS 61A! Lectures, assignments, etc

[edstem.org > CS 61A](https://edstem.org/CS61A)

- Ask questions and view announcements here

composingprograms.com

- Read the free, online textbook here

okpy.org

- Submit all your coding assignments here

howamidoing.cs61a.org

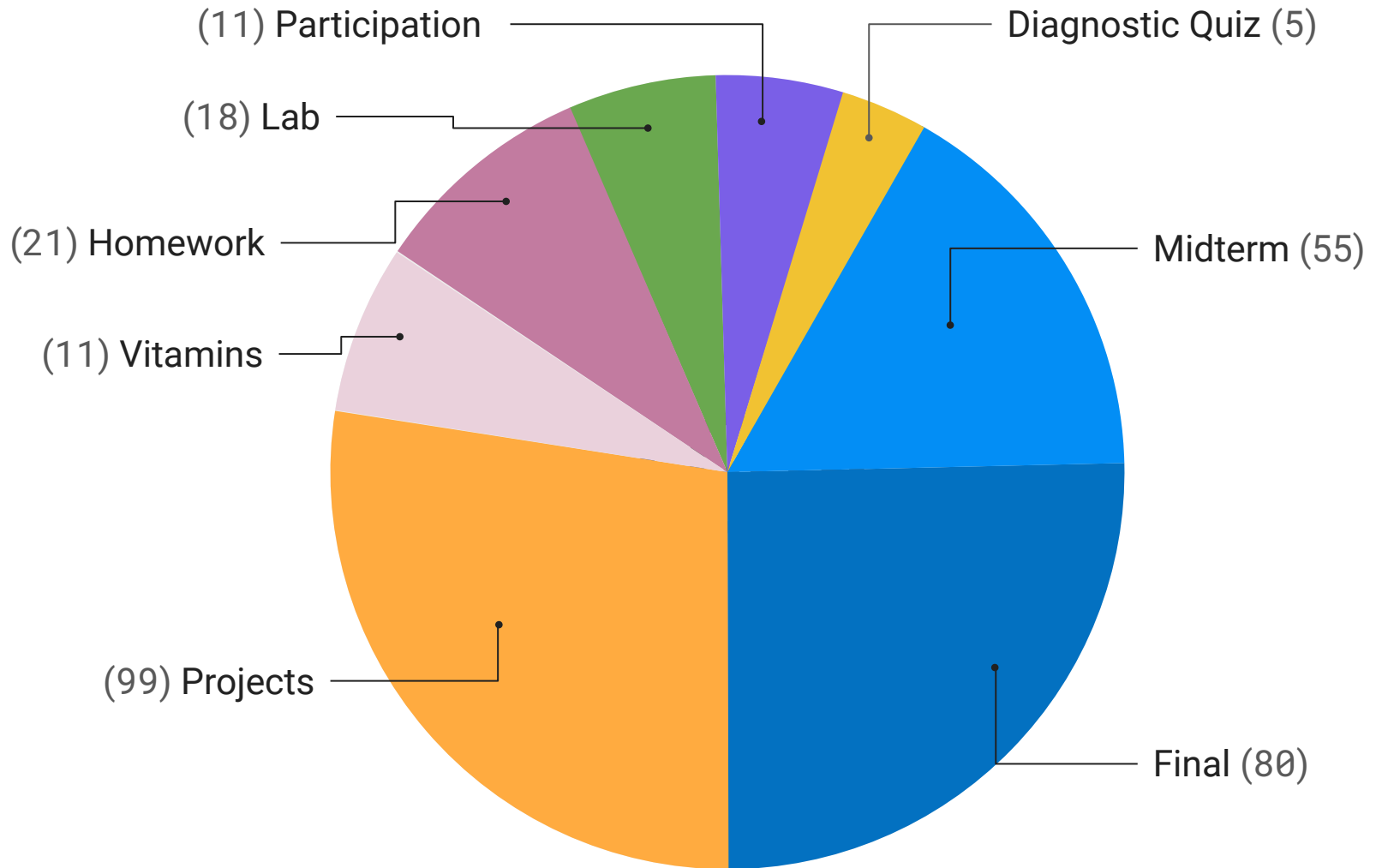
- Check your grades in a convenient online interface here

oh.cs61a.org

- Get help in Office Hours here

Soon: Websites to find your Discussion + Tutoring times!

Grading



Binned, Not Curved

- CS61A is not curved
- Your grade is based on how many points you earn over the course of the semester, out of 300
- There will be a handful extra credit points throughout the summer, perhaps around 8, that will be available to everyone.
- We encourage you to work with others and collaborate. This class is not a competition
- More details <https://cs61a.org/articles/about/#grading>
- Past semester grade distributions can be found on berkeleytime.com

A+	≥ 300	A	≥ 285	A-	≥ 270
B+	≥ 250	B	≥ 225	B-	≥ 205
C+	≥ 190	C	≥ 180	C-	≥ 175
D+	≥ 170	D	≥ 165	D-	≥ 160

Assignments

You can earn **149 points** through assignments:

- (14) 2 pt. Twice a week lab assignments
- (8) 3 pt. weekly programming homework assignments
- (4) 20-30 pt. programming projects
- (13) 1 pt. Twice a week mini quizzes called vitamins

Most assignments are submitted using Ok (okpy.org).

Vitamins will be google forms

You must have an @berkeley.edu address listed as your primary email on CalCentral to be enrolled on Ok.

If you have not been added to OK--which you'll find out when you do your assignments--fill out the form on Ed (edstem.org) ASAP

Lab Assignments

- 2 Lab assignments per week corresponding to Mon/Wed, due Tue/Thur respectively
- Graded on correctness, 2 points, all or nothing
- Lowest two lab grades dropped
- You are highly encouraged to come to lab party and get help!
 - Monday/Wednesday afternoon, 4-6PM
 - Tuesday/Thursday morning, 10AM-12PM
- More details: <https://cs61a.org/articles/about/#labs>

Lab parties are the best place to work on lab!

Homework Assignments

- Generally released on Thurs, due next Wed
- Graded on correctness, 3 points, but with partial credit. Each incorrect answer loses you one point on the homework. Minimum score is 0
- **Homework Recovery:**
 - You can recover one incorrect question per homework by going through our weekly homework recovery session
- More details: <https://cs61a.org/articles/about/#homework>
- Homework 01-07 are graded. Homework 08 is for extra credit

Discussion Participation

- This is part of 300 points.
- Attending each discussion grants one participation point
- You can earn up to **11 participation points**.
- We require all students to have cameras on during discussion section
 - Fill out this form to request to opt out [links.cs61a.org/camera-opt-out](https://cs61a.org/camera-opt-out)
- Also used for exam recovery! Read <https://cs61a.org/articles/about/#exam-recovery> for more info

Vitamins

- Short Google Forms quizzes meant to keep you on-track with lecture
- We ask you to complete the vitamin *before* each discussion.
 - The topics covered on the vitamin will reflect the topics covered in the lectures preceding that discussion.
- We will have a total of 13 vitamins, and you will need to complete 11 of them for full credit.
- Graded on completion
- <http://links.cs61a.org/vitamin0> First vitamin is released!
- Generally they will be due Tuesday 8am PT and Thursday 8am PT before the first discussion
- The vitamin for discussion 0 is due Thursday 8am PT.

Exams

Diagnostic Quiz (5 pts)

When: Friday, July 2 @ 5-6 PM PT

Format: 50 minute electronic exam

Midterm (55 pts)

When: Thursday, July 15 @ 5-7 PM PT

Format: 110 minute electronic exam

Final exam (80 pts)

When: Thursday, August 12 @ 5-8 PM PT

Format: 170 minute electronic exam

Exam Recovery

For each discussion you attend, you'll get a recovery point, allowing you to recover points on the midterm. Maximum of 13 recovery points!

Alternates

If you live in a time zone where the exam starts between 8PM and 6AM, have a conflict, or for some other reason cannot make the normal time, fill out:

links.cs61a.org/diagnostic-alt

links.cs61a.org/midterm-alt

links.cs61a.org/final-alt

Collaboration

- This course is not curved -- collaboration, not competition, is key
- Asking questions and discussing ideas is highly encouraged
- **The only students whom you can share code with are**
 - Your project code with project partner
 - Students who have finished the problem you are working on
 - Your lab code with anyone in the course as long as everyone is learning
- More info: <https://cs61a.org/articles/about/#academic-honesty>

Course Overview

Every part of the course will center around a theme of Computer Science

- Learn the fundamentals of programming
- Become comfortable with Python



Introduction

Functions & Environments

Data Abstraction

Objects

Evaluation

Declarative Programming

Expressions

What do computer programs do?

- Programs work by manipulating **values**
- **Expressions** in programs evaluate to values
 - Expression: $1+2$
 - Value: 3
- The Python interpreter evaluates expressions and displays their values

$$20 + 21$$

$$2^{100}$$

$$\sin \pi$$

$$\lim_{x \rightarrow \infty} \frac{1}{x}$$

$$f(x)$$

$$\frac{20}{21}$$

$$\sum_{i=1}^n i$$

$$\log x$$

$$\sqrt{2021}$$

$$\binom{n}{x}$$

$$-2021$$

An **expression** describes a computation and evaluates to a value.

Call Expressions

Evaluation procedure for **call expressions**

1. Evaluate the **operator**
2. Evaluate the **operands** from left to right
3. **Apply** the operator (a **function**) to the evaluated operands (**arguments**)



Operators and operands are also expressions

So they also *evaluate to values*

add(add(6, mul(4, 6)), mul(3, 5))

Operator Operand Operand

???

Nested Call Expressions

- Humans evaluate inside-out

add(add(6, mul(4, 6)), mul(3, 5))

add(add(6, 24), mul(3, 5))

add(add(6, 24), mul(3, 5))

add(30, mul(3, 5))

add(30, mul(3, 5))

add(30, 15)

add(30, 15)

45

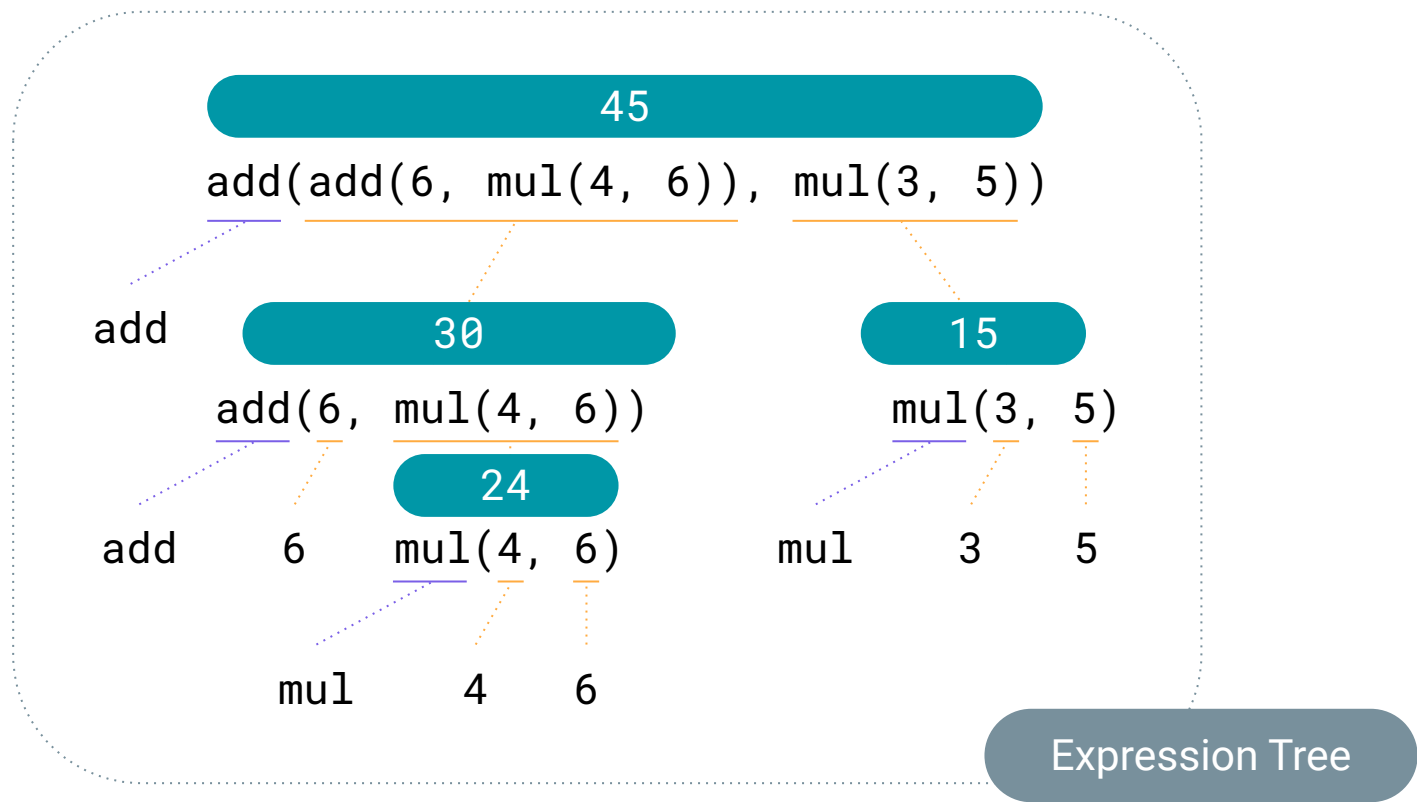
- We can jump ahead or skip around, but Python can't do that!
- How does the computer know which call to evaluate first?

Nested Call Expressions

1 Evaluate operator

2 Evaluate operands

3 Apply!



Functions, Values, Objects, Interpreters, and Data