

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. Feel free to answer this question!

How many pigeons do you think you
could you hold if they were cooperative?

Slides: go.cs61a.org/lec1

Lecture 1: Introduction

June 20th, 2023

Jordan, Noor, Tim

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!

Humans of CS 61A

Instructors

Jordan Schwartz
(she/they)

jordanschwartz@berkeley.edu



Mahnoor Haq
(she/her/hers)

mhaq01@berkeley.edu

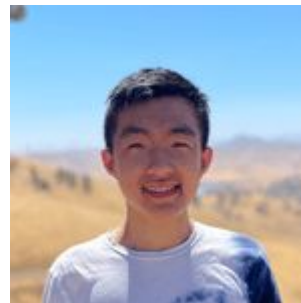
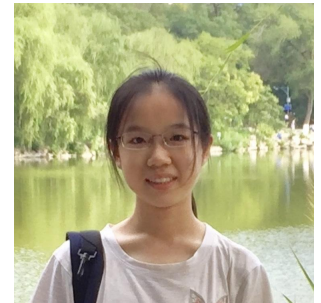
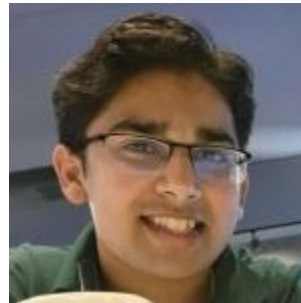
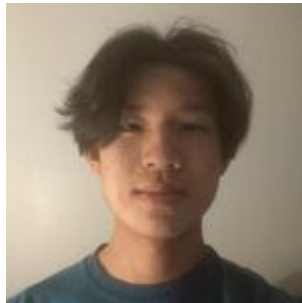
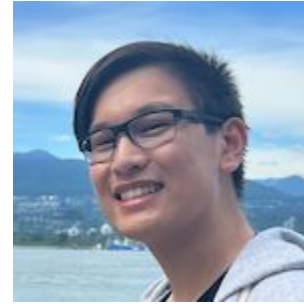


Tim Tu
(he/him)

timothyktu@berkeley.edu



TAs!



Tutors!



You!

490+ students

40+ majors

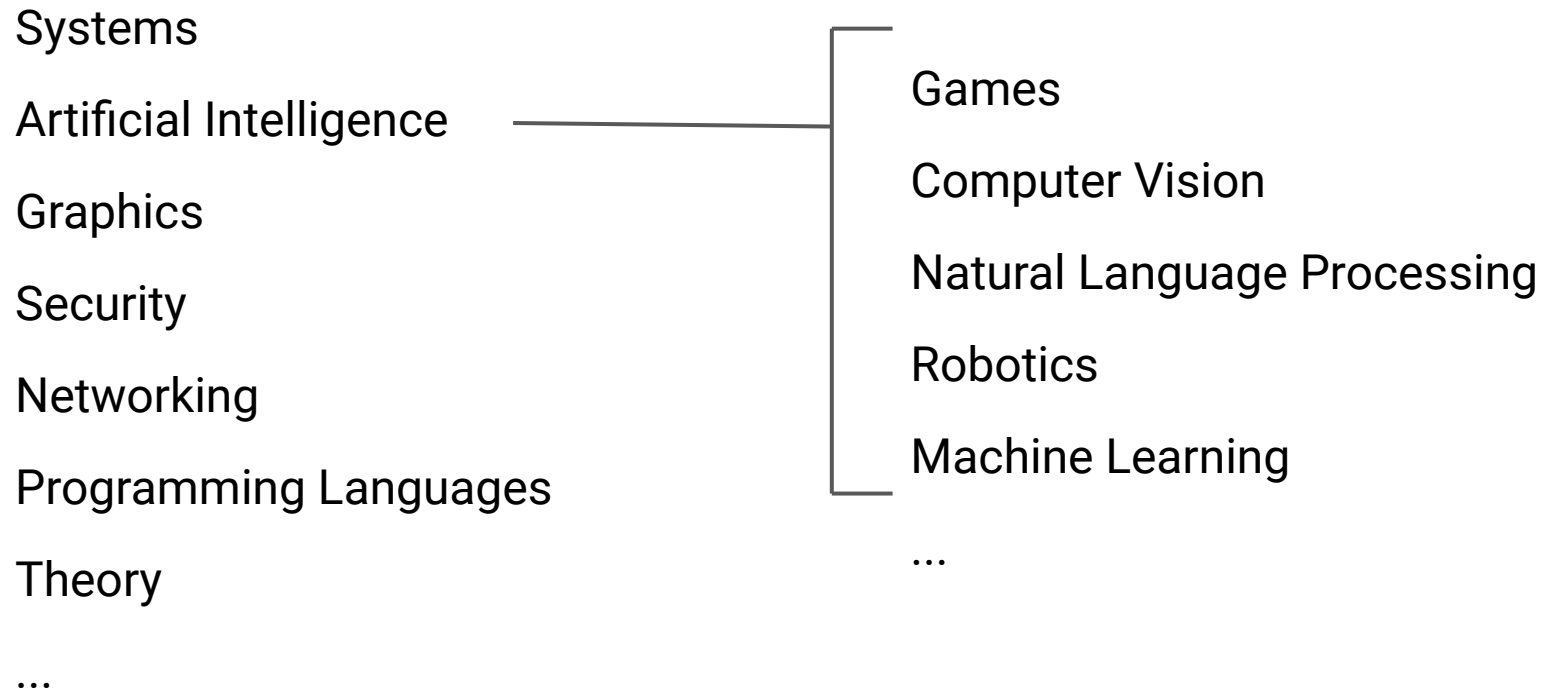
**UC Berkeley Students, Visiting
Students, High Schoolers**

Introduce Yourself!

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?



What is CS 61A?

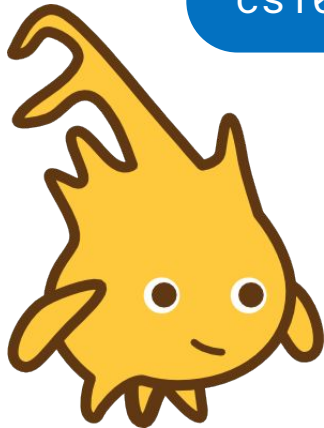
- A course about managing complexity
 - Mastering abstraction
 - Programming paradigms
- An introduction to programming
 - Full understanding of Python fundamentals
 - Combining multiple ideas in large projects
 - How computers interpret programming languages
- Different languages: Scheme & SQL
- A challenging 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

Course Format

Course Format

| | |
|---------------------|--|
| Lecture | MTWTh 11am-12:30pm in Dwinelle 155 |
| Discussion | the most important part of this course |
| Lab | 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 **midterm exam** and 1 **final exam**
- Lots of course support and a great community

Lecture

- Time/Location: 11am-12:30pm in Dwinelle 155 on M/T/W/Th
- First introduction to the material taught by instructors
- Nothing you need to prepare for lecture, just be ready to listen and take notes!
- You will have opportunities to ask questions about the material
- Will be recorded so you can re-watch later
- There will be thread on Ed for each Lecture
- Alternative: Pre-recorded lectures by Professor John DeNero
 - These will not have announcements and may not be up to date

Lab

- Time/Location: Lots of different times on M/W
- 90 minutes, twice a week
- Practice working with the material with brief review, good practice for homework and projects
- Assignment completed on your computer
- Lab section run by your TA
- You can work with other students on the problems and get help from TA and Academic Interns (AIs)
- Attending 10 lab sections is required

Lab Assignments

- Usually more introductory to the material.
- Attending lab section, during which you'll work on the lab assignment, is required.
 - Completion and submission of the lab assignment to receive credit
- Each lab is worth 1 point and you can receive a maximum of 10 points, but there will be more than 10 labs
- ~2 days to complete each lab
- Lab00 is worth 2 points and has 2 Gradescope submissions
 - This is the only lab you are not allowed to drop

Discussion

- Time/Location: Lots of different times/locations on T/Th
- 90 minute sections, twice a week
- Practice working with the concepts more in the direction of what might seem on an exam
- Taught by your TA with more conceptual review & mini-lectures
- Nothing to turn in, just practice with the material
- Cannot be done independently, must come to section to get attendance credit
- Attending 10 discussions is required for full credit
- See your Lab & Discussion on sections.cs61a.org – open now!

Mega section

- Students who choose to enroll in the mega section will not need to attend discussion or labs in-person
- There will be video recordings posted on the home webpage walking through discussion worksheets that can be watched asynchronously
- Labs can be completed at your own time and must be submitted by the final deadline
- Deadline to switch out of mega section and enroll in a regular section is 6/30
- If you opt to enroll in mega section, you don't need to sign up for a section on sections.cs61a.org

Homework

- Often more difficult problems (conceptually or implementation-wise).
- Each homework assignment is worth 2 points
- You will get 1 homework drop
- Released Monday, due Thursday of the same week, in general
 - ~4 days to complete each homework

Office Hours

- Time/Location: Lots of different times/locations throughout the week (including online)
- This is a chance to get 1-on-1 help with the assignments during the week
- 2 options to get help
 1. Come in person and sign up on the online queue and get help when it's your turn
 2. Some office hours will be remote
- TAs/Tutors/Als will spend about 10-15 minutes helping you out with your problem from lab, homework, or project

Small Group Tutoring

- Optional small group tutoring will be provided to students
- 1 hour, twice a week
- These sections will be taught by a tutor with 4-6 students in a section
- It will be a supplement to lab/discussion, not a replacement
- Great chance to practice the material and problem solving more
- Sign up information will be out later this week

Projects

- Contains many problems that work together to build a project (such as a game, or later, an interpreter).
- More difficult and longer than homeworks, labs
- ~1.5 weeks to complete each project with some checkpoints before the deadline
- These will be group (pairs of 2) projects

Optional Assignments

- If you have time and want an excuse to work on some fun project prompts, we will have some optional assignments

Weekly Schedule

| | Monday | Tuesday | Wednesday | Thursday | Friday |
|-----------------|----------|--------------|-----------|--------------|--------|
| Morning | Start HW | Complete Lab | | Complete Lab | |
| 11:00AM-12:30PM | Lecture | Lecture | Lecture | Lecture | |
| Afternoon | Lab | Discussion | Lab | Discussion | |
| Evening | | | | Submit HW | |

Exams

Midterm:

- 64 points of your grade
- Roughly halfway through the summer session (7/13 7-9PM)
- ~ 2 hours

Final:

- 96 points of your grade.
- At the end of the summer session (8/10 6-9PM).
- Cumulative (as the course itself is also fairly cumulative in its ideas).
- ~ 3 hours.

We will have a clobber policy! (earn up to 90% of the points back)

Important Websites

cs61a.org

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

edstem.org

- Ask questions and view announcements here

composingprograms.com

- Read the free, online textbook here

gradescope.com

- Submit assignments here
- Check your grades in a convenient online interface here

oh.cs61a.org

- Get help in Office Hours here

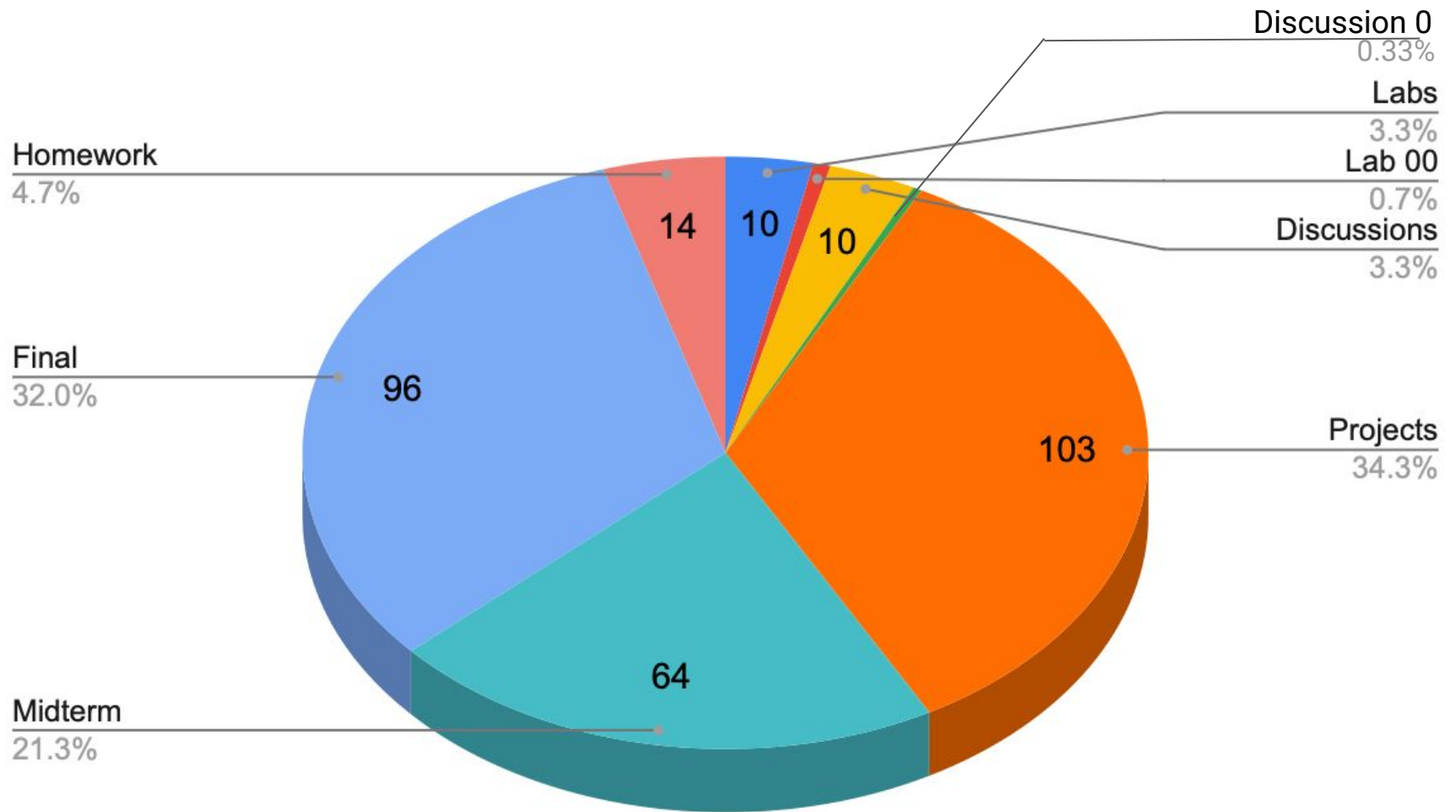
Grading

- Grading bins, not curved (everyone could get an A)
- These will not change

| | | | | | |
|----|-------|---|-------|----|-------|
| | | A | ≥ 290 | A- | ≥ 275 |
| B+ | ≥ 255 | B | ≥ 230 | B- | ≥ 210 |
| C+ | ≥ 190 | C | ≥ 180 | C- | ≥ 175 |
| D+ | ≥ 170 | D | ≥ 165 | D- | ≥ 160 |

- Students can qualify to earn an A+ if they earn more than 295 points in the course and solve the A+ question(s) on exams
- Extra credit: there will be opportunities so be on the lookout for them

Points Breakdown



Academic Misconduct

- We actively encourage collaboration in this class!
- Do:
 - Discuss concepts
 - Share approaches to problems
 - Problem solving, debugging, exam tips
 - Share lab code & share project code with your project partner
- Do not:
 - Share (send nor receive) code with students who are not your project partner or the internet
 - Use AI assistants such as chatGPT, Copilot, Codex, etc.
 - Collaborate on an exam → **F**
- Please be honest! We do check!
- Rather than copying someone else's work, ask for help in Office Hours or Ed!

Extensions and Accommodations

DSP Program and Accommodations

- The DSP (the Disabled Students Program) program is a service at UC Berkeley that advocates and provides support for students with disabilities
- We require students to be enrolled in the DSP program in order to provide accommodations in 61A for equity reasons
- If you are not enrolled in DSP but believe you are eligible for accommodations, you can visit <https://dsp.berkeley.edu/> to begin the process of getting enrolled.
- We also recommend you book an accommodations appointment in the meantime
- Here is a list of accommodations DSP provides: go.cs61a.org/acc-list
- Here is another slideshow about DSP written by DSP students:
go.cs61a.org/dsp-guide

DSP Accommodations

- If you are enrolled in DSP, please send us your letter in AIM
- We will provide accommodations in line with what is given to us in your letter—this most commonly includes increased flexibility on extensions or extended time on exams
- When we have received an accommodations letter for you, you will receive an email from us confirming that and what your provided accommodations will look like—if you do not receive such an email from us, please reach out (this may take some time, we are human too. Emails from us will begin to be sent out 6/24)
- If your letter of accommodations changes this summer, please email us, otherwise we will not know about your updated accommodations

Extensions

- We have a process for extensions! Sometimes life happens and we want to make sure that extenuating circumstances don't negatively impact your ability to succeed in the class
- You can request an extension by filling out <https://go.cs61a.org/extensions> (also linked on the "Contact" page of the course website). All extension policies are available in this form, **please read carefully**
- Any extension for 24 hours (1 day) past an assignment deadline will be granted, requested before the original deadline; Any extension for up to 72 hours (3 days) past an assignment deadline, made by a DSP student with an accommodation for assignment extensions, will be granted
- We will be flexible and reasonable with longer extensions—our goal is to support your well-being and ensure that you're keeping up with the class, not to enforce arbitrary deadlines; it's always best to just submit a request and it will be handled on a case by case basis ASAP
- This class over the summer moves very quickly. If you feel as though you need a lot of extensions or some extenuating circumstance is impacting your ability to complete assignments on time, please book an accommodations appointment to discuss further possible arrangements

Accommodations Appointments

- You can book a short appointment to discuss accommodations at <https://go.cs61a.org/acc-appt> if you believe extenuating circumstances will affect your ability to perform in the class. This will give you the opportunity to meet with one of our student support TAs to talk about ways we can make this class work best for you
- You can book an appointment for any reason, especially regarding personal, family, or medical emergencies. We also recommend that you book an appointment if you believe that you are eligible for DSP accommodations and are currently going through the process of enrolling in DSP, as we can discuss ways to support you intermediately while you are enrolling

Support Resources (Academic and Otherwise)

Finding resources

- We have a lot of study resources linked here on our course site!
<https://cs61a.org/resources/>—this is great if you're looking for extra review materials
- We also have a collection of campus resources linked here—<https://cs61a.org/articles/campus-res/>, which can be helpful for support in all kinds of things, from adjunct tutoring to basic needs services

Tutoring

- We will have supplemental tutoring sections this summer
- There will be small group tutoring where you will be put in 4-6 person groups and will meet with a tutor twice a week for an hour and do a worksheet
 - Students historically have signed up for these sections if they feel like they want more time to work on the content and get more practice
- Two types of section: Discussion Review and Exam Prep
 - Discussion Review: Covers discussion questions not covered in discussion section; designed as a good second pass of content
 - Exam Prep: Covers exam-level questions from previous semesters

Advising OH

- Most office hours are reserved for questions that directly pertain to course content, however, some TAs offer advising office hours as well—you can find more information at <https://cs61a.org/articles/advising/>
- These are a great opportunity to ask about study strategies, do a grade check-in, talk about research or internship opportunities, planning for future CS courses, or anything else you want to ask about!

Class Climate

In this section we will be briefly discussing sexual harassment and discrimination—if you would not feel comfortable being present for this part of lecture you may use this time to step out (we'll take a short break)

Quick Break

Sexual Harassment

- Often, people don't realize they are sexually harassing someone. The behavior still has the same impact and **impact is what matters**
- What is it?
 - Unwanted sexual or romantic advances:
 - Repeatedly asking someone out
 - Unwanted physical contact
 - Unwanted comments on someone's physical appearance
- See a full list in the [UCOP policy on SVSH](#)

Professional Boundaries

- While the summer session is happening, you should maintain professional boundaries with course staff (Tutors/TAs/As)
- Don't message them on social media or dating apps, don't give them physical compliments, don't flirt with them
- Just let them do their job

Reporting SVSH

- All faculty and staff members are Mandated Reporters. If we ever receive an incident report, we will need to make a report to OPHD—we cannot act as confidential resources
- Two goals to reporting
 - Getting you the support you need
 - Increasing our awareness of incidents so that we can try to make improvements

Where to Report

- [Office for the Prevention of Harassment and Discrimination](#): Includes an online reporting form
- [Path to Care](#): Includes 24/7 care line and advocate appointments
- [CS 61A Anonymous Feedback Form](#): If you want to stay anonymous but make the instructors aware of something that happened in the course
- [EECS Anonymous Climate Form](#): This will make the EECS department aware of any issues. You can also contact Susanne Kauer (skauer@berkeley.edu) directly.

Racism

- In previous semesters of CS 61A, students reported that other students made racist comments suggesting that they did not belong in CS
- We live in a society with a long history of racism and need to actively combat that in both our actions and language

Community Values

- From the [Berkeley Principles of Community](#):
 - “We affirm the dignity of all individuals and strive to uphold a just community in which discrimination and hate are not tolerated”
- From the EECS Mission:
 - “Diversity, equity, and inclusion are core values in the Department of Electrical Engineering and Computer Sciences. Our excellence can only be full realized by faculty, students, and staff who share our commitment to these values. EECS’s mission is to serve the communities to which we belong, at local, national, and international levels, with a deep awareness of our ethical responsibilities to our profession and to society”
- We need to bring more people into CS, so that tech can create a better future for all

Behavior We Want to See

- **Helping each other** understand concepts in the class, whether in section, Ed, or study groups, without expectation of anything in return
- **Saying “congratulations” to classmates** when they finish an assignment, receive a job offer, get into a club, etc.
- **Being a great partner while pair programming.** If driving, listen to what your partner suggests and consider their suggestions. If navigating, brainstorm ideas for how to tackle the problem, re-read the project description, check Ed, etc.
- **Recognize that we’re all valuable members of the CS community!**

Expressions

$$20 + 23$$

$$\frac{20}{23}$$

$$\sin \pi$$

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

$$f(x)$$

$$\log x$$

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

$$20.23$$

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

$$-2023$$

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

Expressions in Python

- The Python Interpreter evaluates expressions and displays their value

```
>>> 1 + 2      # add 1 to 2, output the result
```

```
3
```

```
>>> 3 ** 2     # square 3, output the result
```

```
9
```

$$20 + 21$$

$$2^{100}$$

$$\sin \pi$$

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

$$f(x)$$

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

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

$$\log x$$

$$\sqrt{2021}$$

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

$$-2021$$

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

What is a computer program?

- We will use **programs** to do manipulation of **values**
- **Expressions** in programs evaluate to values
- These expressions can be represented by **function calls** in Python

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

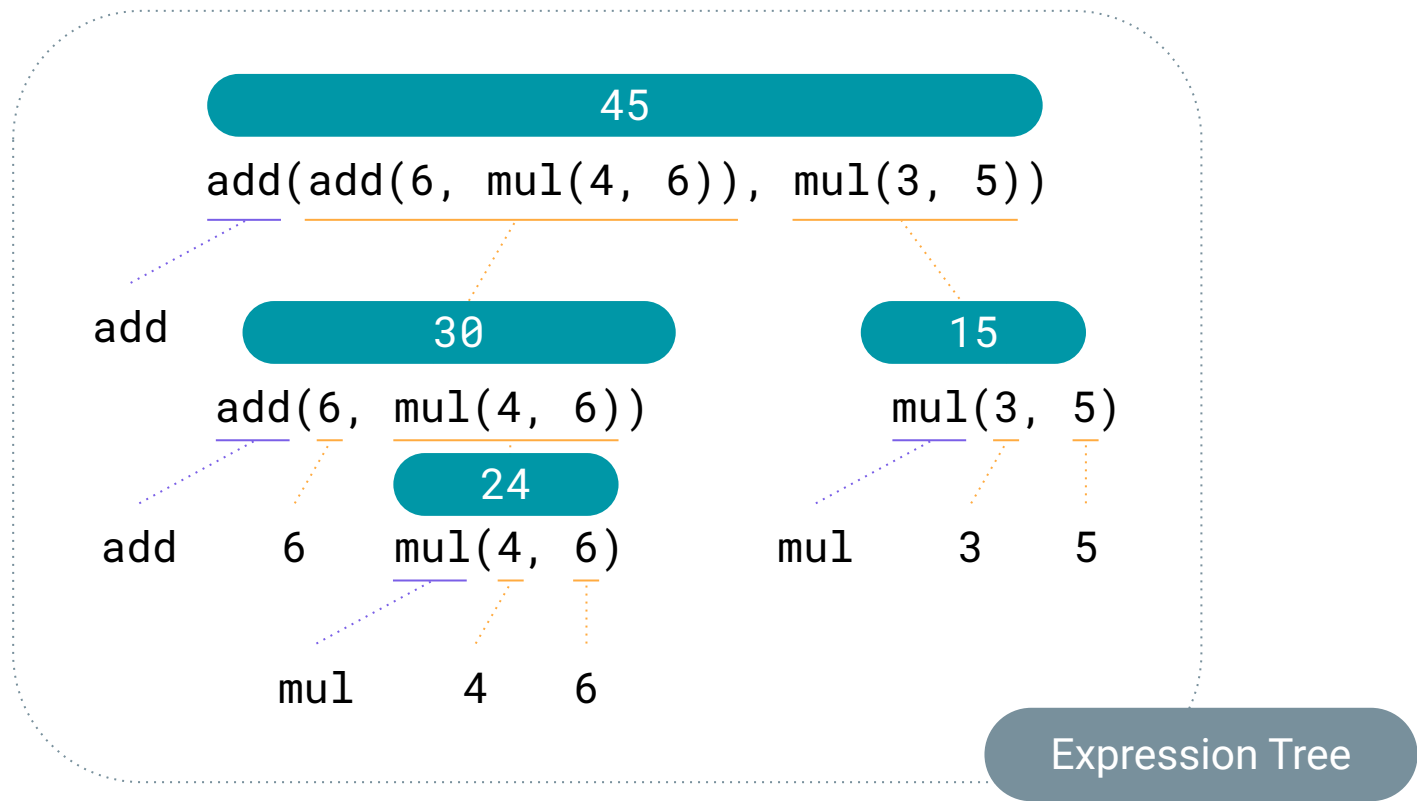
- We can jump ahead or skip around, but Python can't do that!
- How does the computer know which call to evaluate first?

Nested Call Expressions

1 Evaluate operator

2 Evaluate operands

3 Apply!



Values, Expressions, Objects, and Data

What's happens next?

- Today: Sign up for & Attend Discussion 00 (if you haven't already)
- Tomorrow: Lecture 2, Lab 00
- Thursday: Lecture 3, Discussion 01, Technical OH
- Friday: Lecture 4
- Ask any questions you have after lecture today or on Ed
- Instructor OH begins next week!

Let's have a great semester!