Agenda

1. Icebreaker
2. Course / section intro
3. Go over mini-assignments, Q&A.
4. Analysis of algorithms.
5. Big-O Proof
Section Intro

- Section is a great place to learn and reinforce material!
- Other resources:
  - Mentorship meetings - meet one-on-one with TAs
  - Hours and Piazza
  - Clinic hours - work together with other students on problems
  - Conceptual hours - quicker group TA hours to learn concepts
  - Anonymous questions form for section! (You can ask anonymous questions to your section TAs [here](#))
Mentorship

- This year, we’re opening up the option for students to find mentors that they might find more closely related to their interests and or identities.
  
  FORM HERE: [https://tinyurl.com/cs16mentormatch](https://tinyurl.com/cs16mentormatch)

  - **Pros for defaulting:** *We highly suggest defaulting to your section TAs.* Students in the past have told us they felt more comfortable developing a relationship with their section TAs, whereas meeting a TA with whom they have no experience with can be very impersonal. If you choose to go with one of the section TAs, you don’t need to fill out this form!

  - **Pros for opting-in:** If you have a very specific interest or question about CS in the dept. that you believe can only be answered by certain TAs, this might be better for you.

- **FORM DUE DATE: THIS SUNDAY AT 5PM.**
Section Expectations

● Students will **NC** the course if they miss more than **3 sections**
● To switch, email TAs **before** your section time
● Bring mini assignments to all sections
Mini Assignment

Modulo problems:

1. $60 \% 6 = 0$
2. $368 \% 13 = 4$
3. $4901 \% 172 = 85$
Analysis of Algorithms
Finding Big-O Runtime

Problem 1: What is the runtime of this function?

```python
run_race(x, y, z):
    for i in range 0 to x:
        for j in range 0 to y:
            print "LAP COMPLETED"
    return 3*z
```

Problem 2

Write a function `sum_list` that takes in a list and returns the sum of all elements in the list. What is the runtime of this function.
Prove that $f(n) = n^2 + 5n + 7$ is $O(n^2)$

**Definition (Big-O):** $T_A(n)$ is $O(T_B(n))$ if there exists positive constants $c$ and $n_0$ such that:

$$T_A(n) \leq c \cdot T_B(n)$$

for all $n \geq n_0$
Ethics Discussion
“Move fast and break things. Unless you are breaking things, you aren’t moving fast enough”

“If you've got a good idea, and it's a contribution, go ahead and DO IT. It is much easier to ask for forgiveness than it is to get permission.”

“The computer was born to solve problems that did not exist before.”
“Move fast and break things. Unless you are breaking things, you aren’t moving fast enough.”

“If you've got a good idea, and it's a contribution, go ahead and DO IT. It is much easier to ask for forgiveness than it is to get permission.”

“The computer was born to solve problems that did not exist before.”

- What are the pros and cons of this mindset in CS?
- How do you see a culture like this affecting ethics in computer science?