Lecture 1:

INTRODUCTION TO COMPUTER SCIENCE

CSC111: Introduction to CS through Programming

R. Jordan Crouser

Assistant Professor of Computer Science

Smith College

jcrouser.github.io/CSC111

Introductions

Jordan

Professor

David

Lab Instructor

Anastasia, Grace, Logan, Ojaswi,
Jenny, Georgina, & Prayasha

(our team of **T**eaching **A**ssistant**s**)

What I do: analytical tools for **messy data**

Big idea behind my research

Humans and machines have **complimentary strengths**

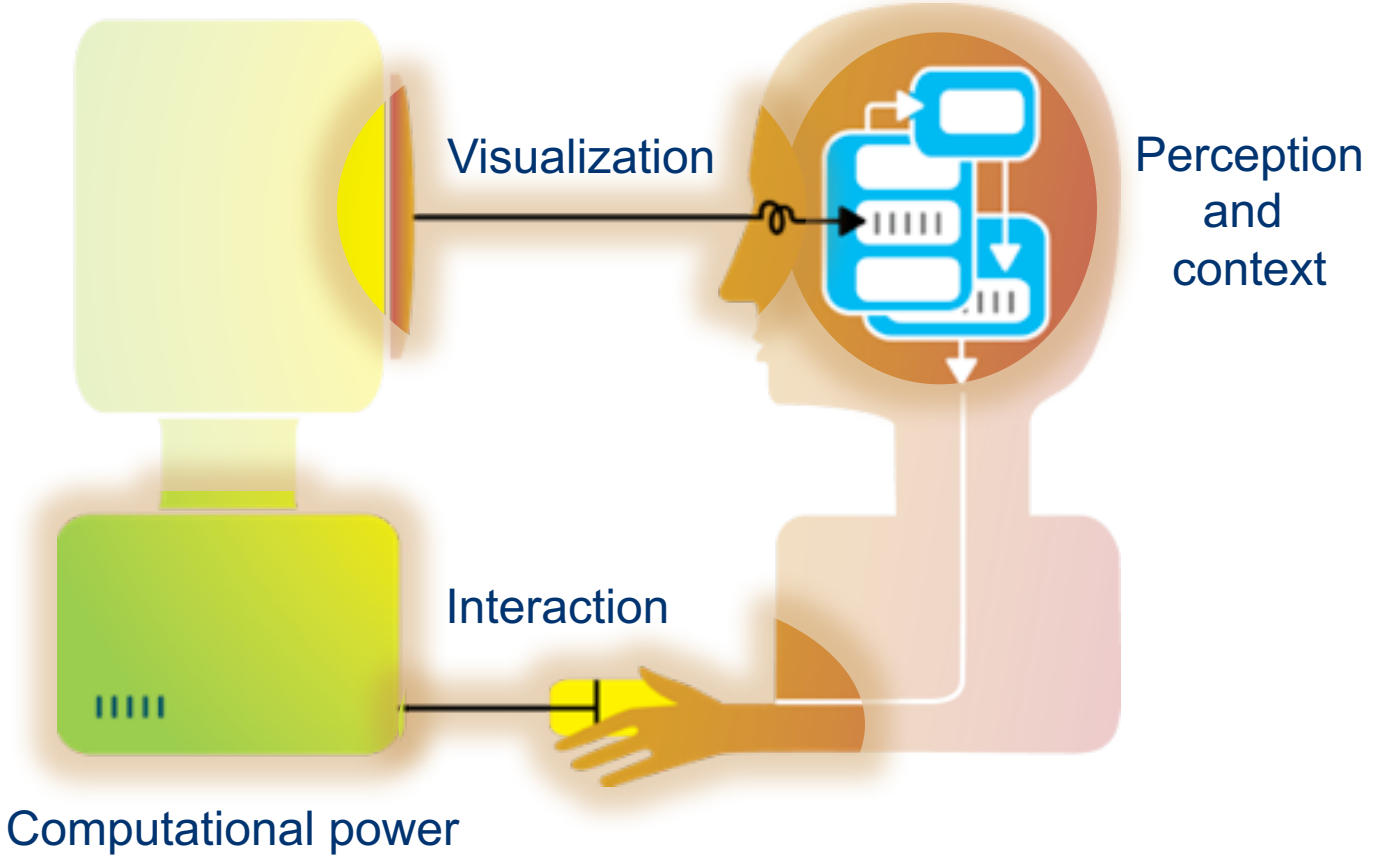
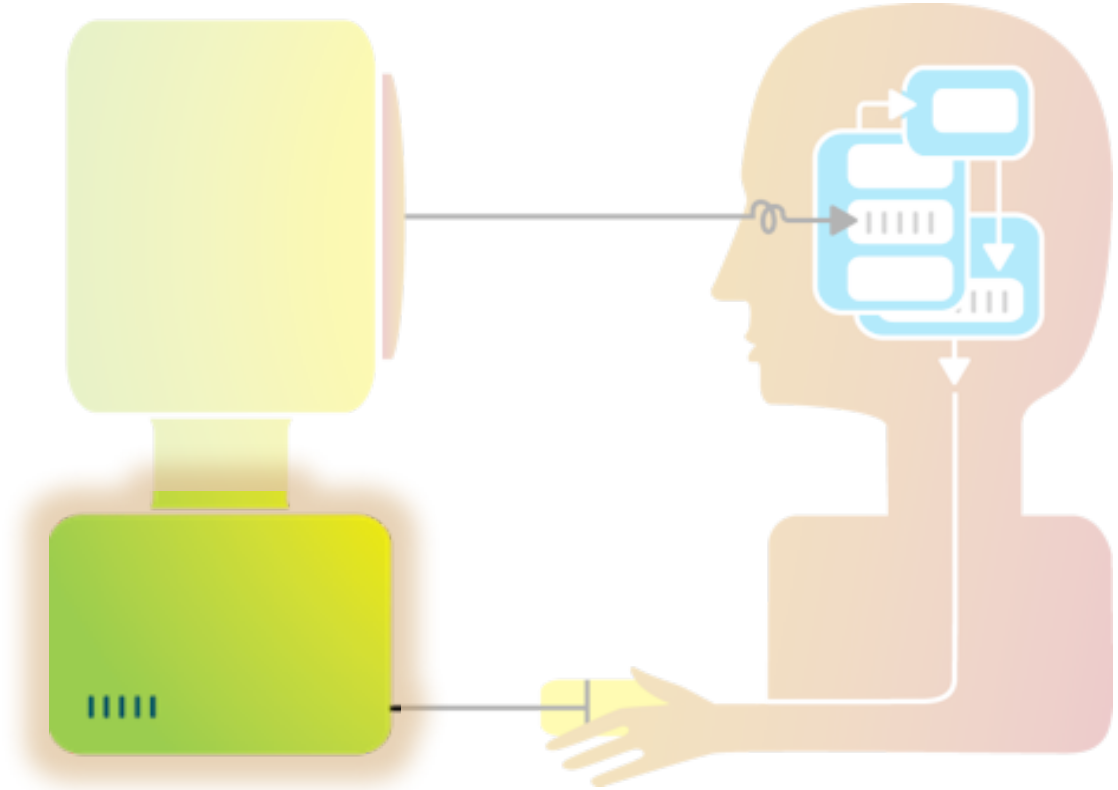


Image credit: Ali Ansari

This course

How does this part **work**,

and how to we get it to **do what we want**?



Computational power

Image credit: Ali Ansari

3-minute biographies

About **you**:

- Your name and pronouns
- Your class year
- Your intended major / area of focus (if you know)

3 questions:

1. What brought you **this course**?
2. What's one **big thing** you hope to get out of it?
3. What's one **big question** that sometimes keeps you up at night?

Goals of this course

Learn basic **programming constructs** and **internal workings** of a computer

Explore **foundational techniques** using **Python**

Opportunities to **work together** on projects you find **interesting**

Rough structure of the course

- **Monday and Wednesday lectures**:
  - Introduce a new idea
  - Mini-lab / activity
  - Brief discussion
  - Set up the assignment for the week
- **Wednesday and Thursday labs**
  - Hands-on practice with ideas from lecture
- **Fridays**: learn/practice a CS “Life Skill”
- **End of the semester**: group project (more on this later)

Setting expectations

- Bring a functional* laptop to class every day
  * this means remembering your **charger**

- **Do** use it for:
  - note-taking
  - coding
  - class website

- **Don't** use it for:
  - social media
  - shopping
  - other stuff

Outline

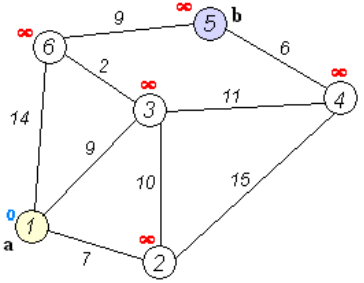
- ✓Introductions
- ✓Goals / structure
- Big questions for this course
- **Activity**: computational thinking
- Resources and details

Quick question…

#protip

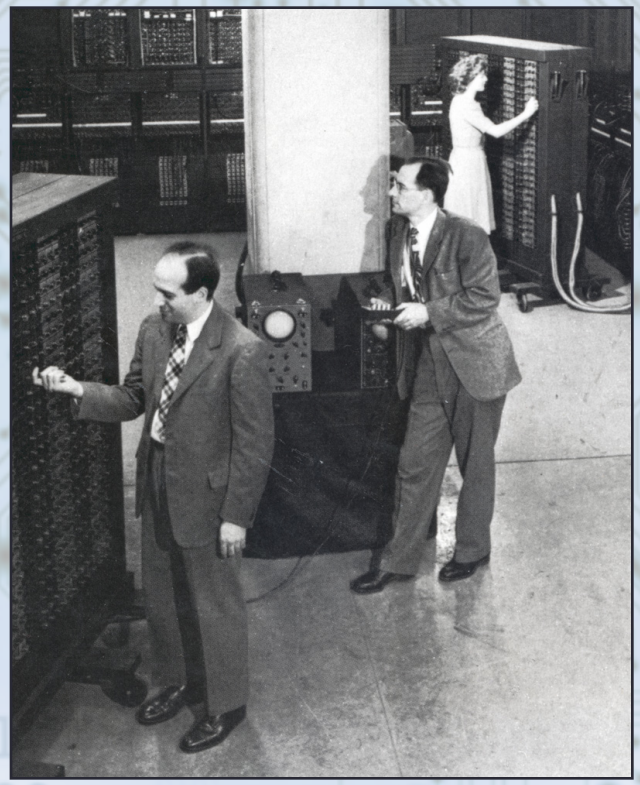
in classes with Jordan, this icon means "your turn to talk"

Computer science ≠ “computers”



Edsger
Dijkstra

1950s: “Electronic Brains”



www.computerhistory.org

2018: “Appliances”

- Essential tools in modern society
- Personal computers (desktops and laptops)
  - write papers, manage personal finances, ...
  - entertainment: games, video, audio, ...
- Business computers
  - day-to-day operations: payroll, billing, ...
  - customer service: web sites, customer data, ...
- Embedded computers
  - microchips used as controllers in cars, phones, wearables, ...
- Supercomputers
  - large “number crunchers” used in scientific research and other areas
  - parallel processing: from a few dozen to a few thousand CPU chips

Things computers **can't** do

Things computers **can't** do

Things computers **can't** do

Things computers **can't** do

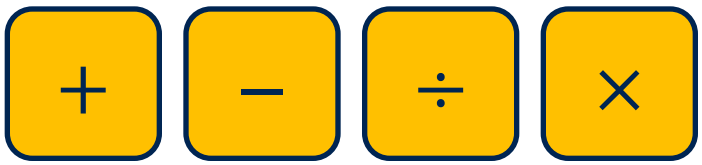
```
while (True):
    print("looping...")
```

Discussion: computational limits

What do you notice?

Computation (def.)

"a sequence of **well-defined operations**
that lead from an initial starting point
to a desired final outcome"

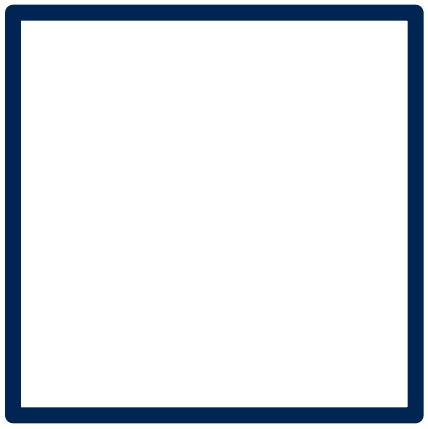


mathematical

logical

Activity: computational thinking

Activity: computational thinking



Task 1

Task 2

the 1st letter of your first name

Task 3

Discussion

- What do you notice?
- Were there any letters you couldn’t draw?
- Can you tell in advance which shapes are impossible?

Computer science (def.)

"the **study of computation**"

- **Problems** that can be solved computationally
- **Languages** used to describe computational processes
- **Machines** that carry out those processes
- **Theoretical limits** of computation
- **Computational solutions** to problems in math, science, medicine, business, education, journalism, ...

Programming language for this course

Quick cool thing: stuff written in Python

reddit

Instagram

Dropbox

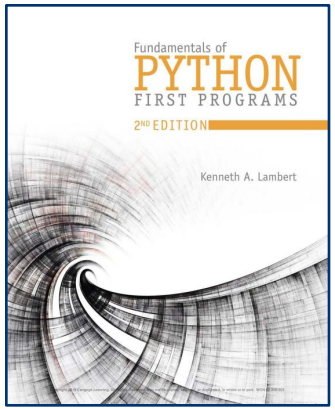
Goals for next week

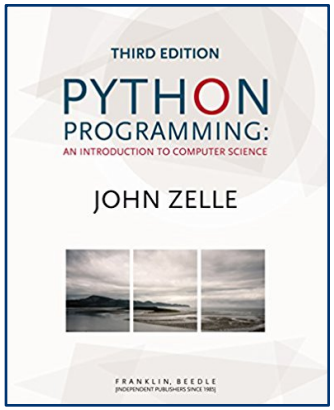
- Learn the basics **of how a computer works**
- Get **python** installed on your laptop
- Write simple programs that use **variables**, **loops**, read in **user input** and **output information**
- Learn a valuable life skill: **pair programming**

About this course: (no) textbooks

- There is **no required textbook** for this course
- There are two recommended references:

free





- And tons of **online resources**:

Python Academy

codecademy

About this course: major assessments

- Weekly assignments (out on Monday, due Sunday)
- There is one **take home midterm** in this course
- One final group project with an in-class demonstration
  - **Big idea**: apply concepts we've learned to a domain of your choice
  - More on this after fall break

About this course: grade breakdown

Component	Weight
Homework	35%
Labs	15%
Midterm	20%
Final Project	20%
Participation	10%
TOTAL	100%

About this course: late policy

Due to the number of students enrolled in this course, as well as the pace of the material, **no extensions will be given, and no late assignments will be accepted** without a written request from the student's Dean in the case of extenuating circumstances.

None.

Seriously.

I mean it.

However…

About this course: late policy

Due to the number of students enrolled in this course, as well as the pace of the material, **no extensions will be given, and no late assignments will be accepted** without a written request from the student's Dean in the case of extenuating circumstances.

Each student is permitted to drop one homework assignment without penalty, as well as one lab. If a student turns in all the assignments and labs, the one with the lowest grade will be dropped automatically.

Resources: where to get in-person help

- **Jordan's office hours** (Ford 355):

  Tuesdays 1pm to 3pm

  Fridays 9:30am to 11am

- **TA Hours** (Ford 241):

  Weeknights (Sun – Thurs) 7:30pm to 9:30pm

  Sundays 1:30pm to 3:30pm

Resources: course website & syllabus

jcrouser.github.io/CSC111

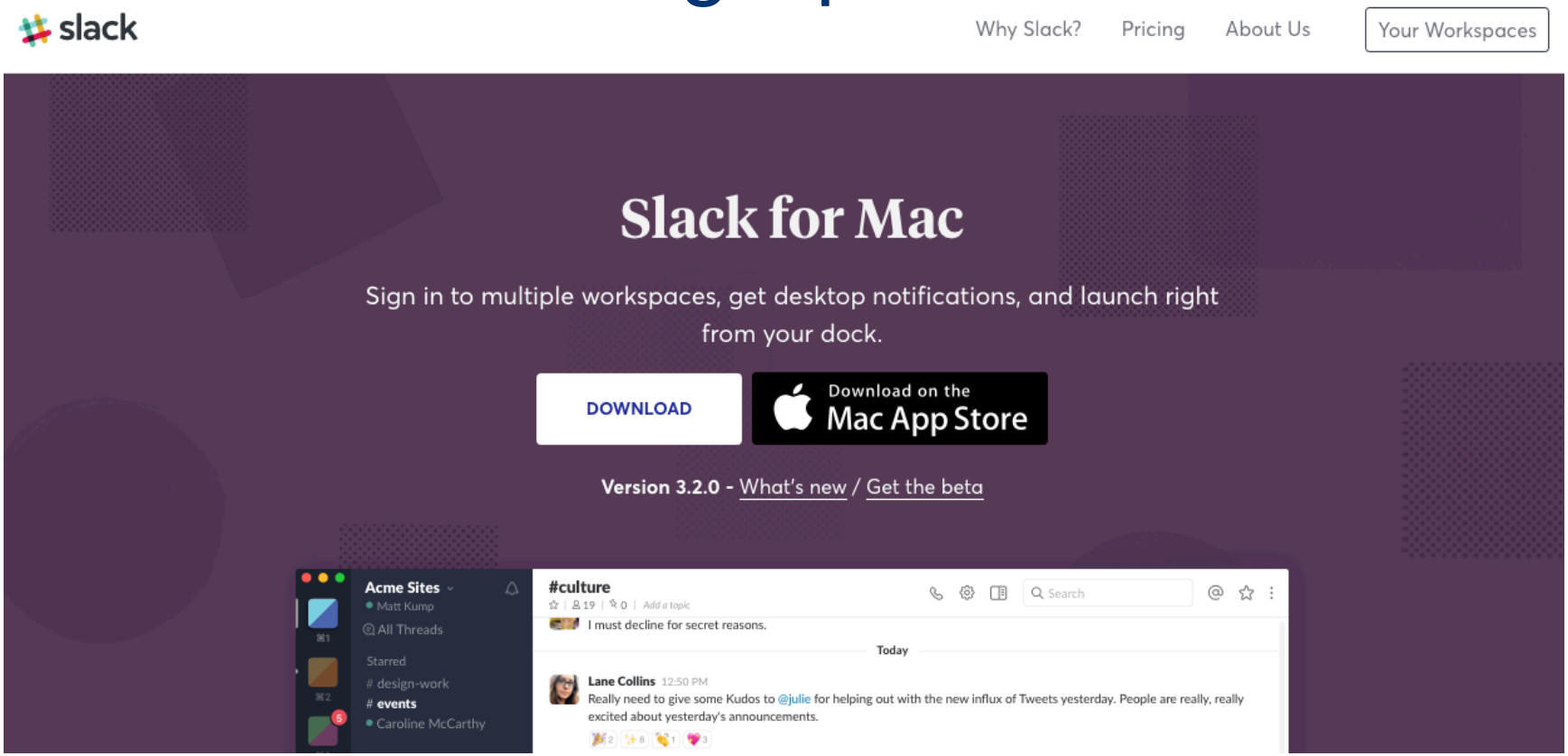
Resources: slack (all communication)

smith-csc111-f2018.slack.com

Resources: moodle (lecture notes, hw)

moodle.smith.edu

Homework for tonight pt. 1



Log into slack, set up desktop / phone app

slack.com/downloads

Homework for tonight pt. 2

Install Python 3 on your laptop

python.org/downloads

A note about enrollment

- As of 9AM:
  - 88 of 96 seats filled
  - 10 of you are **officially waitlisted**
  - others came to class today with a waiver hoping to get in

- Here's the deal:
  - If you are waitlisted, you have an email from me regarding lab
  - Reply by the **end of the day today**
  - I will triage the **official** waitlist over the weekend
  - **Everyone else**: I will take your form, and I will let you know Sunday evening if I am able to offer you a spot