

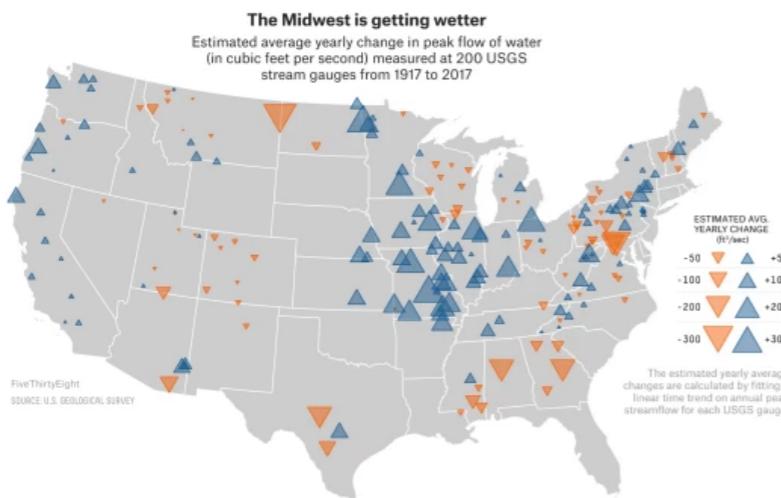
# [301] Introduction

Tyler Caraza-Harter

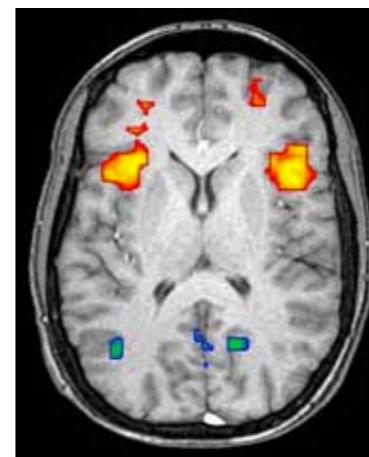
# Welcome to Data Programming!

Data is exploding in many fields

- Journalism
- Biology, physics, chemistry
- Psychology, sociology, economics, business
- Engineering (mechanical, electrical, industrial, etc)



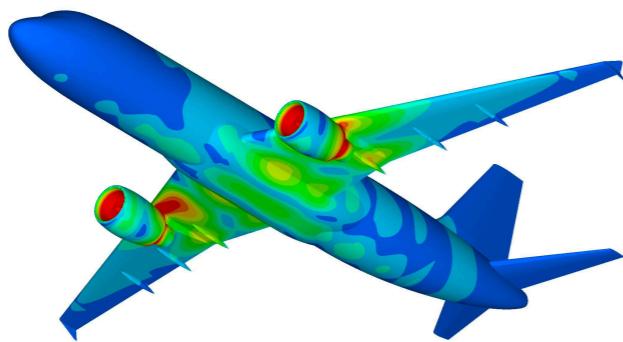
<https://fivethirtyeight.com/features/the-midwest-is-getting-drenched-and-its-causing-big-problems/>



<https://en.wikipedia.org/wiki/Neuroimaging>



<https://home.cern/topics/large-hadron-collider>



<http://www.stressebook.com/finite-element-analysis-in-a-nut-shell/>



<https://science.howstuffworks.com/life/genetic/gattaca-gaptacaz-adding-letters-the-genetic-alphabet.htm>

# Welcome to Data Programming!

Data is exploding in many fields

- Journalism
- Biology, physics, chemistry
- Psychology, sociology, economics, business
- Engineering (mechanical, electrical, industrial, etc)

How can we gain insights from that data?

- With computation

# Welcome to Data Programming!

Data is exploding in many fields

- Journalism
- Biology, physics, chemistry
- Psychology, sociology, economics, business
- Engineering (mechanical, electrical, industrial, etc)

How can we gain insights from that data?

- With computation

**Approach 1: human computation**



# Welcome to Data Programming!

Data is exploding in many fields

- Journalism
- Biology, physics, chemistry
- Psychology, sociology, economics, business
- Engineering (mechanical, electrical, industrial, etc)

How can we gain insights from that data?

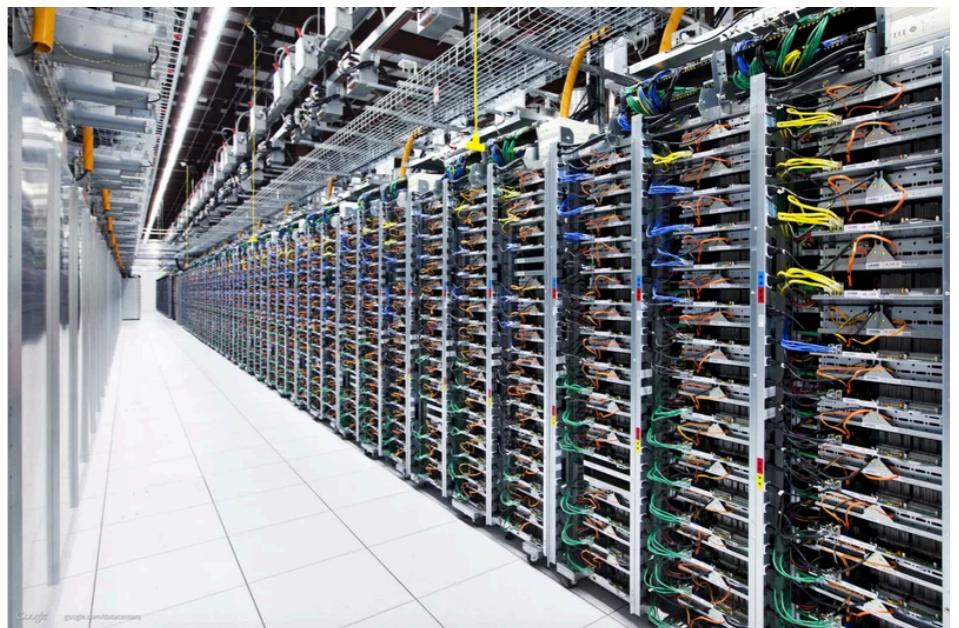
- With computation

**Approach 1: human computation**



[https://en.wikipedia.org/wiki/Human\\_computer](https://en.wikipedia.org/wiki/Human_computer)

**Approach 2: machine computation**



<http://fortune.com/2015/11/15/intel-super-7/>

# Welcome to Data Programming!

CS 301 is about approach 2

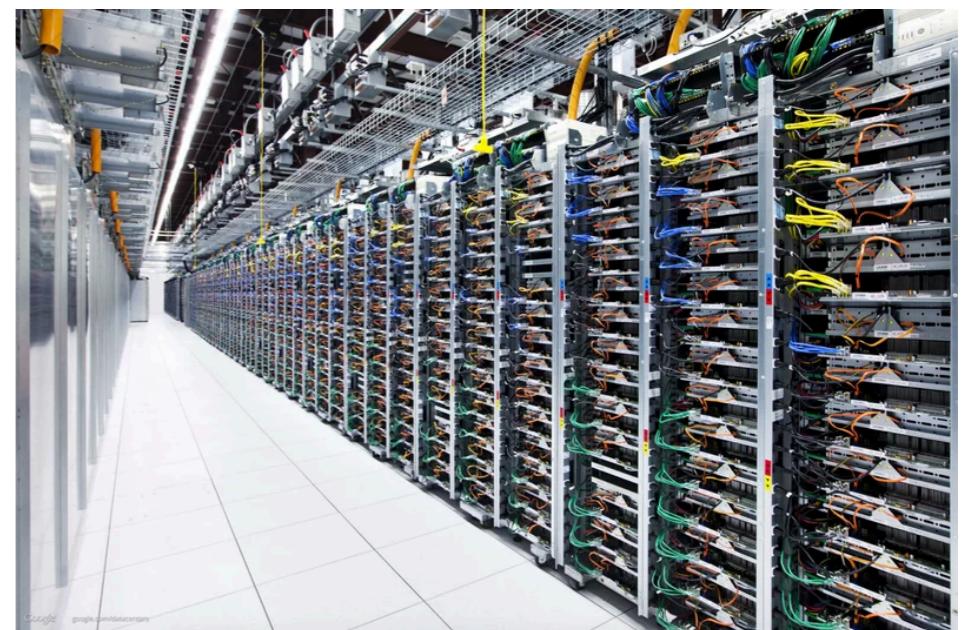
- Faster, more reliable, can churn through more data

**Approach 1: human computation**



[https://en.wikipedia.org/wiki/Human\\_computer](https://en.wikipedia.org/wiki/Human_computer)

**Approach 2: machine computation**



<http://fortune.com/2015/11/15/intel-super-7/>

# Welcome to Data Programming!

CS 301 is about approach 2

- Faster, more reliable, can churn through more data
- Automate to save human effort

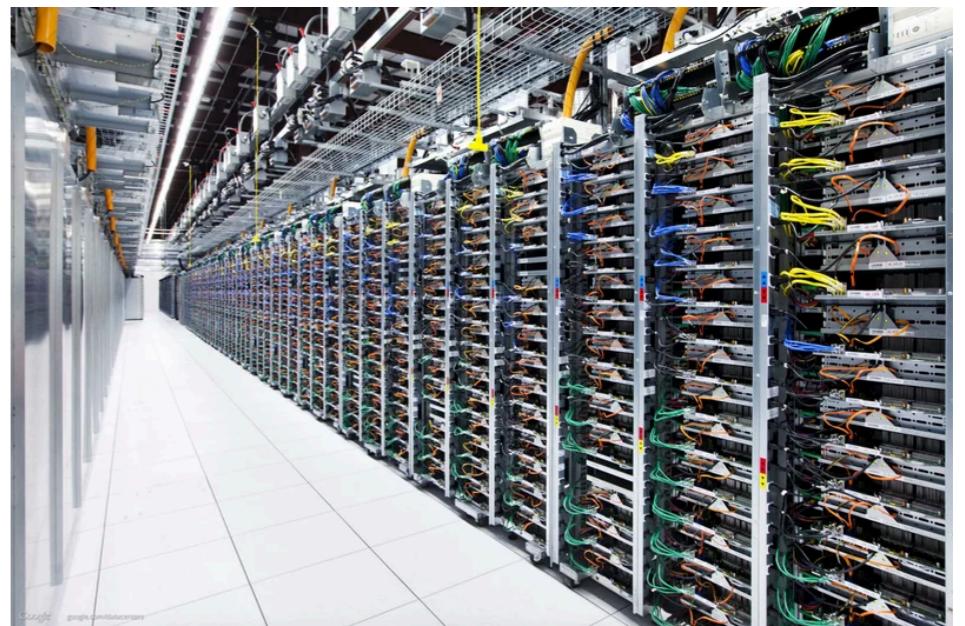
*“Find the leverage in the world, so you can **be more lazy!**”*

~ Larry Page

Approach 1: human computation



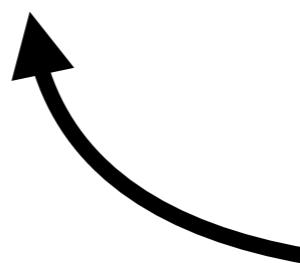
Approach 2: machine computation



# Welcome to Data Programming!

CS 301 is about approach 2

- Faster, more reliable, can churn through more data
- Automate to save human effort
- Requires being able to tell computers what to do!

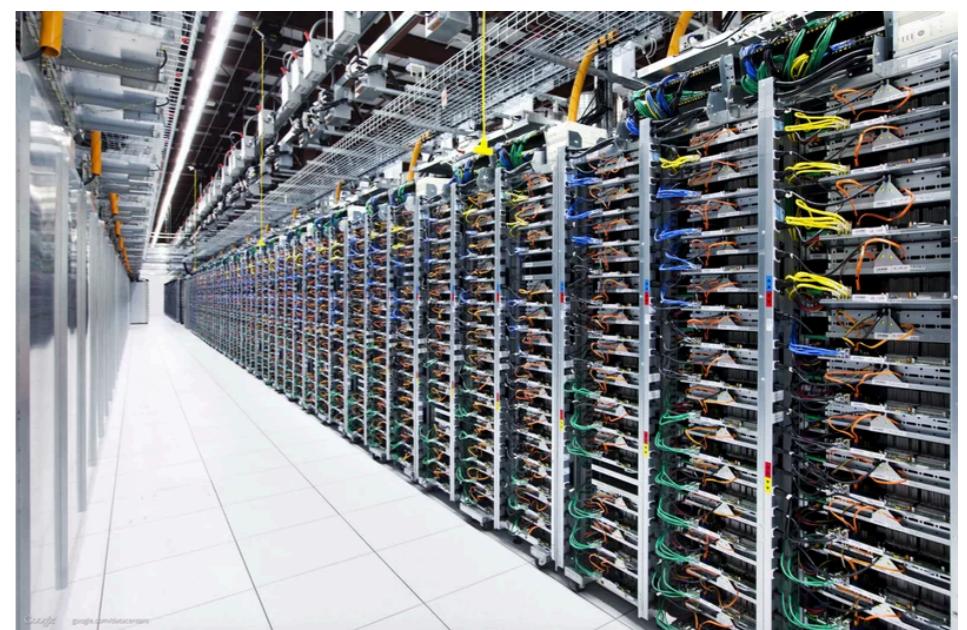


society needs more **domain experts**  
in specific fields **who can write code**

Approach 1: human computation



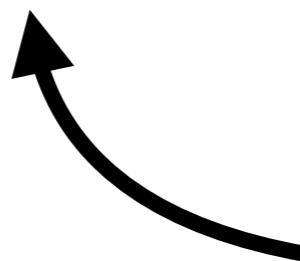
Approach 2: machine computation



# Welcome to Data Programming!

CS 301 is about approach 2

- Faster, more reliable, can churn through more data
- Automate to save human effort
- Requires being able to tell computers what to do!



society needs more **domain experts** in specific fields **who can write code**

Goal: become "bilingual"

- Speak the language of **X** (biology, mech eng, journalism, etc)
- Speak the language of **computing**

# Why CS 301?

Common approach to introductory CS courses

- Use a programming language like C++ or Java
- Teach CS students and other majors together
- Emphasis on writing large programs (OOP, encapsulation) and theory (complexity analysis)
- Light on data

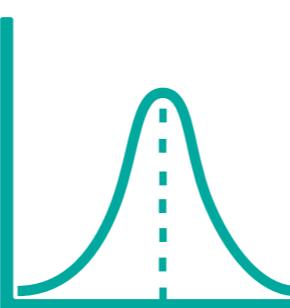
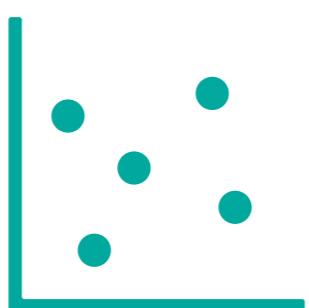
# Why CS 301?

Common approach to introductory CS courses

- Use a programming language like C++ or Java
- Teach CS students and other majors together
- Emphasis on writing large programs (OOP, encapsulation) and theory (complexity analysis)
- Light on data

CS 301 approach

- Pioneered by Laura Hobbes LeGault
- Use **Python** (powerful but ~~easy~~ easier to learn)
- Goal: bring more programming into other fields
- Practical, minimal theory
- **Emphasis on data**, simulation, analysis, plotting



# Today's Topics

## Introductions

- Who am I? Who are you?

## Course overview

## Computer hardware basics

## Website

# Who am I?

Tyler Caraza-Harter

- Email: [tylerharter@gmail.com](mailto:tylerharter@gmail.com)
- Just call me “Tyler”



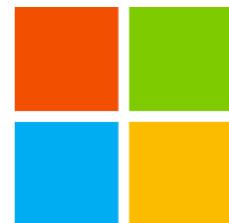
Long time badger

- Did undergrad, masters, and PhD at UW-Madison
- 2nd time teaching CS 301



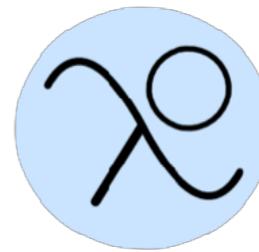
Work in industry

- Worked at Microsoft on SQL Server and Cloud
- Other internships/collaborations:  
Qualcomm, Google, Facebook, Tintri



Open-source projects

- OpenLambda project (Python-based platform)
- PivotLibre project (preferential-voting tool)



# Who are You?

Year in school?

- 1st year? 2nd? Junior/senior? Grad student?

Area of study

- Natural science, social science, engineering, other?

How many have programmed before?

- Any language? Python? Taken a class?

# Survey (counts for participation)

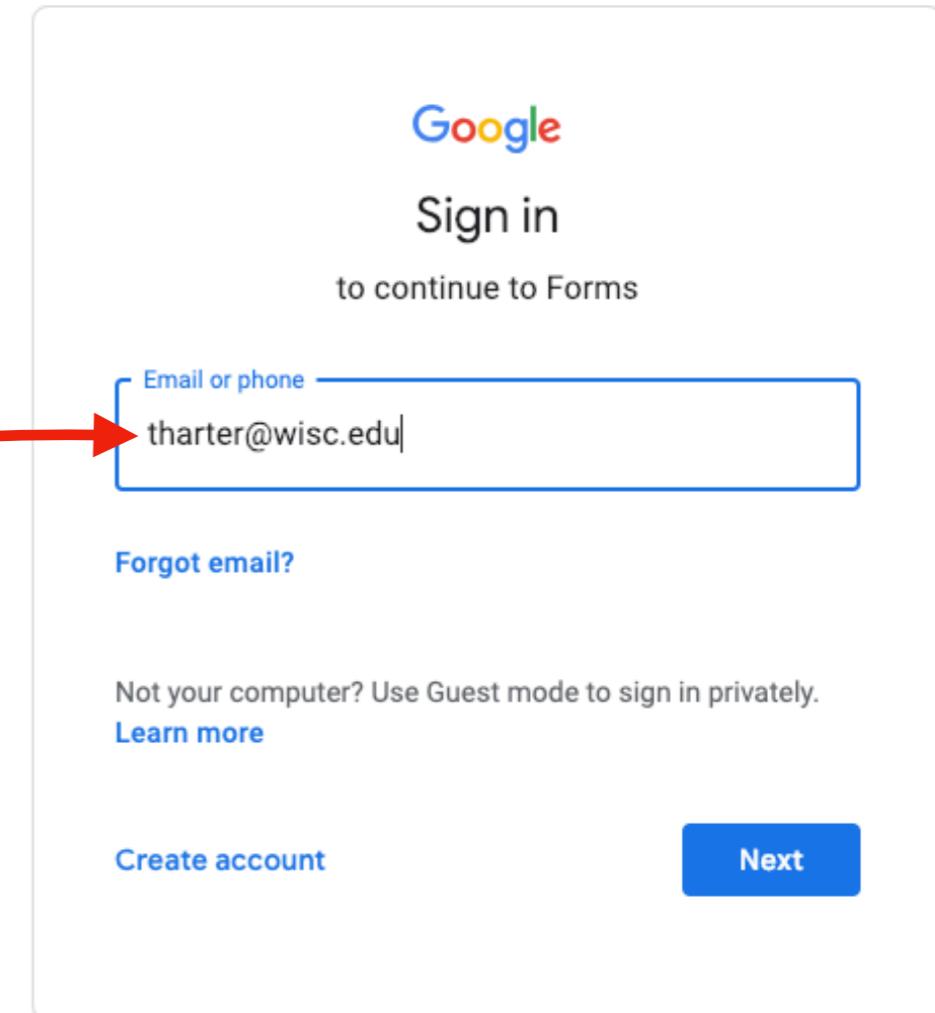
Please help us get to know you (not anonymous):

<https://goo.gl/forms/WtWRjr7qFt4jYfEV2>

Purposes:

- gauge class interest/experience
- determine who on waitlist is attending
- correlate experience with later scores

be sure to use your  
campus email!!!



# Survey: Common Technical Issues

You need permission

This form can only be viewed by users in the owner's organization.

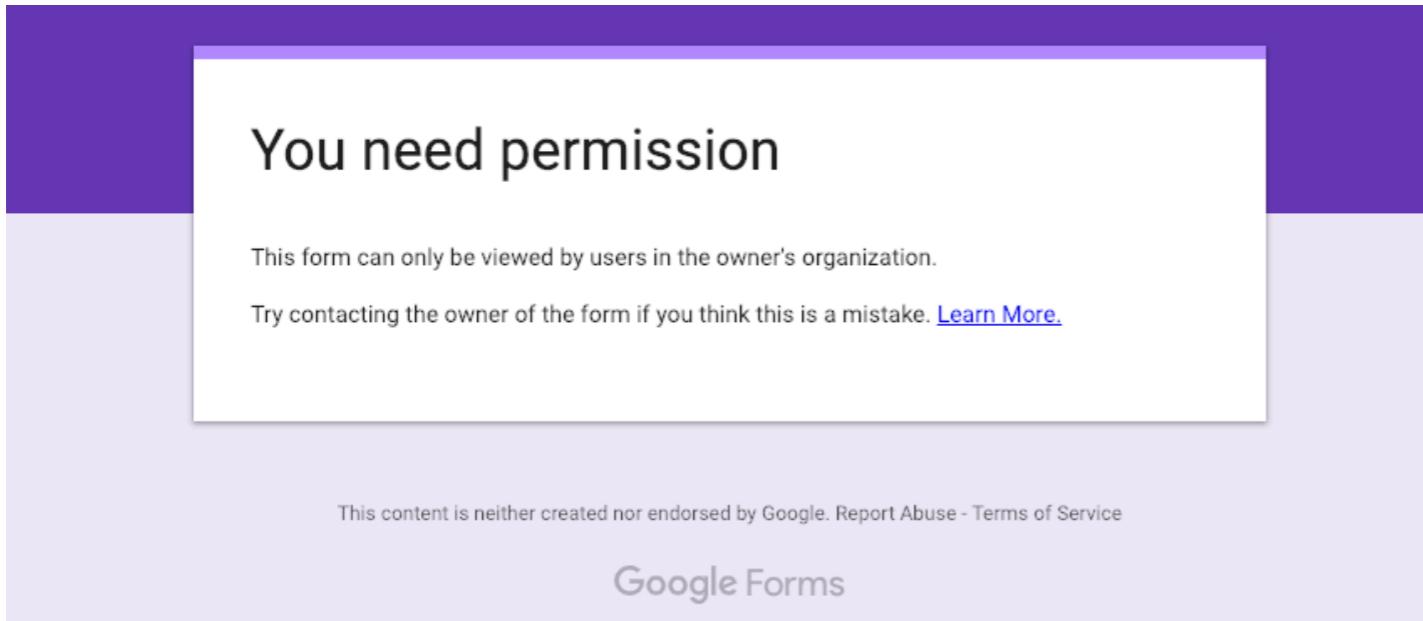
Try contacting the owner of the form if you think this is a mistake. [Learn More.](#)

This content is neither created nor endorsed by Google. Report Abuse - Terms of Service

Google Forms

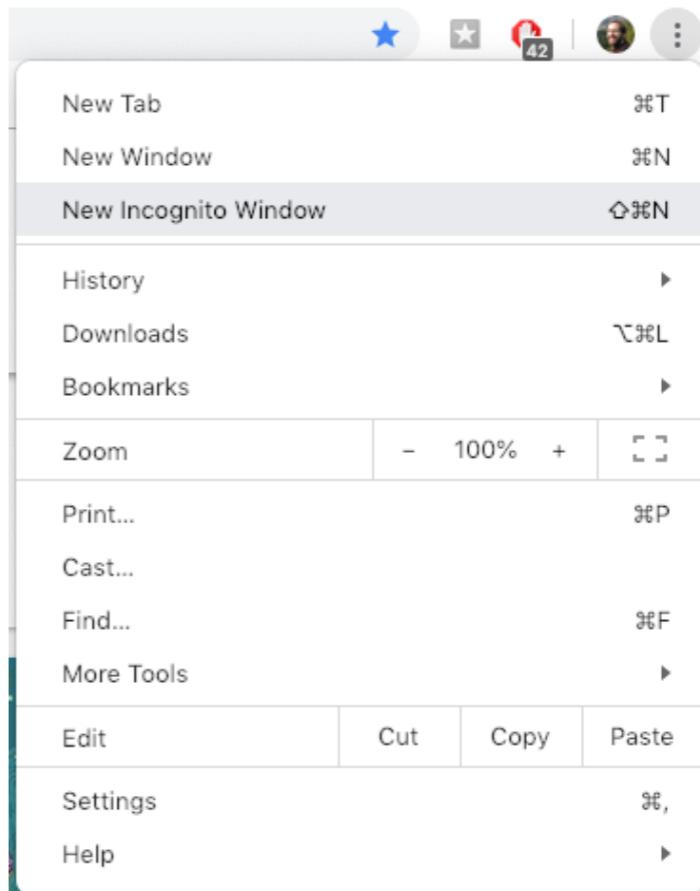
if you see this, it means you're signed in via Gmail instead of your campus email

# Survey: Common Technical Issues



if you were automatically signed into gmail without being asked, consider clearing cookies or using an Incognito Window (in Chrome)

if you see this, it means you're signed in via Gmail instead of your campus email



# Three types of students...

## 1. Lazy CS senior

*"I am a senior CS student, this class was very easy for me" [from F18 course eval]*



**Comment:** drop this course and take something that challenges you.  
You're not the target audience, and we'll make zero effort to make 301 interesting to you.

## 2. Mandatory participants

*"Make it significantly easier. None of [us] will ever code again..." [from F18 course eval]*



**Comment:** we're working with the assumption that your future careers WILL involve writing code in the future. 301 is challenging because our goal to prepare you for it.

## 3. Enthusiastic beginner

*"Good course, I think there is a good pace for this course, speaking as someone with zero programming experience coming into the class." [from F18 course eval]*



**Comment:** this course is for you! Invest the time, and you'll walk away with a superpower (the ability to make computers work for you)

# Today's Topics

Introductions

Course overview

Computer hardware basics

Website

# Today's Topics

Introductions

Course overview

- Learning objectives
- Lecture
- Lab
- Readings
- Class communication
- Grades
- Projects
- Exams

Computer hardware basics

Website

# 301 Topics

## Basic Python

- How to get data in/out of a program [I/O]
- How does program remember things [state]
- What is executed when [control Flow]

## Data structures

- How to organize lots of data? [data structures]

## Popular data formats

- How to ingest tabular data, hierarchical data, web pages

## Database basics

- Asking questions about data [querying]

## Plotting

- Choosing and implementing visual communication

# Today's Topics

Introductions

Course overview

- Learning objectives
- Lecture
- Lab
- Readings
- Class communication
- Grades
- Projects
- Exams

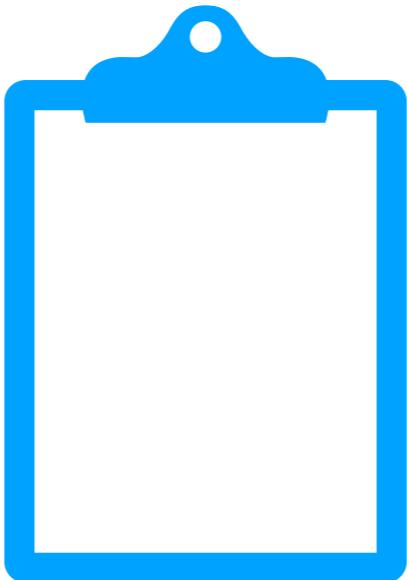
Computer hardware basics

Website

# Lecture Style



general concepts



worksheet practice



live coding

## Your role

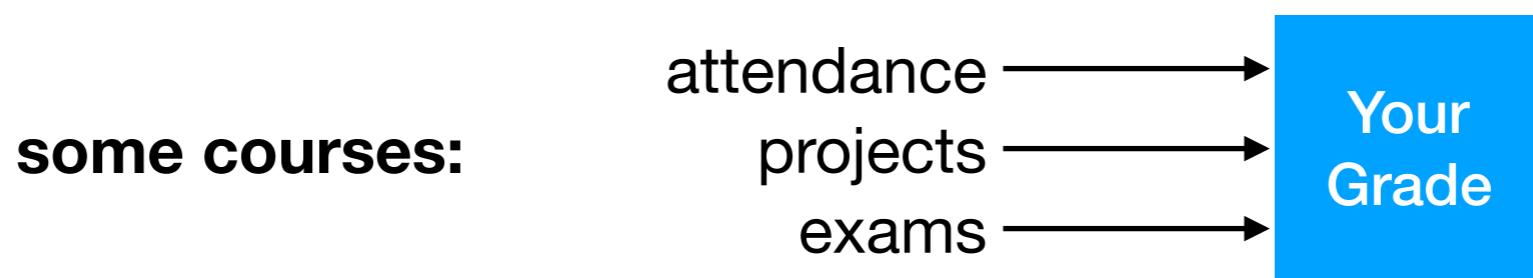
- do **readings** before or after
- feel free to bring your **laptop** along!  
(but avoid the temptation to get distracted)
- I love to get **questions**



# Lecture Attendance

Attendance isn't graded

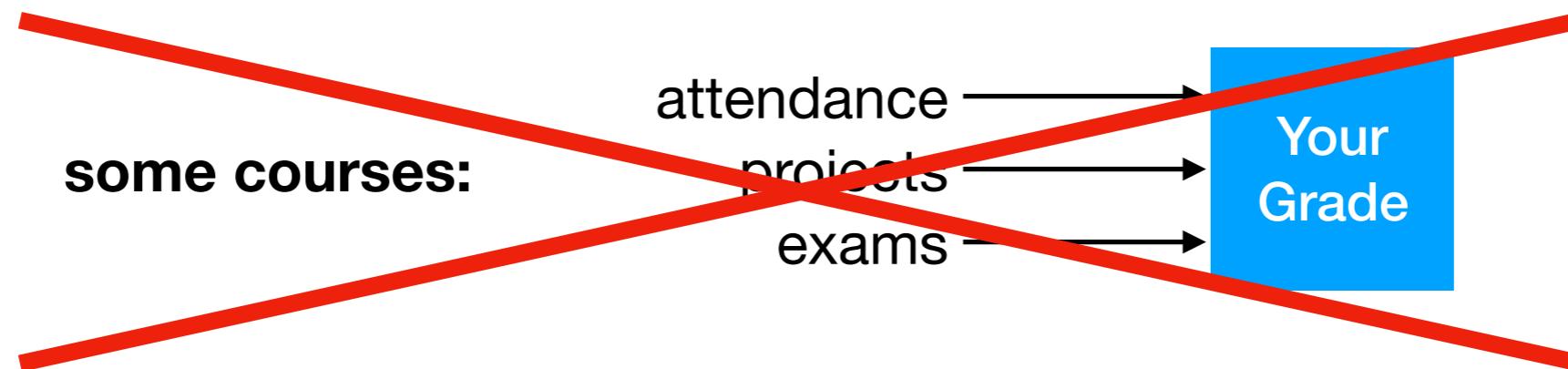
- One student's suggestion for improvement (f18):  
*"I think one MAJOR thing to do is make lecture mandatory"*



# Lecture Attendance

Attendance isn't graded

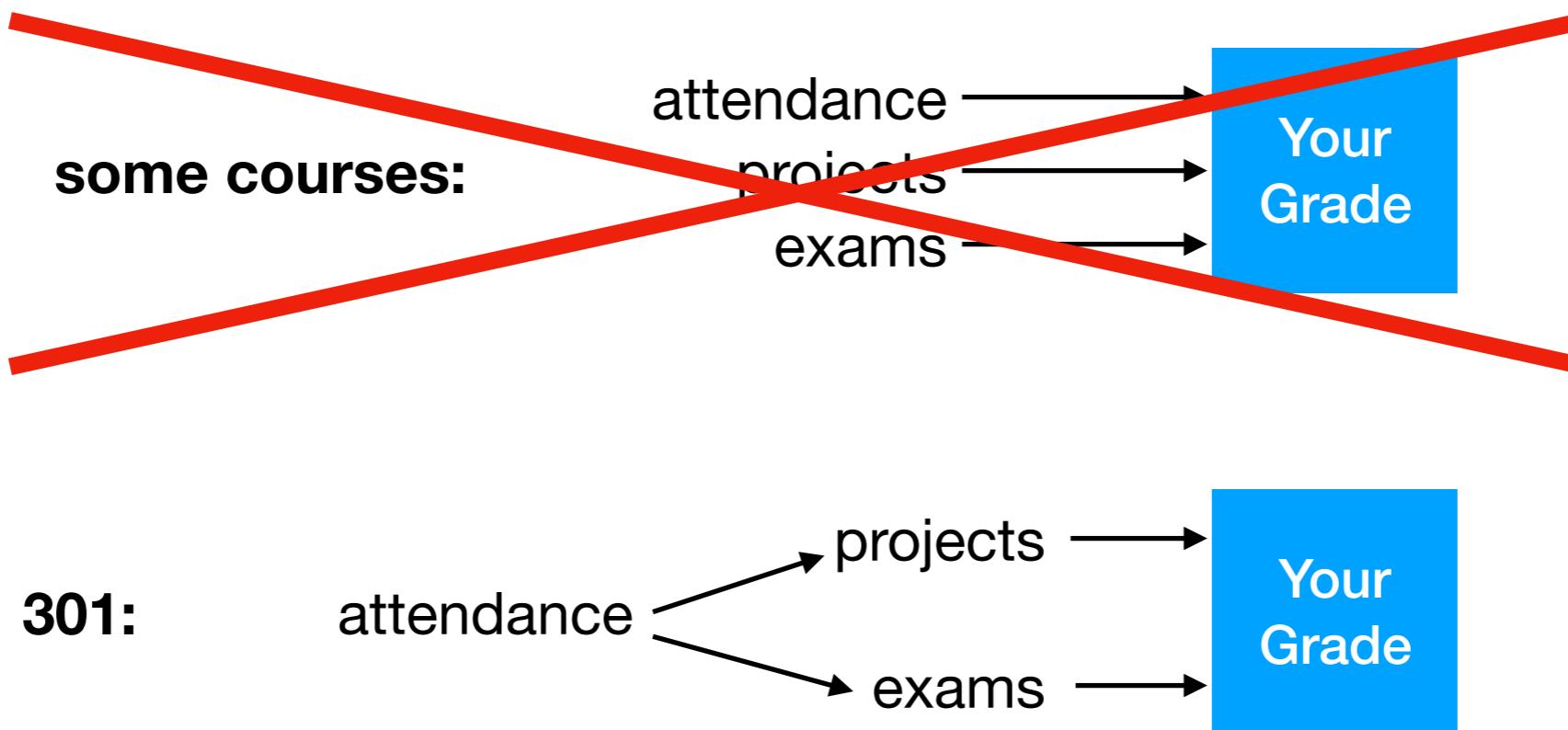
- One student's suggestion for improvement (f18):  
*"I think one MAJOR thing to do is make lecture mandatory"*



# Lecture Attendance

Attendance isn't graded

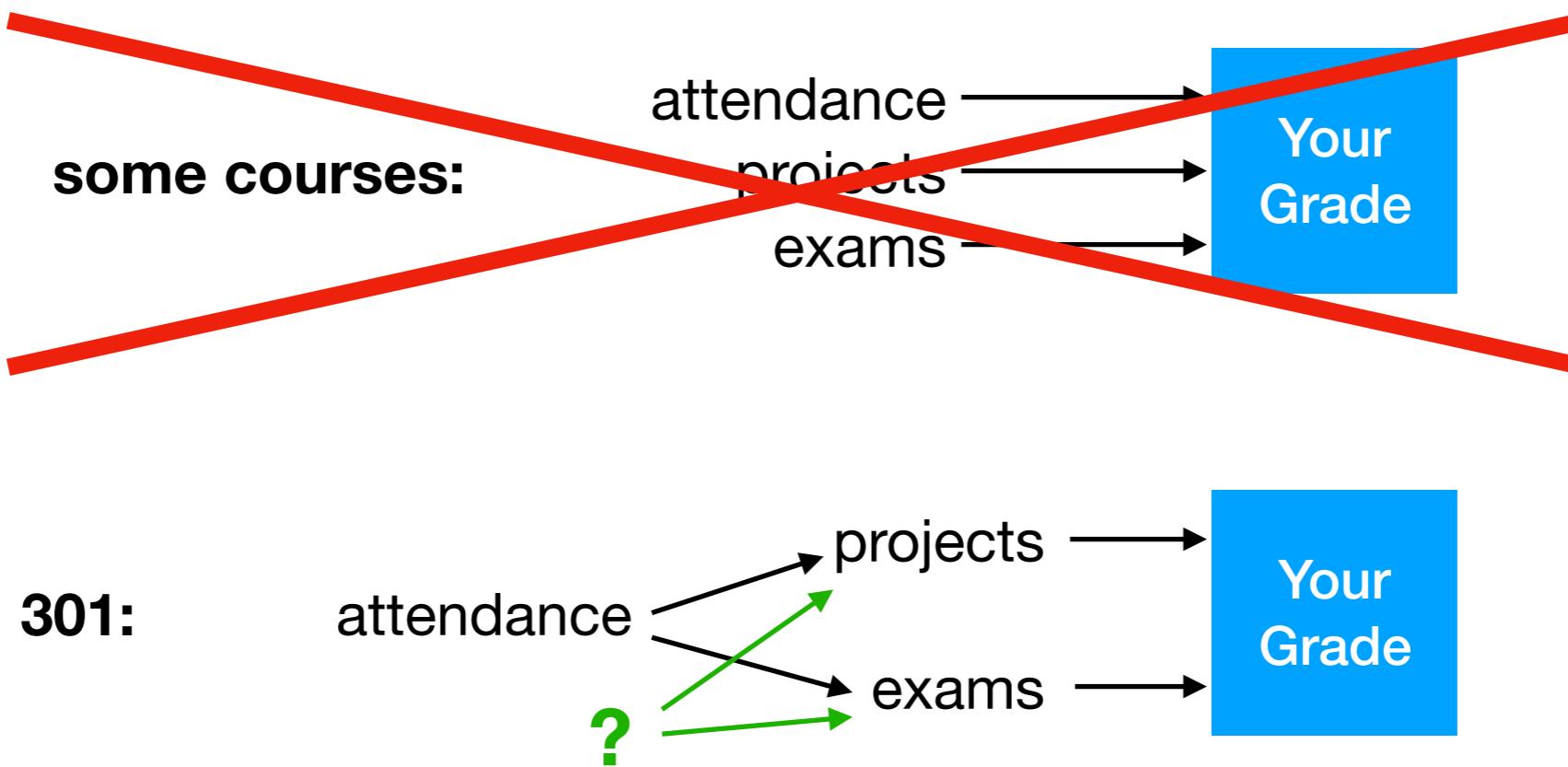
- One student's suggestion for improvement (f18):  
*"I think one MAJOR thing to do is make lecture mandatory"*



# Lecture Attendance

Attendance isn't graded

- One student's suggestion for improvement (f18):  
*"I think one MAJOR thing to do is make lecture mandatory"*

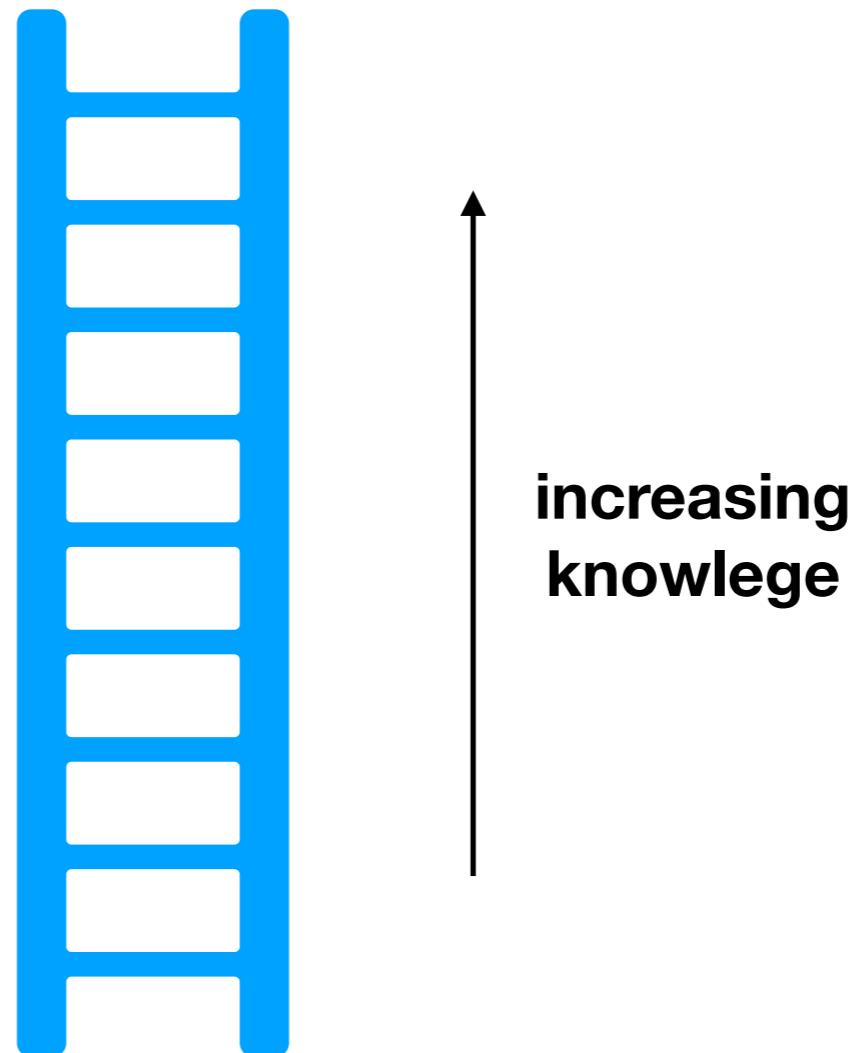


**Tip:** as a student, you can make anything mandatory for yourself you like!  
Think carefully about what you should, then practice following through.

# Topic Progression

Programming knowledge builds

- For first couple months, trying to skip foundational knowledge will be problematic throughout the semester



# Today's Topics

Introductions

Course overview

- Learning objectives
- Lecture
- Lab
- Readings
- Class communication
- Grades
- Projects
- Exams

Computer hardware basics

Website

# Labs

## Format

- 75 minutes on Thu or Fri, leave when you're done
- typically: work through a practice document
- self guided, not graded
- **purpose: prep for project**

## People

- pair with another student each time
- 1-2 TAs will be there to answer questions

## Computers

- lab computers are provided (must have 2 students per computer)
- feel free to bring your own laptop

**we will have labs this first week**

# Today's Topics

Introductions

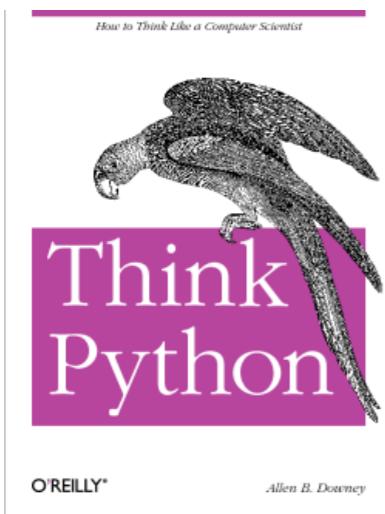
Course overview

- Learning objectives
- Lecture
- Lab
- **Readings**
- Class communication
- Grades
- Projects
- Exams

Computer hardware basics

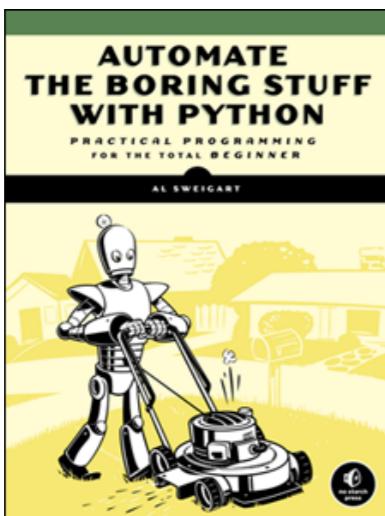
Website

# Readings (all free!)



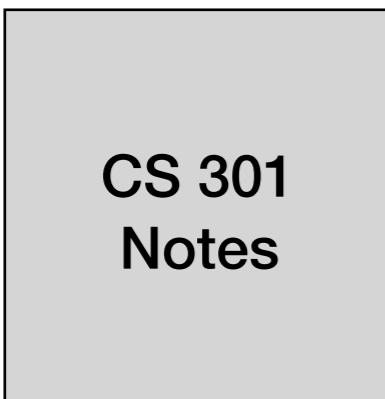
## Think Python, 2nd Edition

- Allen B. Downey
- Assumes no programming background
- It's very concise
- Get the 2nd edition, which is for **Python 3!**



## Automate the Boring Stuff

- Al Sweigart
- Useful for some more advanced topics related to using data



## Course Notes

- 301 instructors
- Mostly for data science part of class

# Today's Topics

Introductions

Course overview

- Learning objectives
- Lecture
- Lab
- Readings
- **Class communication**
- Grades
- Projects
- Exams

Computer hardware basics

Website

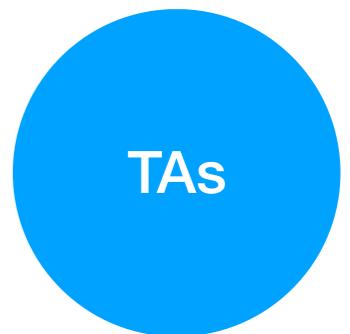
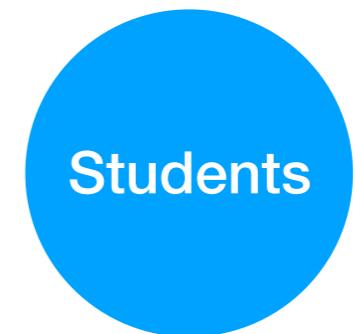
# Communication is CS 301

**Good communication is critical for a class of this size**

- Who needs to communicate: students, TAs, instructors

**Besides direct email, we'll use five communication tools**

- Piazza
- Email lists
- Feedback Form
- Project Submission
- Canvas



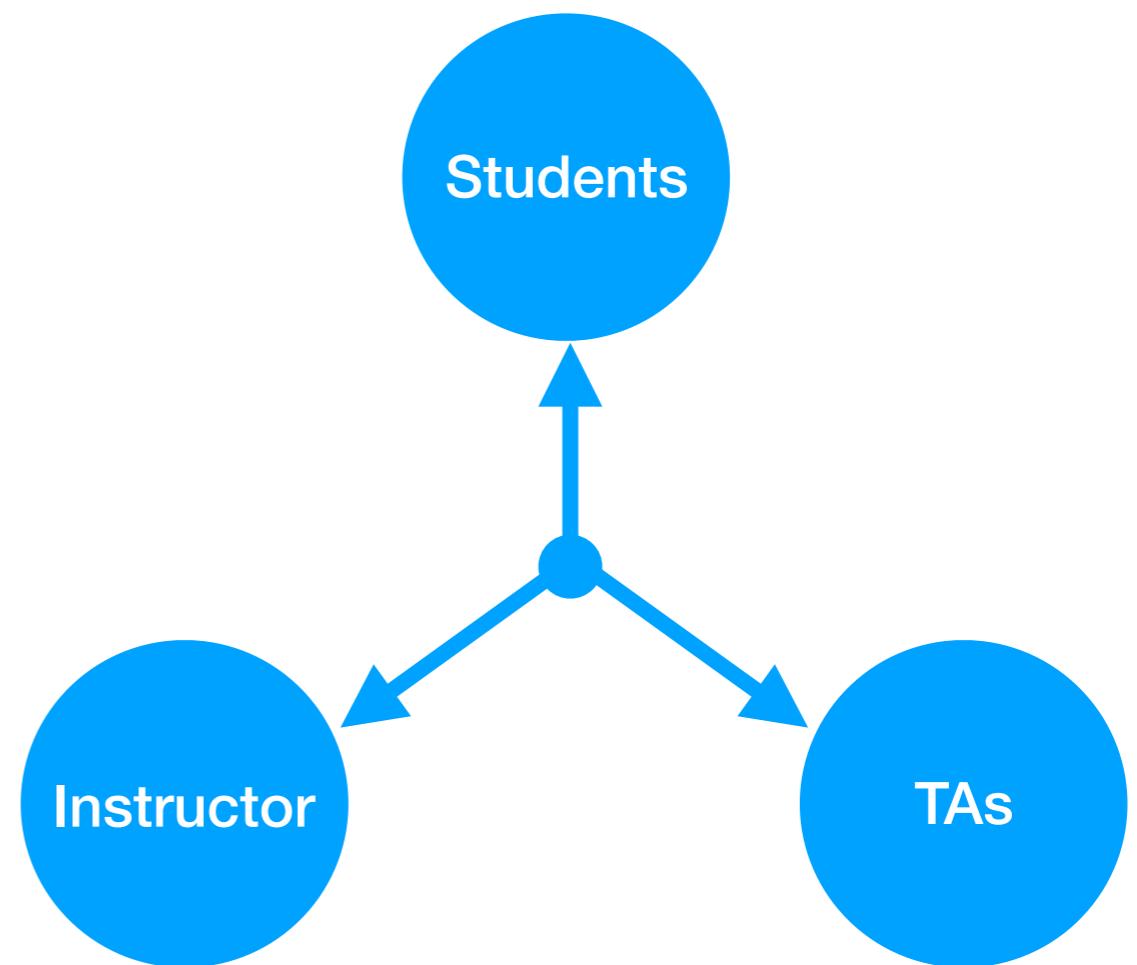
# Communication is CS 301

**Good communication is critical for a class of this size**

- Who needs to communicate: students, TAs, instructors

**Besides direct email, we'll use five communication tools**

- Piazza
- Email lists
- Feedback Form
- Project Submission
- Canvas



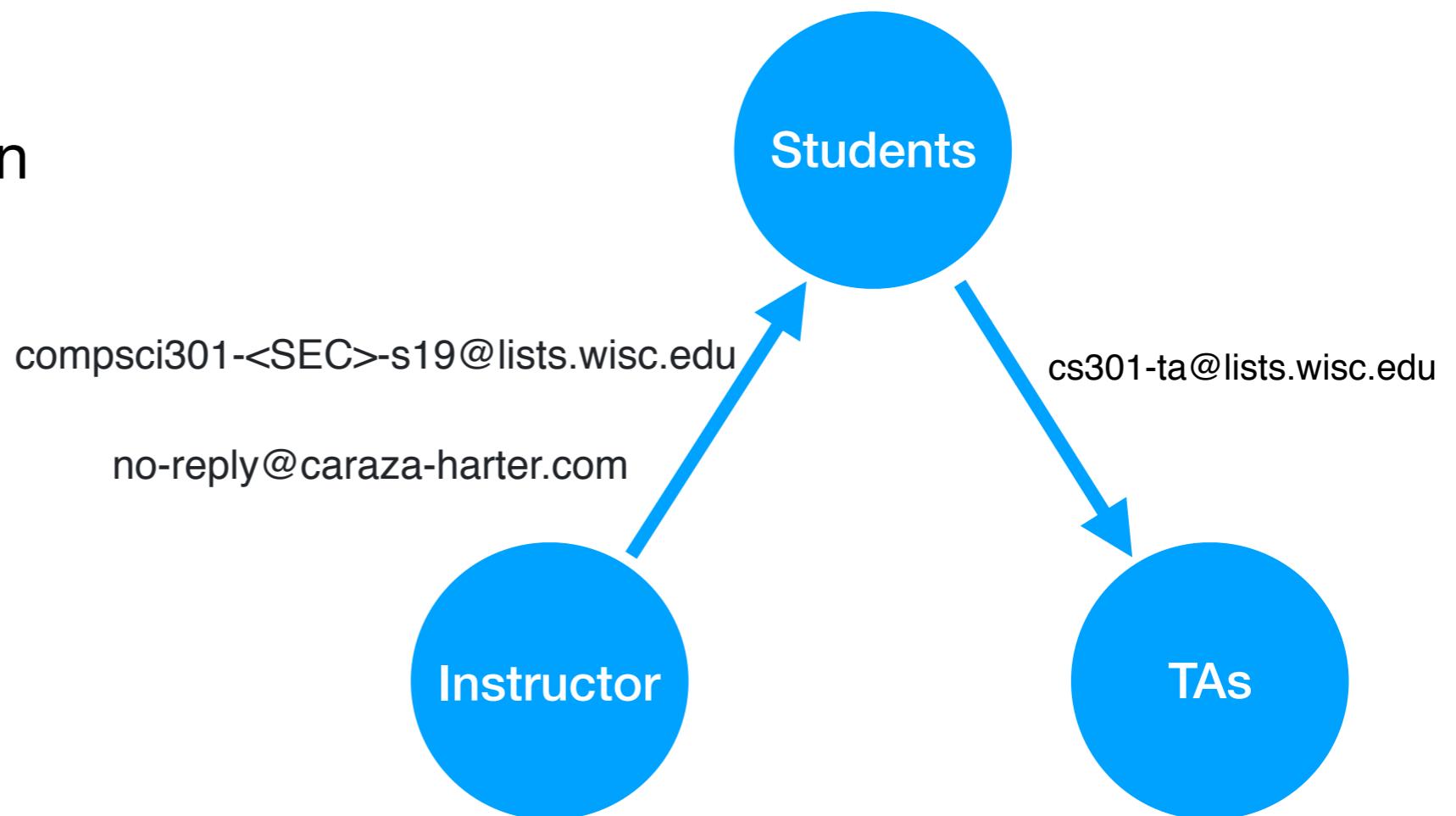
# Communication is CS 301

**Good communication is critical for a class of this size**

- Who needs to communicate: students, TAs, instructors

**Besides direct email, we'll use five communication tools**

- Piazza
- Email lists
- Feedback Form
- Project Submission
- Canvas



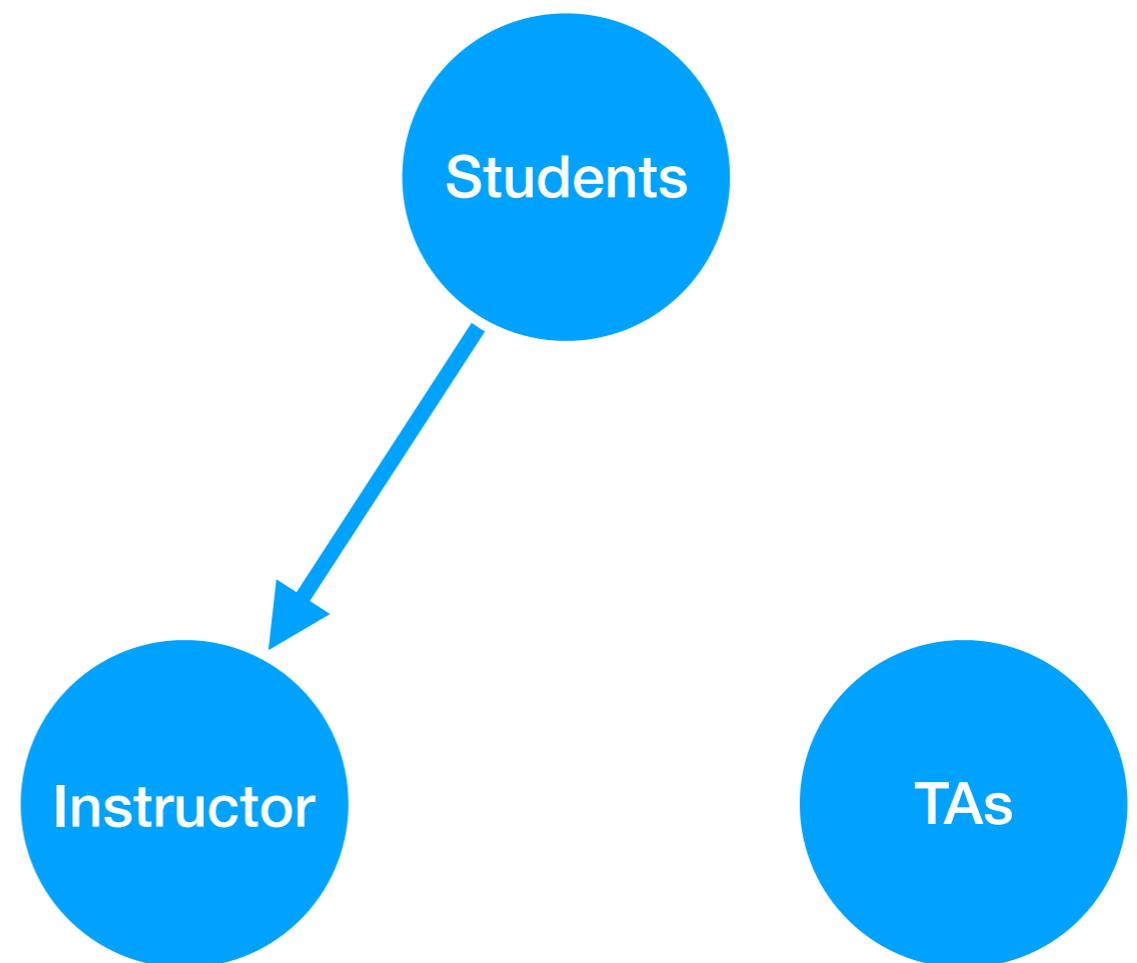
# Communication is CS 301

**Good communication is critical for a class of this size**

- Who needs to communicate: students, TAs, instructors

**Besides direct email, we'll use five communication tools**

- Piazza
- Email lists
- **Feedback Form**
- Project Submission
- Canvas



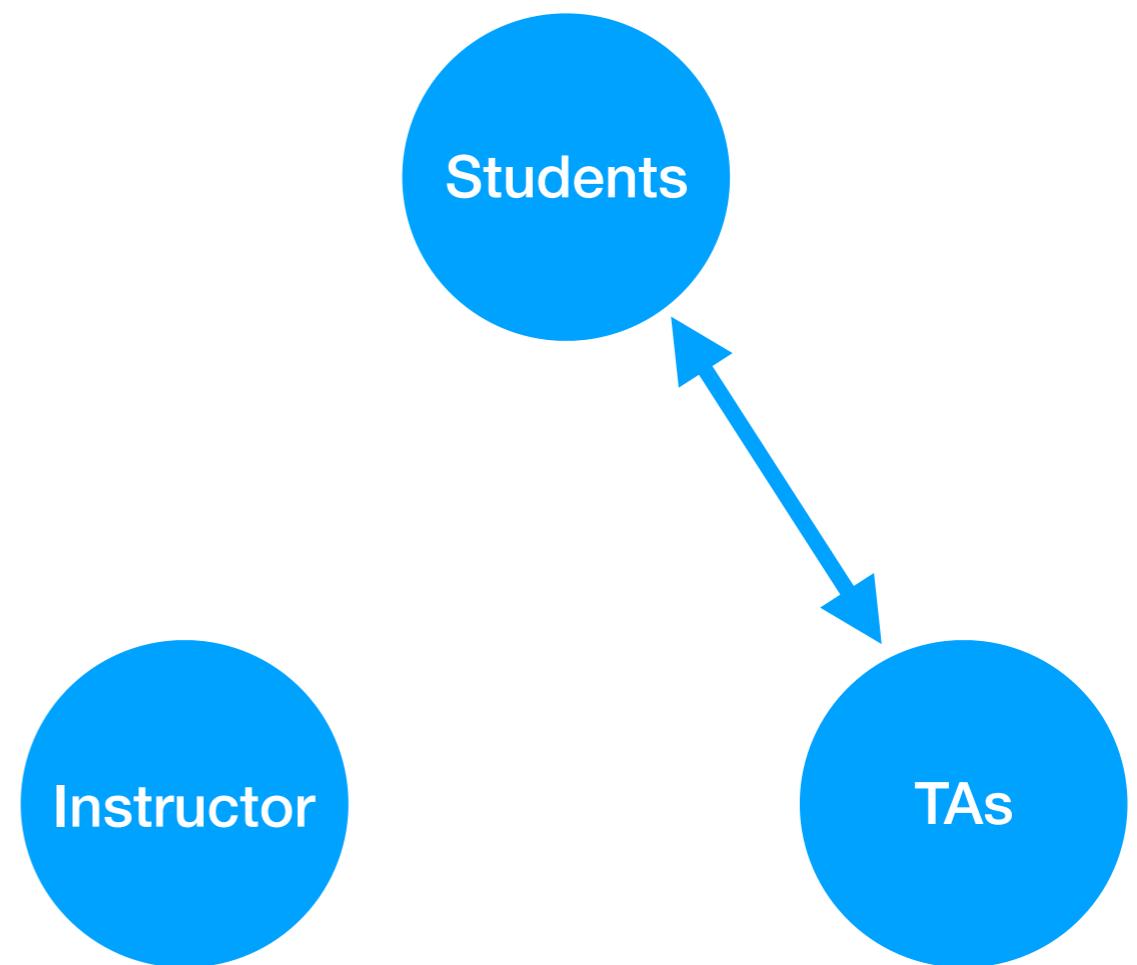
# Communication is CS 301

**Good communication is critical for a class of this size**

- Who needs to communicate: students, TAs, instructors

**Besides direct email, we'll use five communication tools**

- Piazza
- Email lists
- Feedback Form
- Project Submission
- Canvas



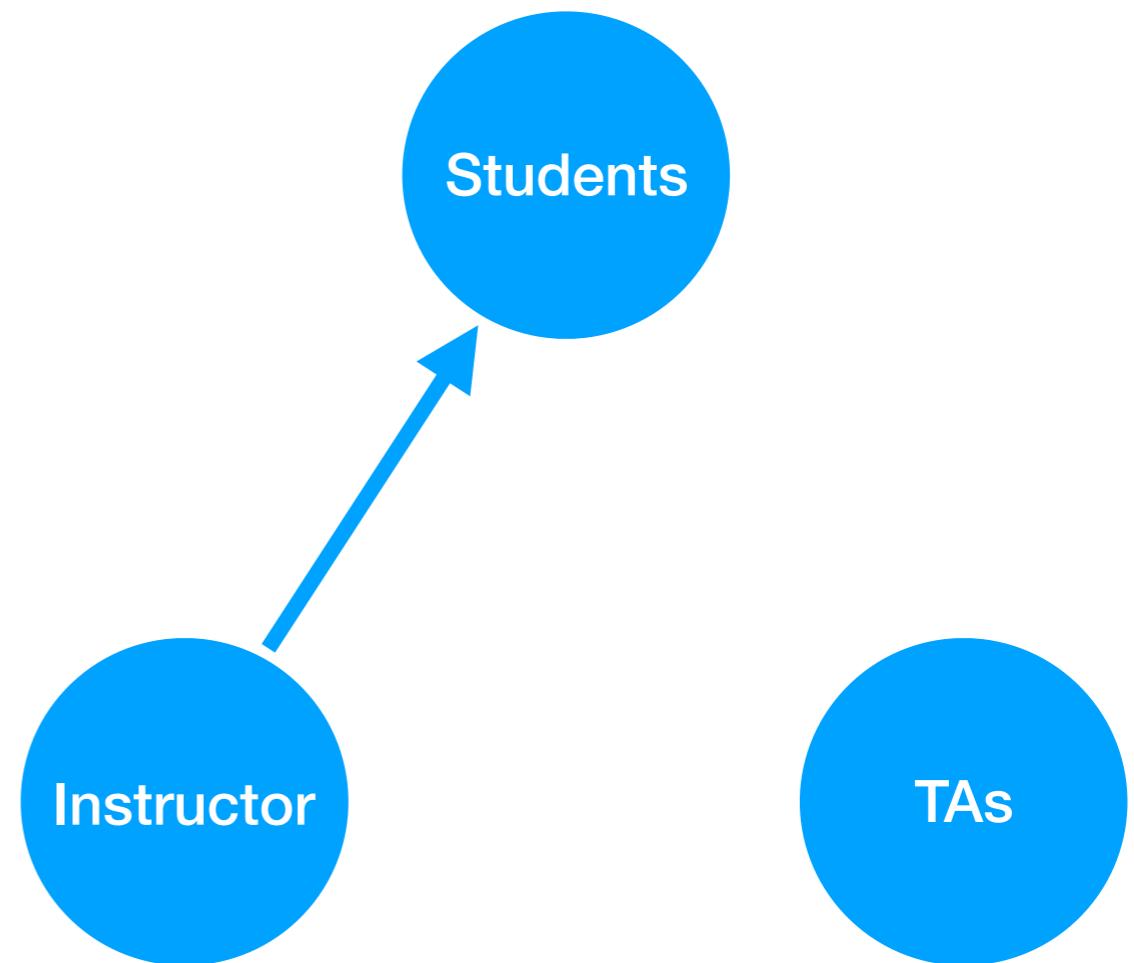
# Communication is CS 301

**Good communication is critical for a class of this size**

- Who needs to communicate: students, TAs, instructors

**Besides direct email, we'll use five communication tools**

- Piazza
- Email lists
- Feedback Form
- Project Submission
- Canvas



# Today's Topics

Introductions

Course overview

- Learning objectives
- Lecture
- Lab
- Readings
- Class communication
- **Grades**
- Projects
- Exams

Computer hardware basics

Website

# Grades

**49%** - programming projects

- **10 projects**, not evenly weighted
- we'll share grading tests with you - **avoid surprise**
- Learning to program is the most import part of the course

**50%** - exams

- 15% midterm 1 (evening)
- 15% midterm 2 (evening)
- 20% final
- times coming soon

**1%** - participation

- filling surveys, following directions, other

**At end of semester, we'll add up your total score,  
then set a curve to determine letter cutoffs.**

# Today's Topics

Introductions

Course overview

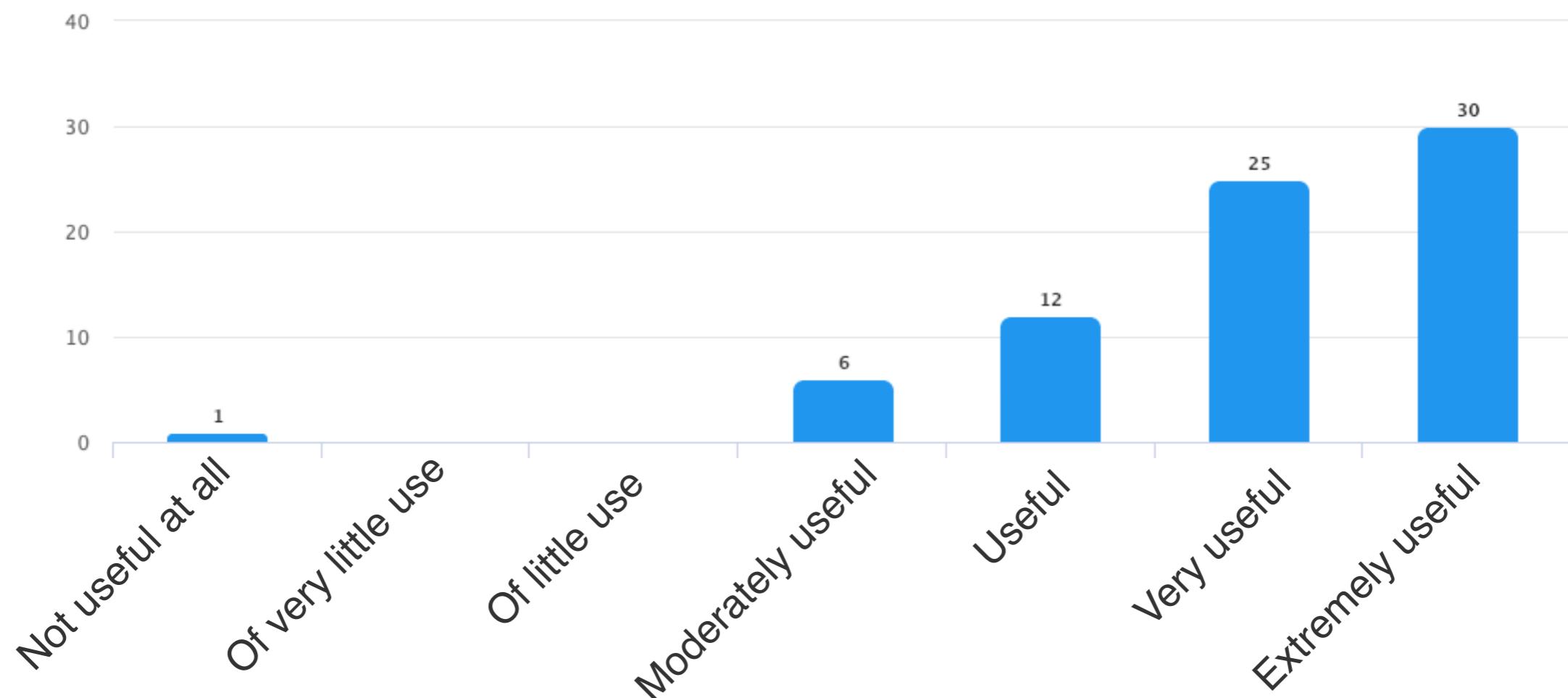
- Learning objectives
- Lecture
- Lab
- Readings
- Class communication
- Grades
- Projects
- Exams

Computer hardware basics

Website

# Prior student reaction to projects

Projects: How useful were projects to your learning?



**Projects are the heart and soul of CS 301**

# Project Overview

**Nearly all projects will relate to some dataset**

## Timeline

- Projects will be due most weeks, on **Wed, at midnight**
- You get 5 late days, use them wisely!
- Contact us about any issues

## Getting help

- Piazza (don't share substantial code) or email (do share code)
- Lab sessions
- Instructor or TA office hours
- CSLC (Computer Science Learning Center)

# Pair Programming

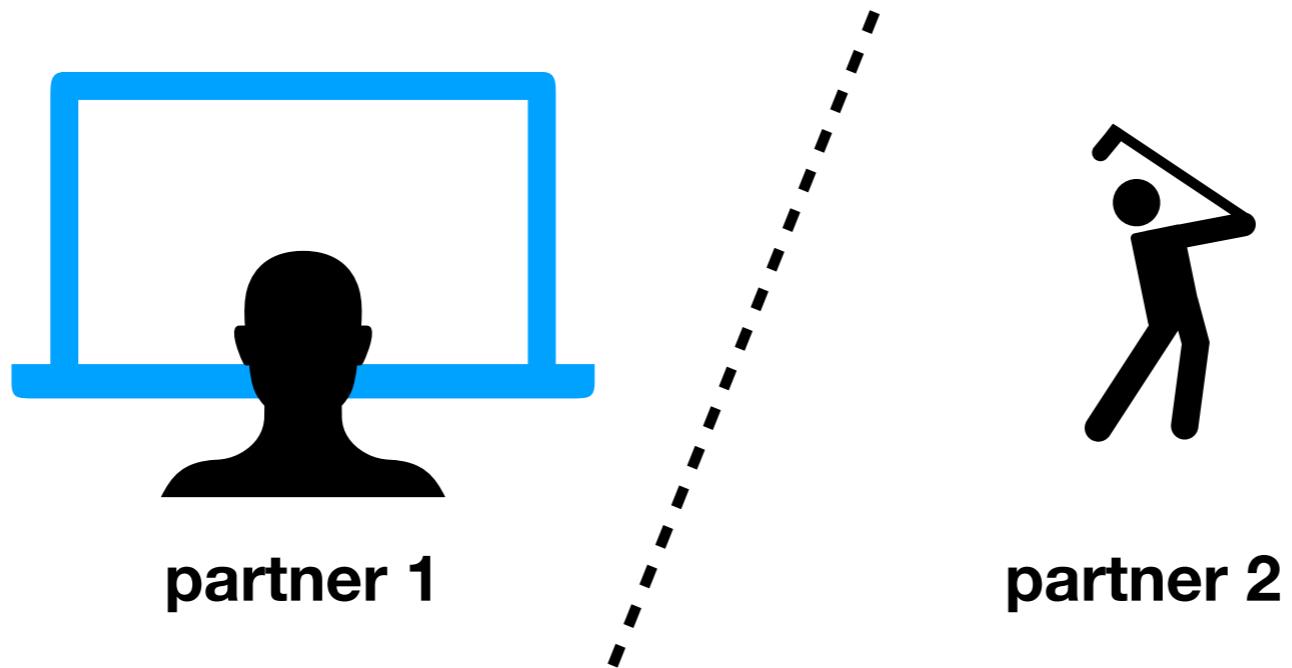
**You can optionally work in pairs of two**

- Partnerships across sections allowed
- Switch partners between projects (or keep with same partner)

# Pair Programming

**You can optionally work in pairs of two**

- Partnerships across sections allowed
- Switch partners between projects (or keep with same partner)

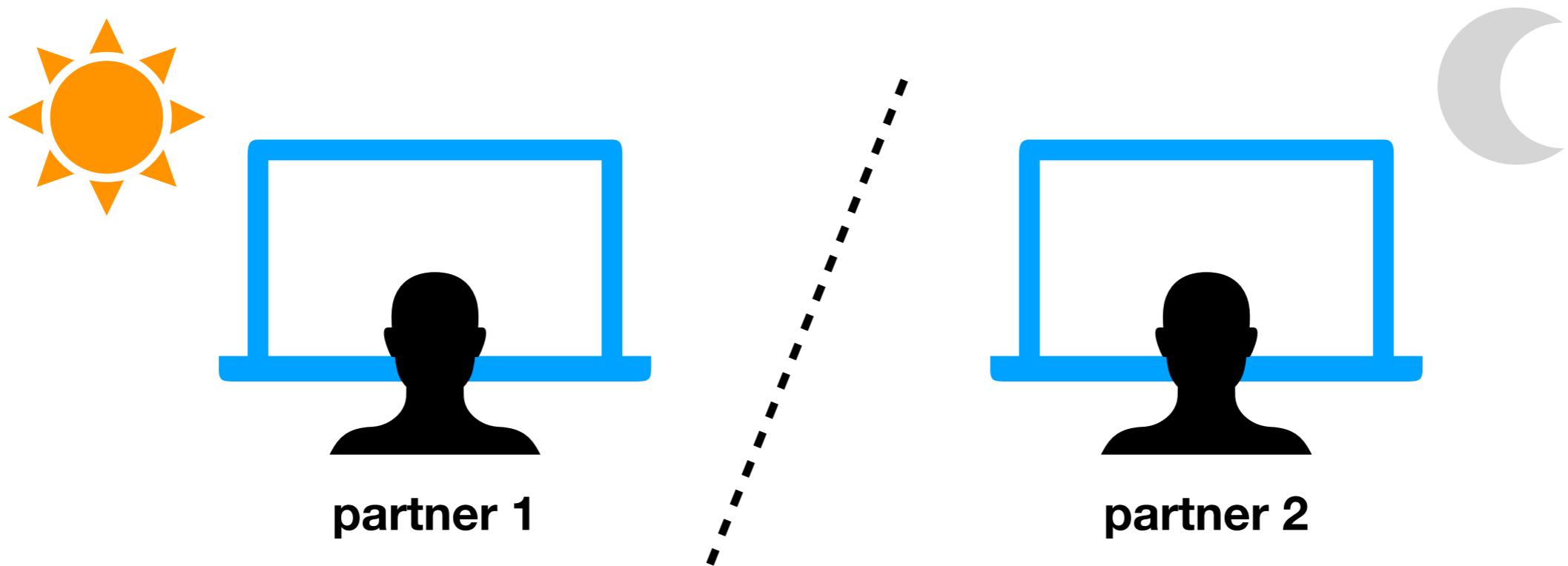


**bad:** partners don't share work

# Pair Programming

**You can optionally work in pairs of two**

- Partnerships across sections allowed
- Switch partners between projects (or keep with same partner)

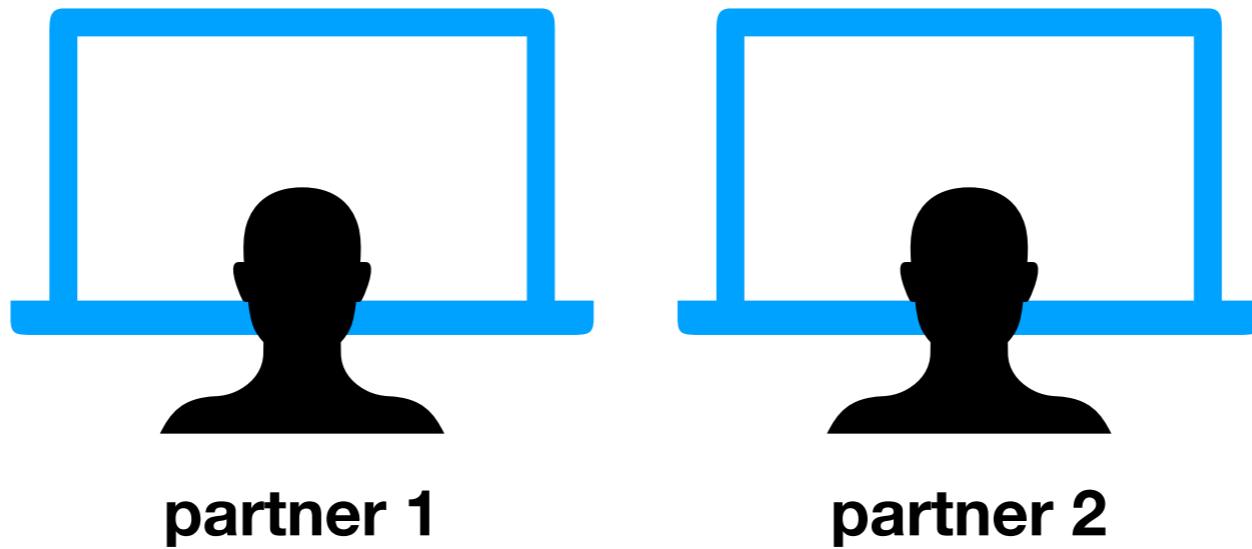


**bad:** working on different parts at different times

# Pair Programming

**You can optionally work in pairs of two**

- Partnerships across sections allowed
- Switch partners between projects (or keep with same partner)

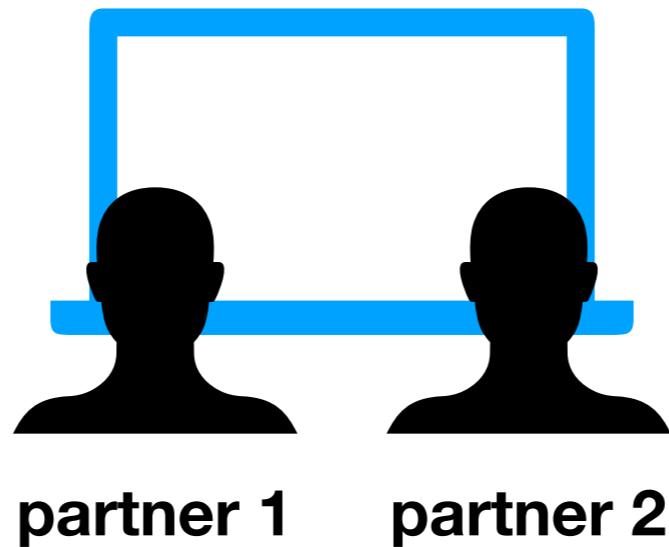


**better:** working alongside each other

# Pair Programming

**You can optionally work in pairs of two**

- Partnerships across sections allowed
- Switch partners between projects (or keep with same partner)

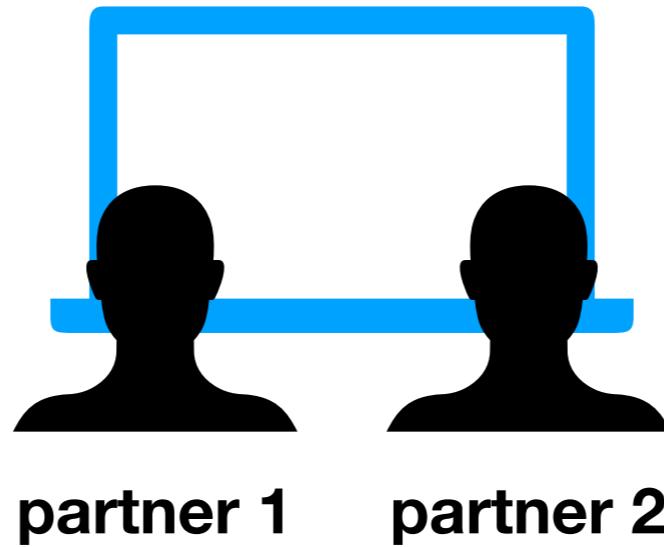


**best:** working on same computer

# Pair Programming

**You can optionally work in pairs of two**

- Partnerships across sections allowed
- Switch partners between projects (or keep with same partner)



## Suggestions

- Take turns coding (don't be greedy/aggressive!)
- One person types, other makes suggestions and thinks about design

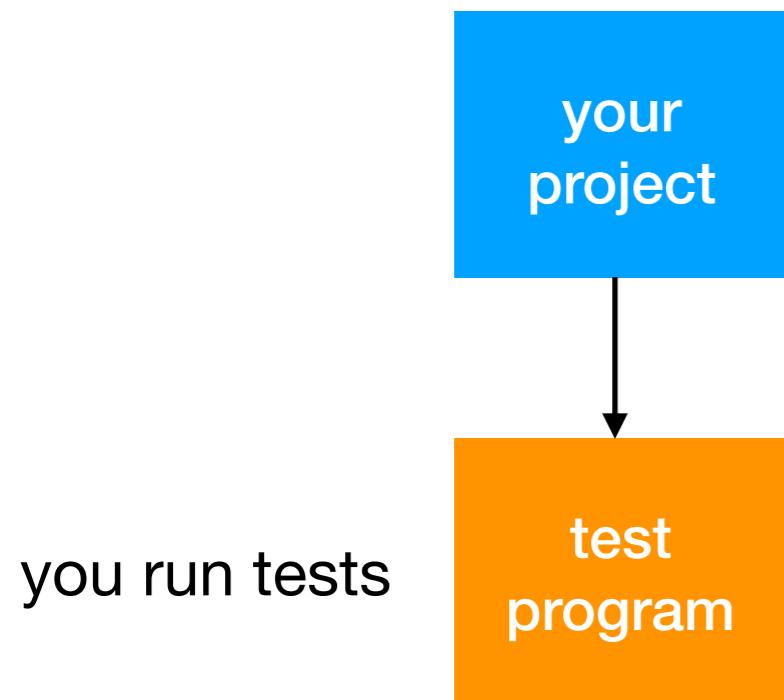
# Project Grading

**YOU**

your  
project

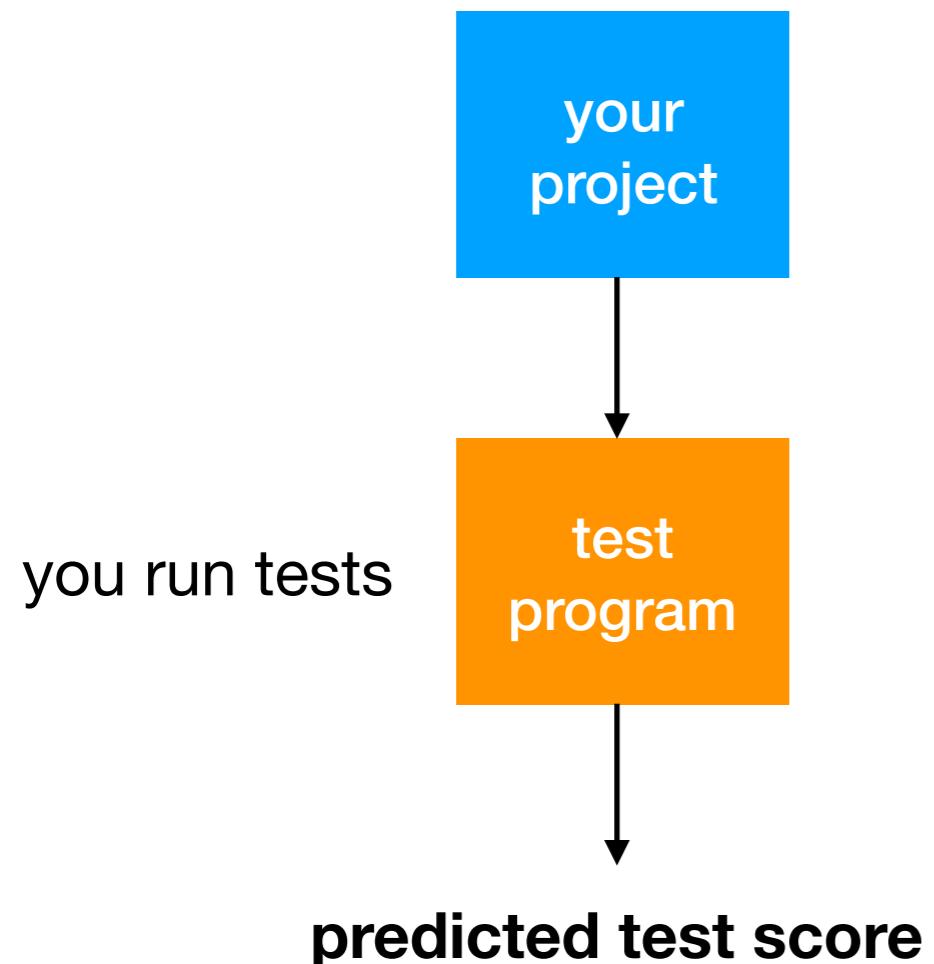
# Project Grading

**YOU**

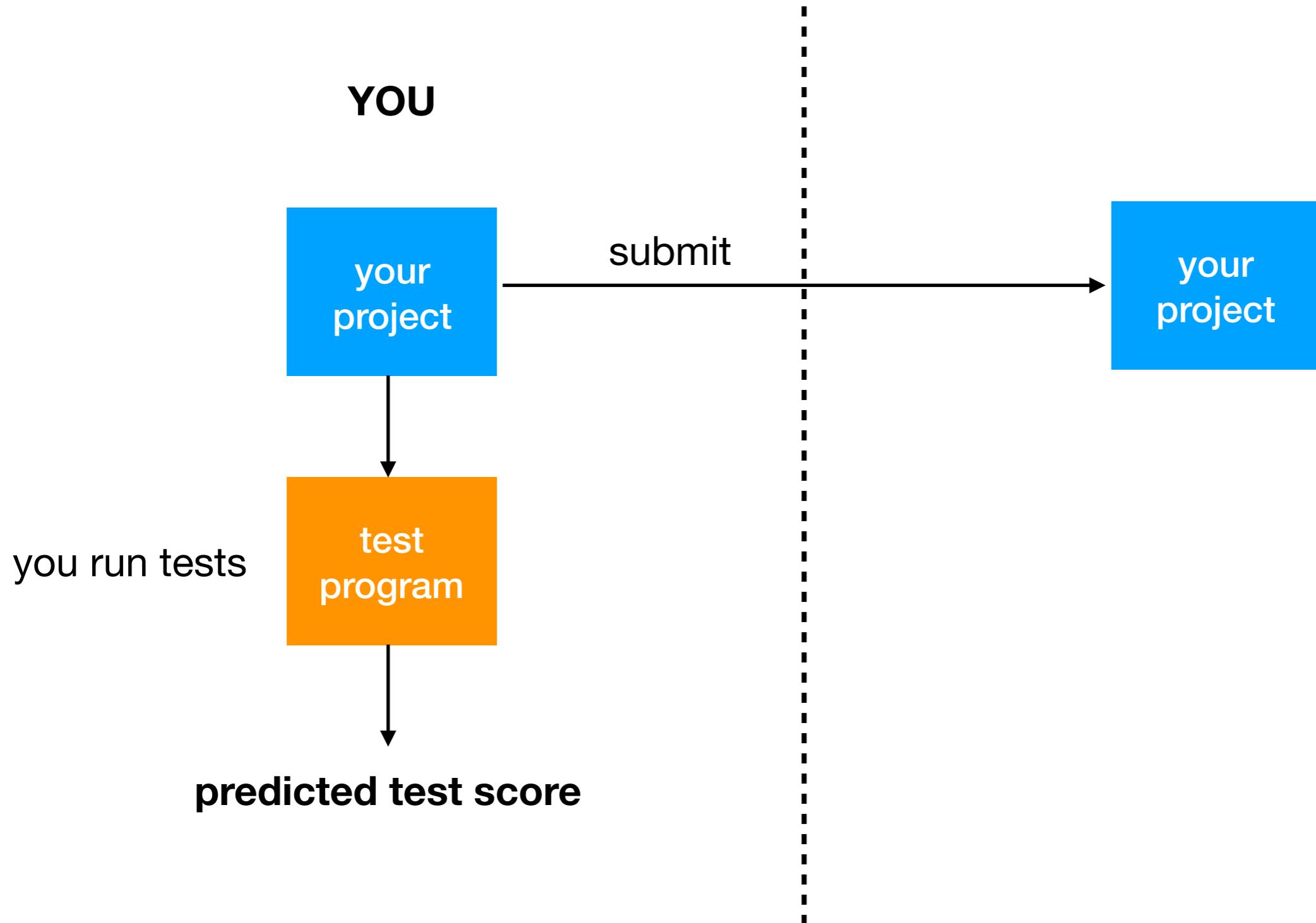


# Project Grading

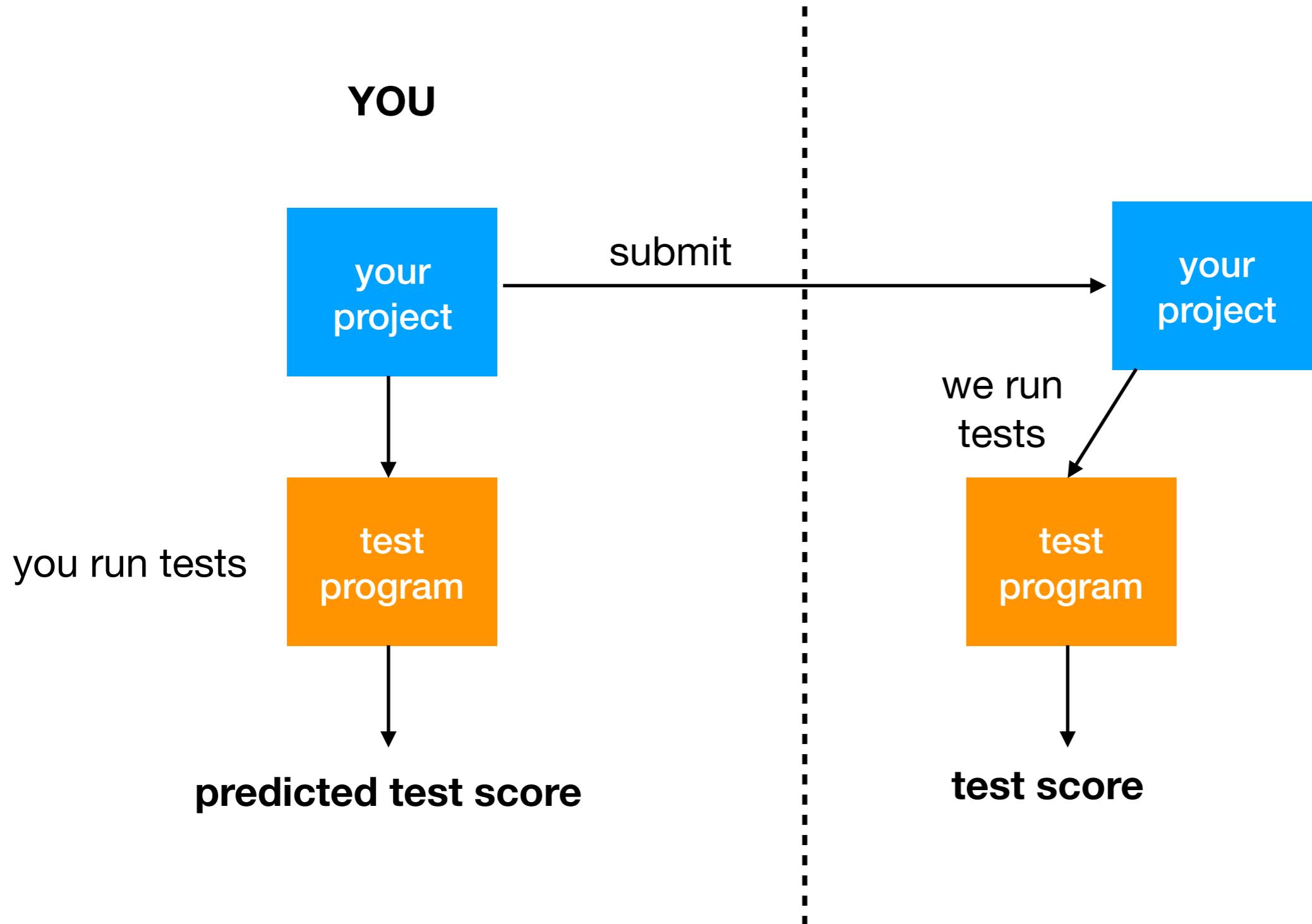
**YOU**



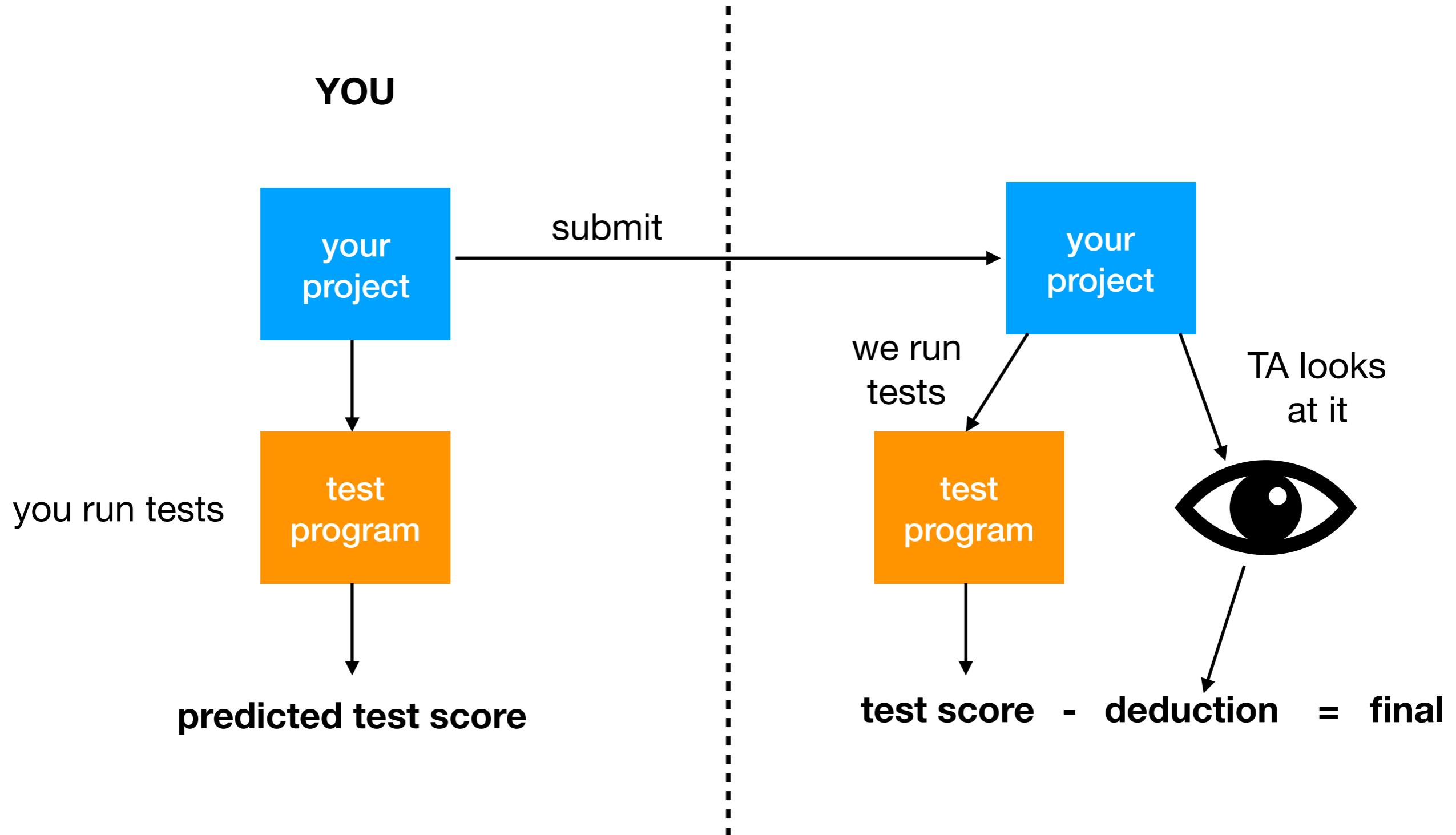
# Project Grading



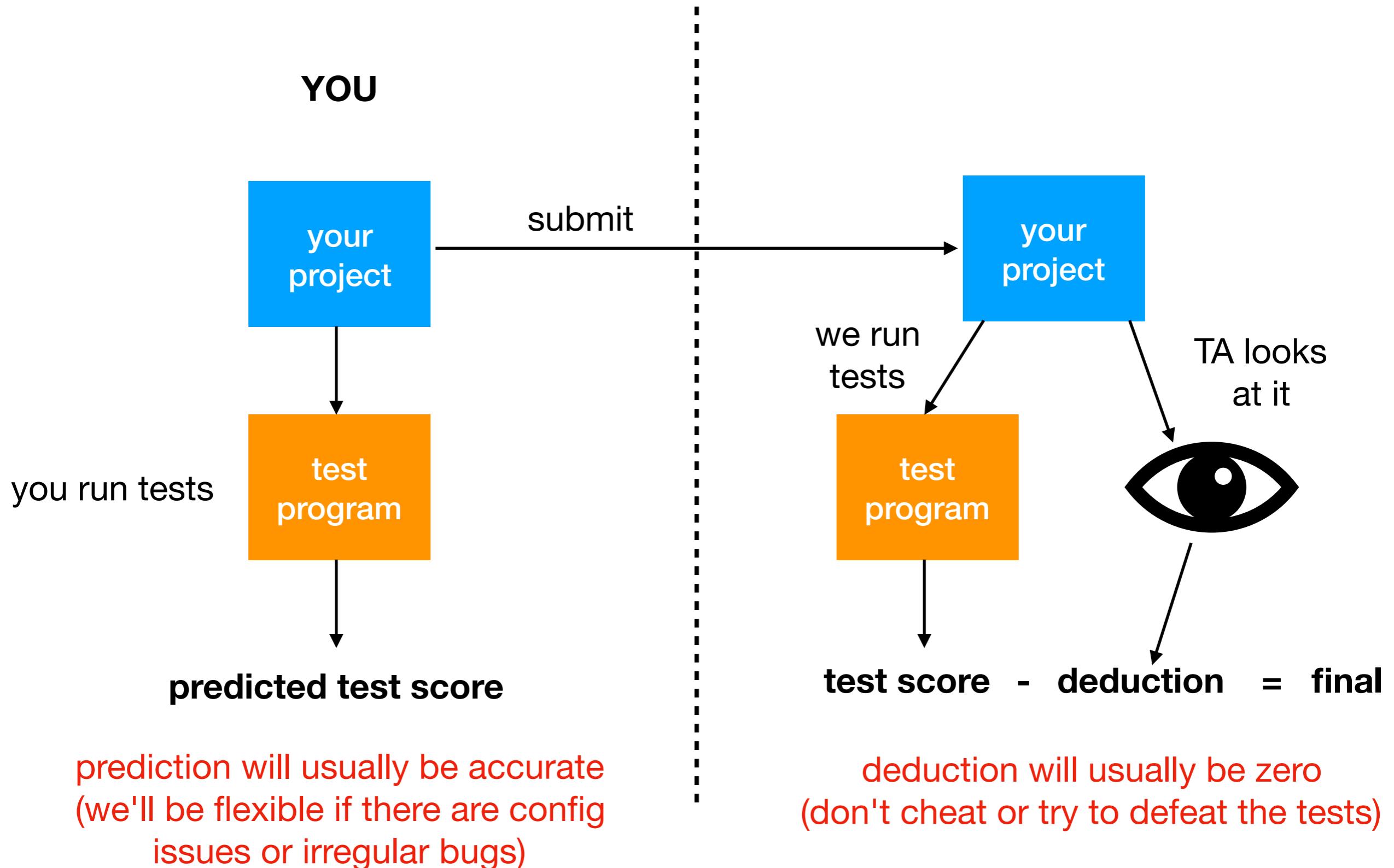
# Project Grading



# Project Grading

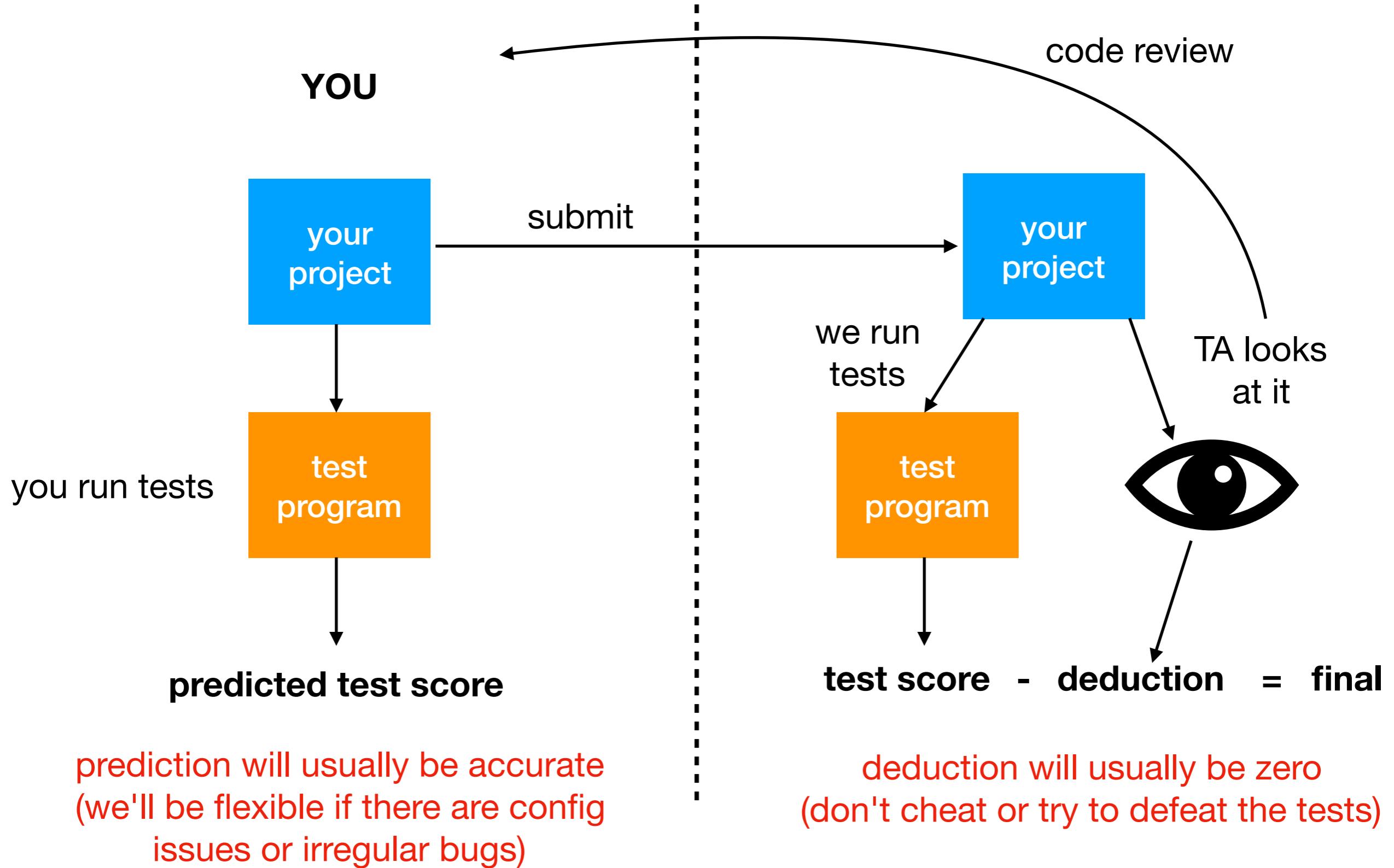


# Project Grading



# Project Grading

feedback is mostly about how to do things better or more simply (valuable even if you score 100%)



# Today's Topics

Introductions

Course overview

- Learning objectives
- Lecture
- Lab
- Readings
- Class communication
- Grades
- Projects
- Exams

Computer hardware basics

Website

# Exams

## **There will be two midterms and one final**

- Check website for dates/locations
- One 8.5 by 11 in note sheet allowed only
- Exams will be multiple choice scantron

## **Contents**

- cumulative
- ideally not much time pressure
- one goal: reward project partners doing more work over those slacking

# Today's Topics

Introductions

Course overview

## Computer hardware basics

- Input/Output
- CPU
- Memory
- Storage
- Networking

Website

# Today's Topics

Introductions

Course overview

Computer hardware basics

- Input/Output
- CPU
- Memory
- Storage
- Networking

Website

# Input/Output

I/O (stands for input/output)

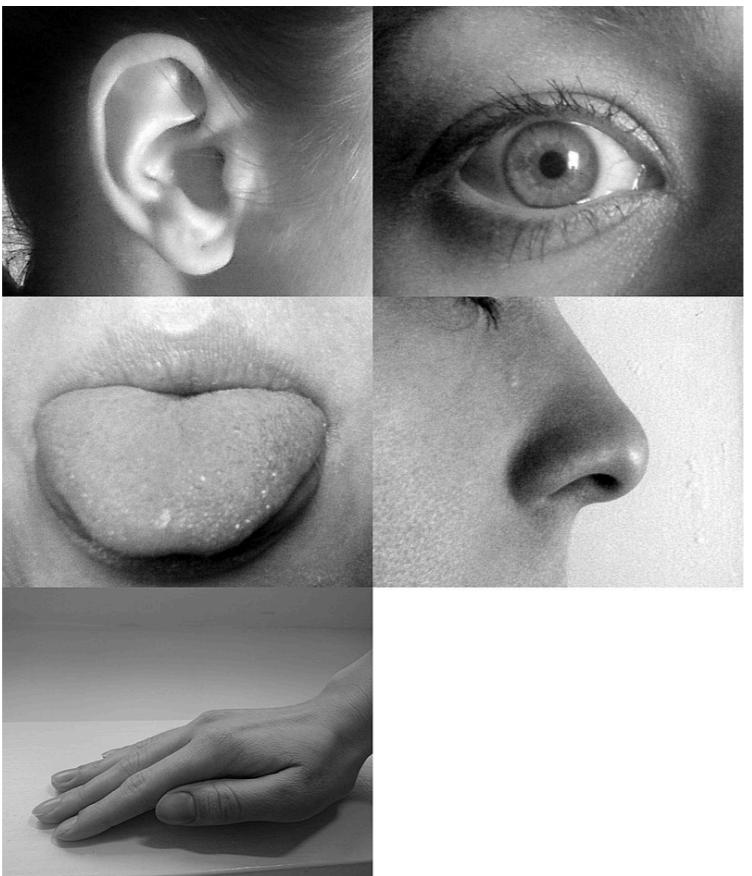
- What are examples for human?

# Input/Output

I/O (stands for input/output)

- What are examples for human?

**input: senses**

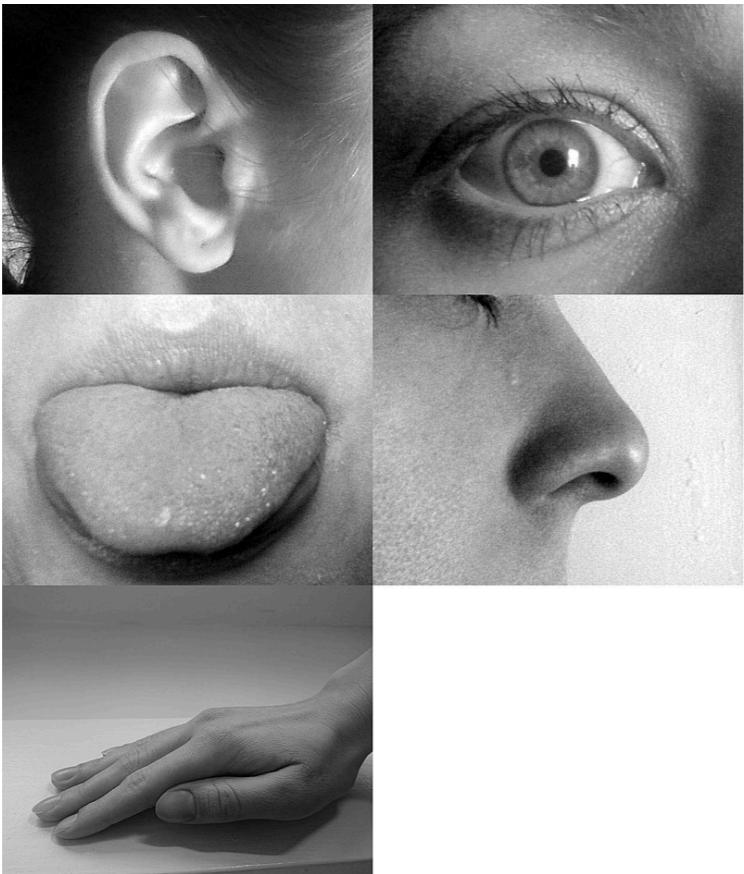


# Input/Output

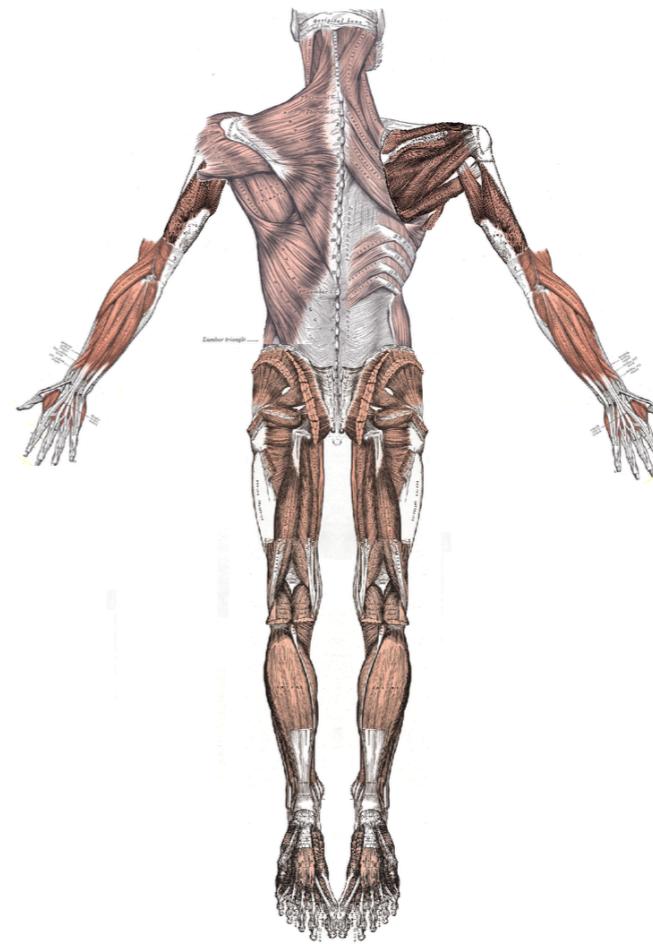
# I/O (stands for input/output)

- What are examples for human?

# **input: senses**



# output: muscles



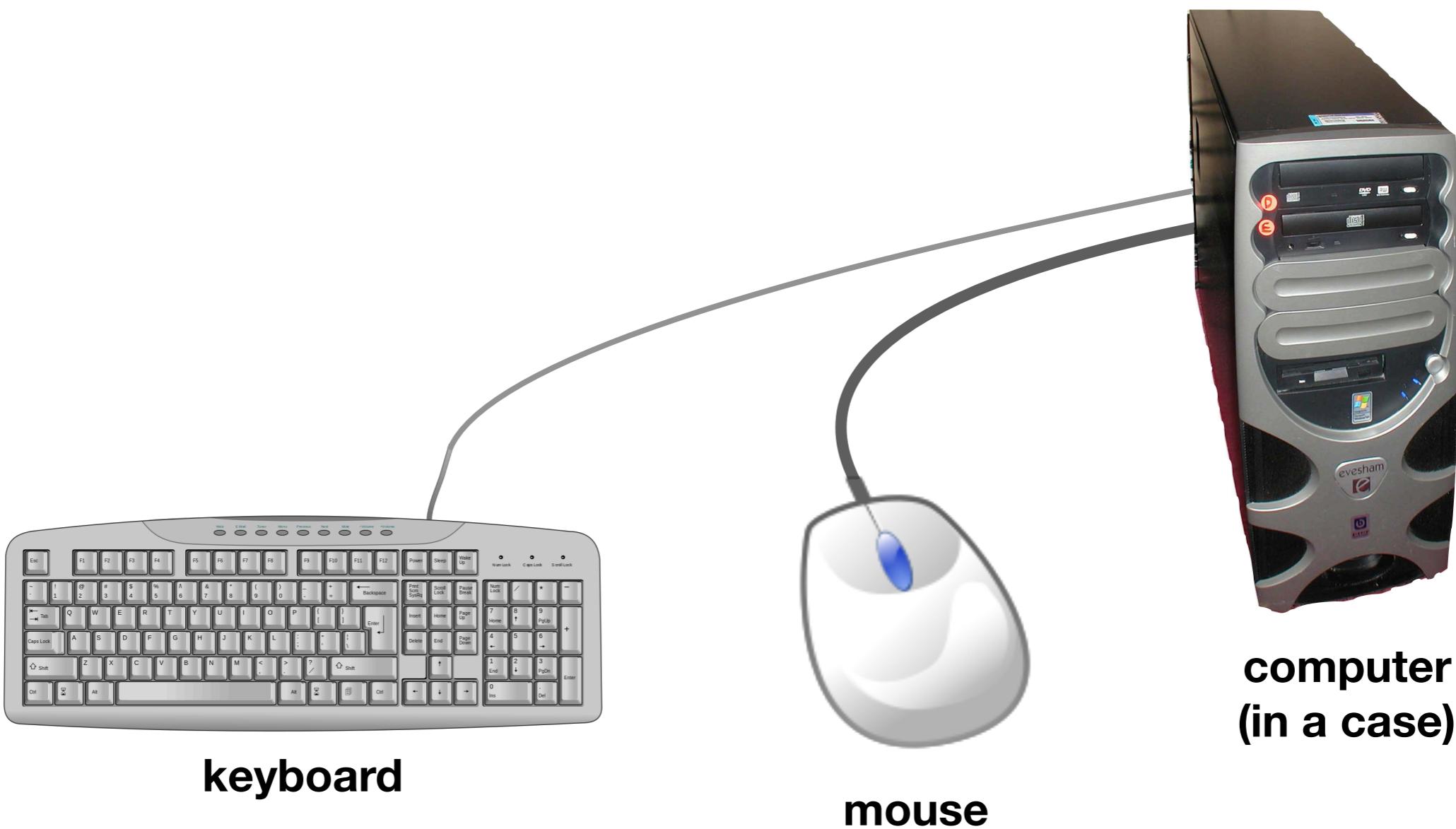
# Computer Input/Output

**what are some common compute inputs?**



**computer  
(in a case)**

# Computer Input/Output

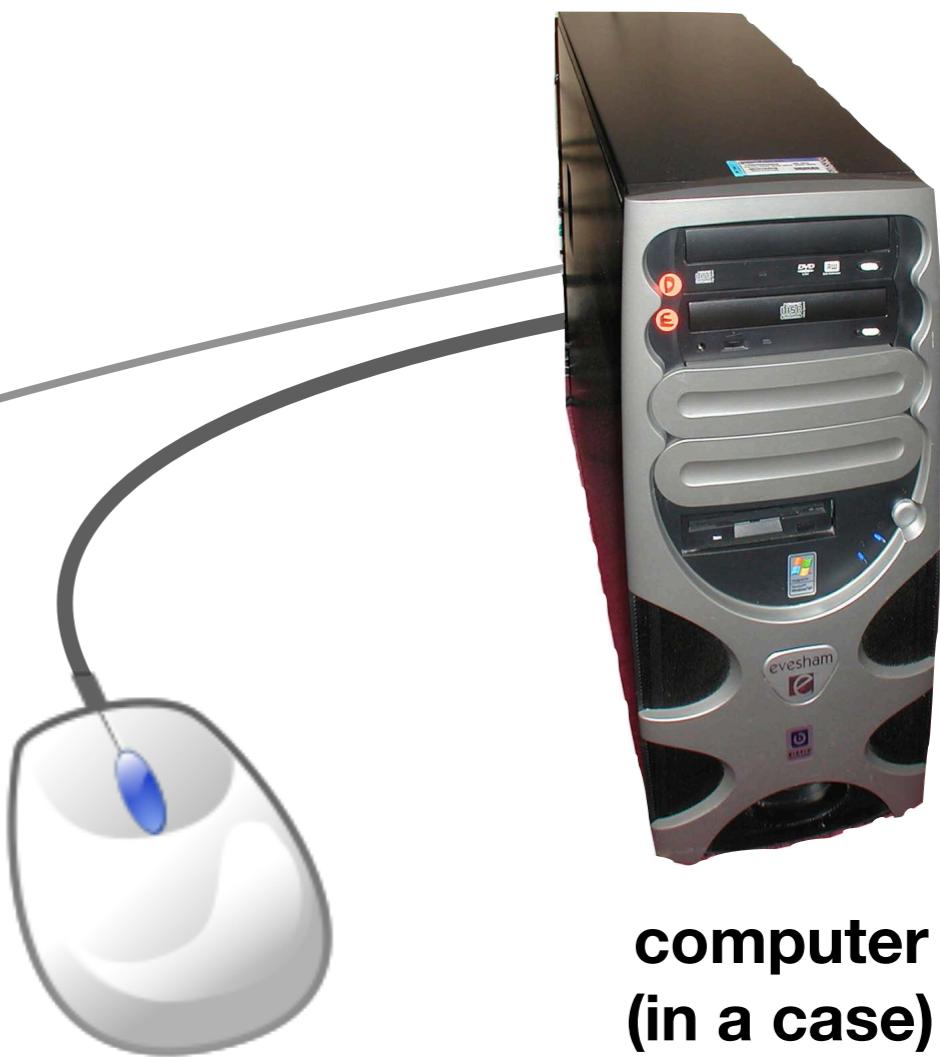


# Computer Input/Output

**what are some common compute outputs?**



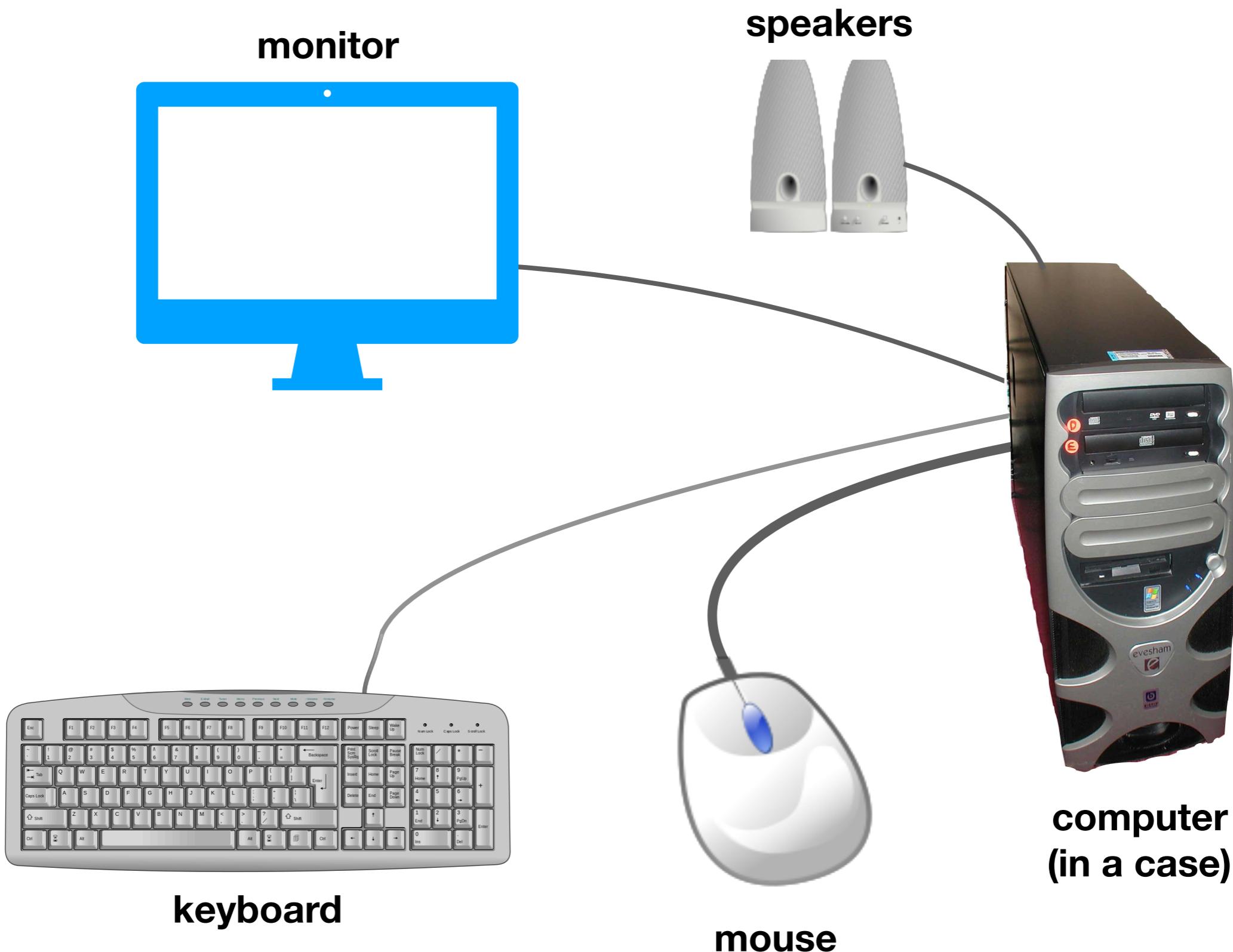
**keyboard**



**mouse**

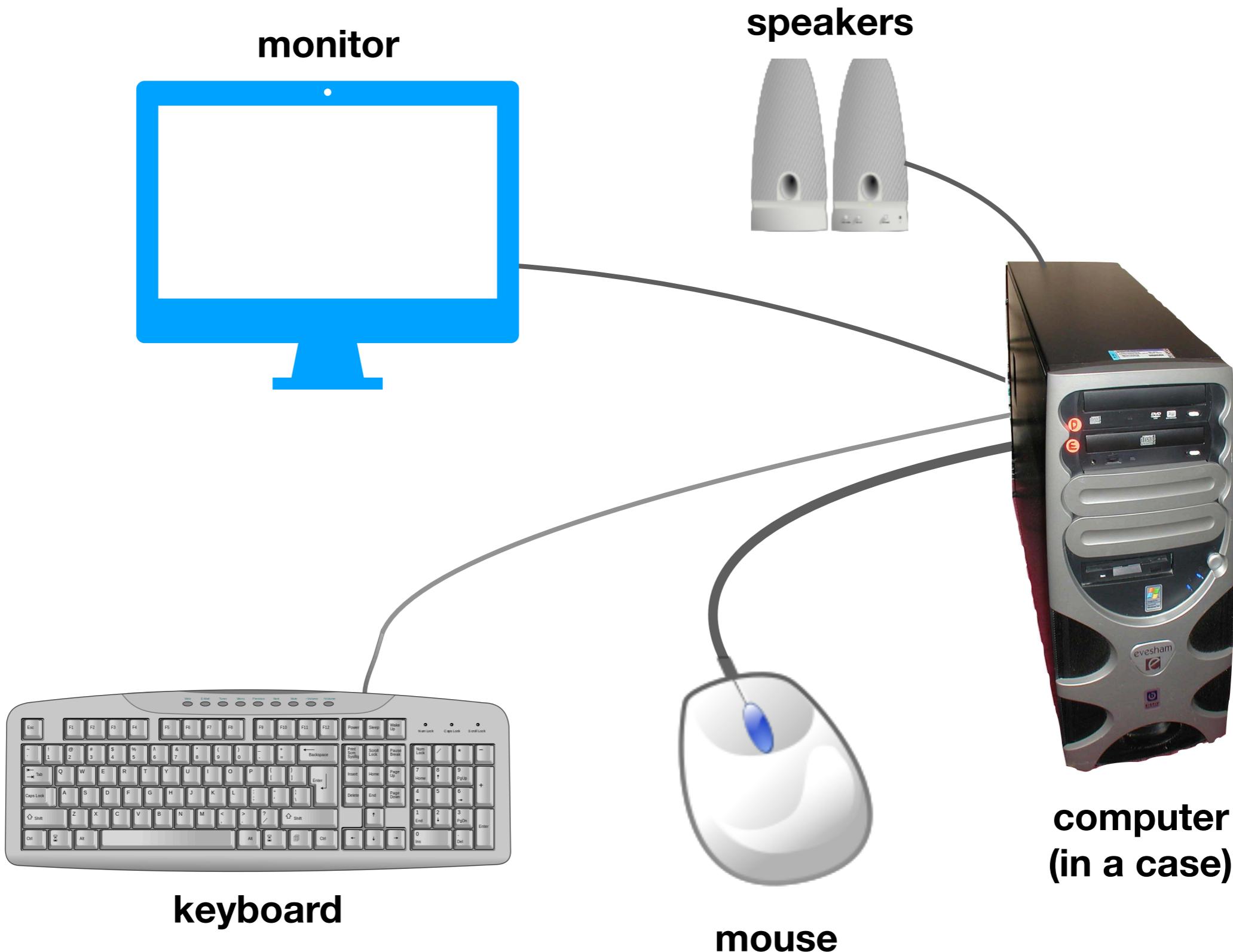
**computer  
(in a case)**

# Computer Input/Output



# Computer Input/Output

I/O devices attach via “ports” (e.g. USB) in back of computer



# Computer Input/Output

monitor



speakers



**What's inside a  
computer case?**



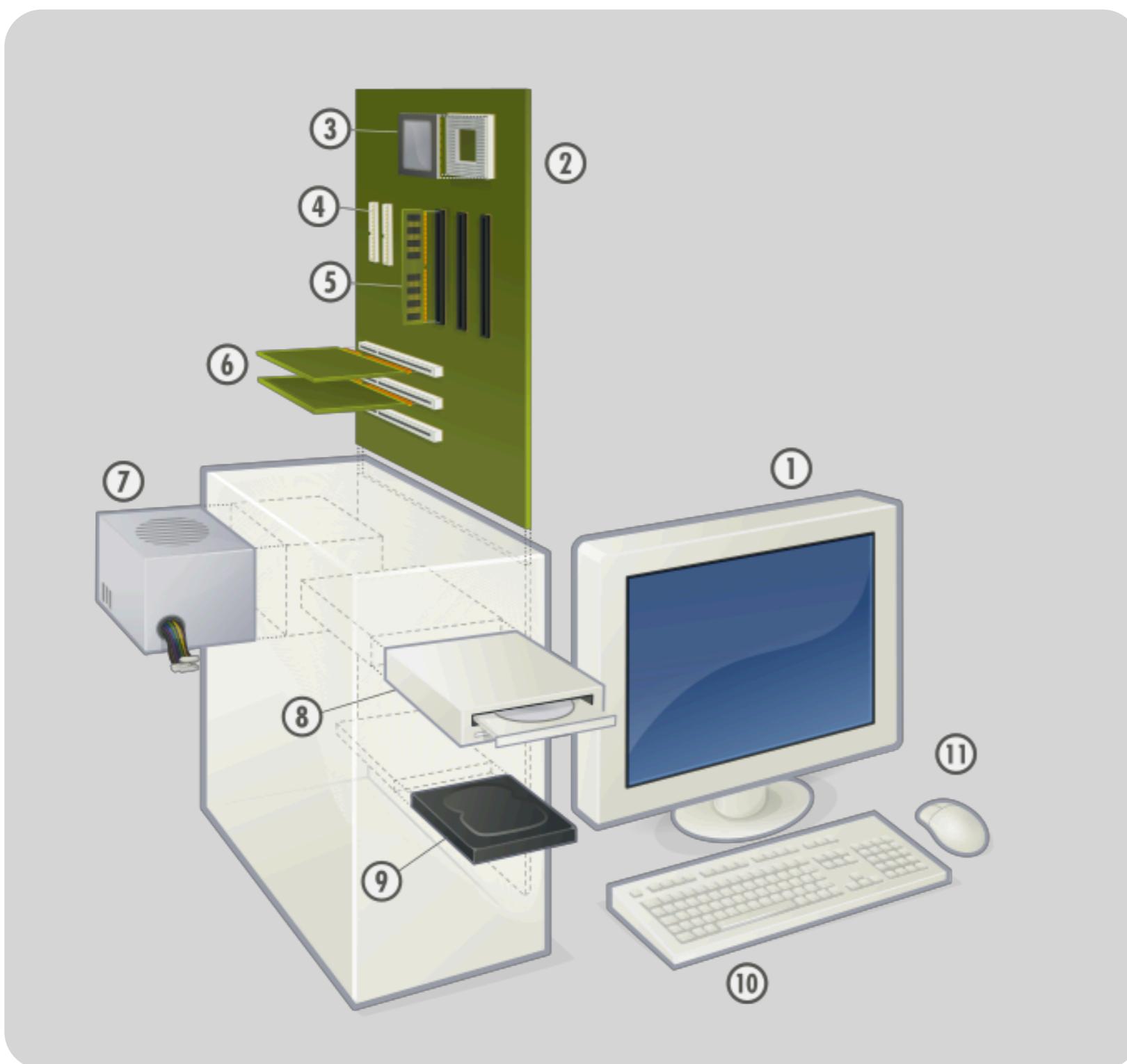
keyboard

mouse

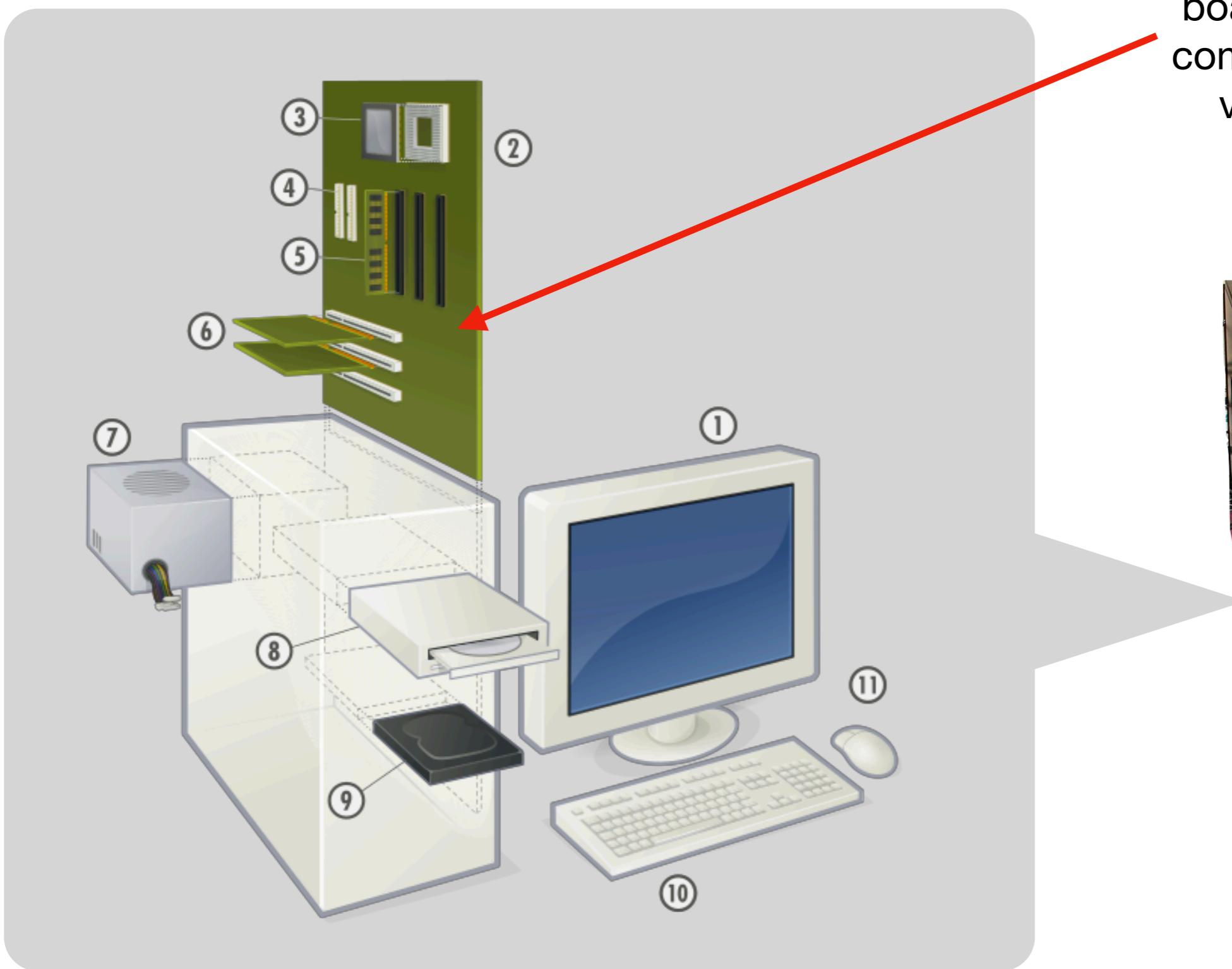


**computer  
(in a case)**

# Computer Internals



# Computer Internals



**Motherboard:** main circuit board to which other components connect, via sockets/slots



# Today's Topics

Introductions

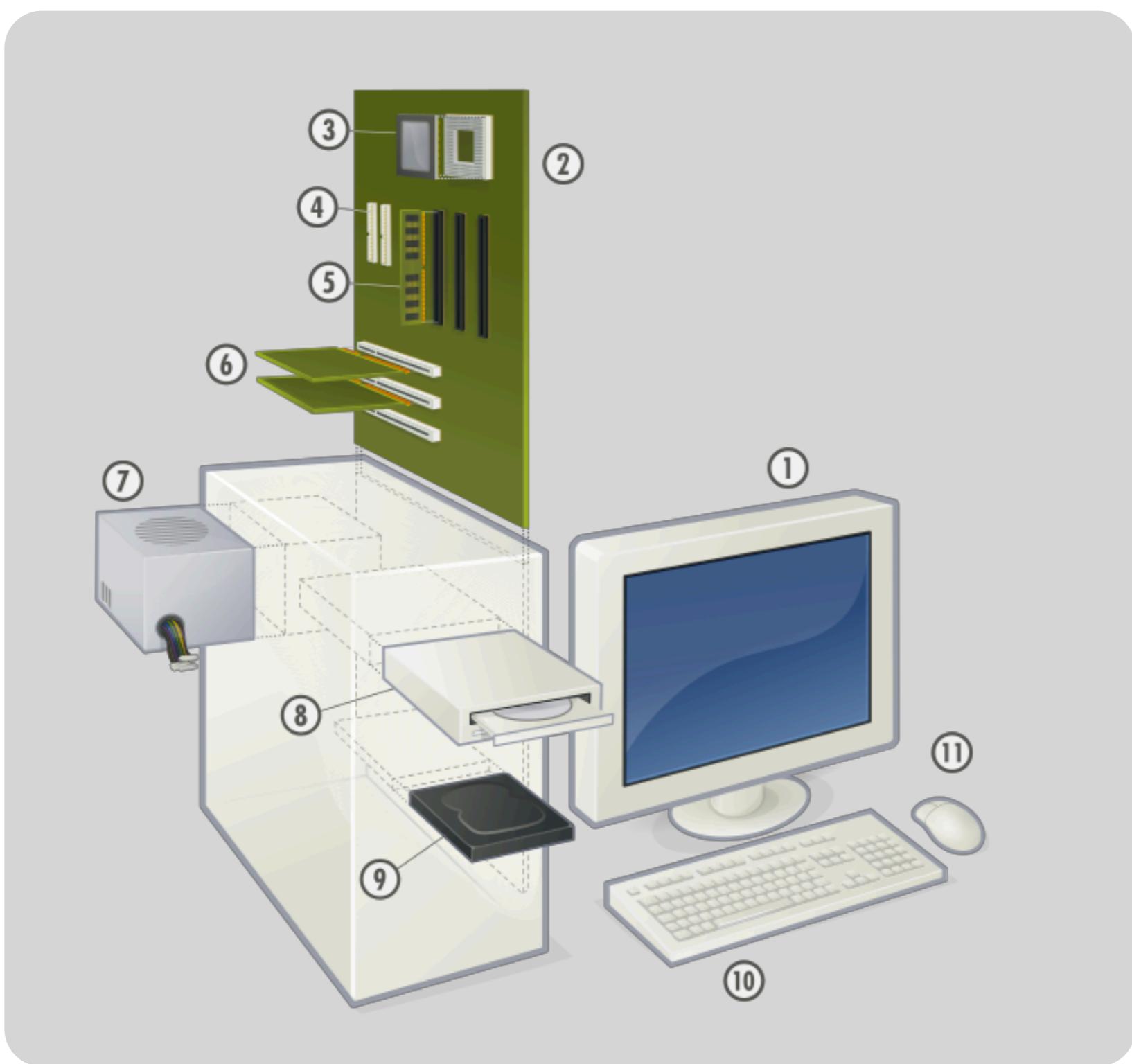
Course overview

Computer hardware basics

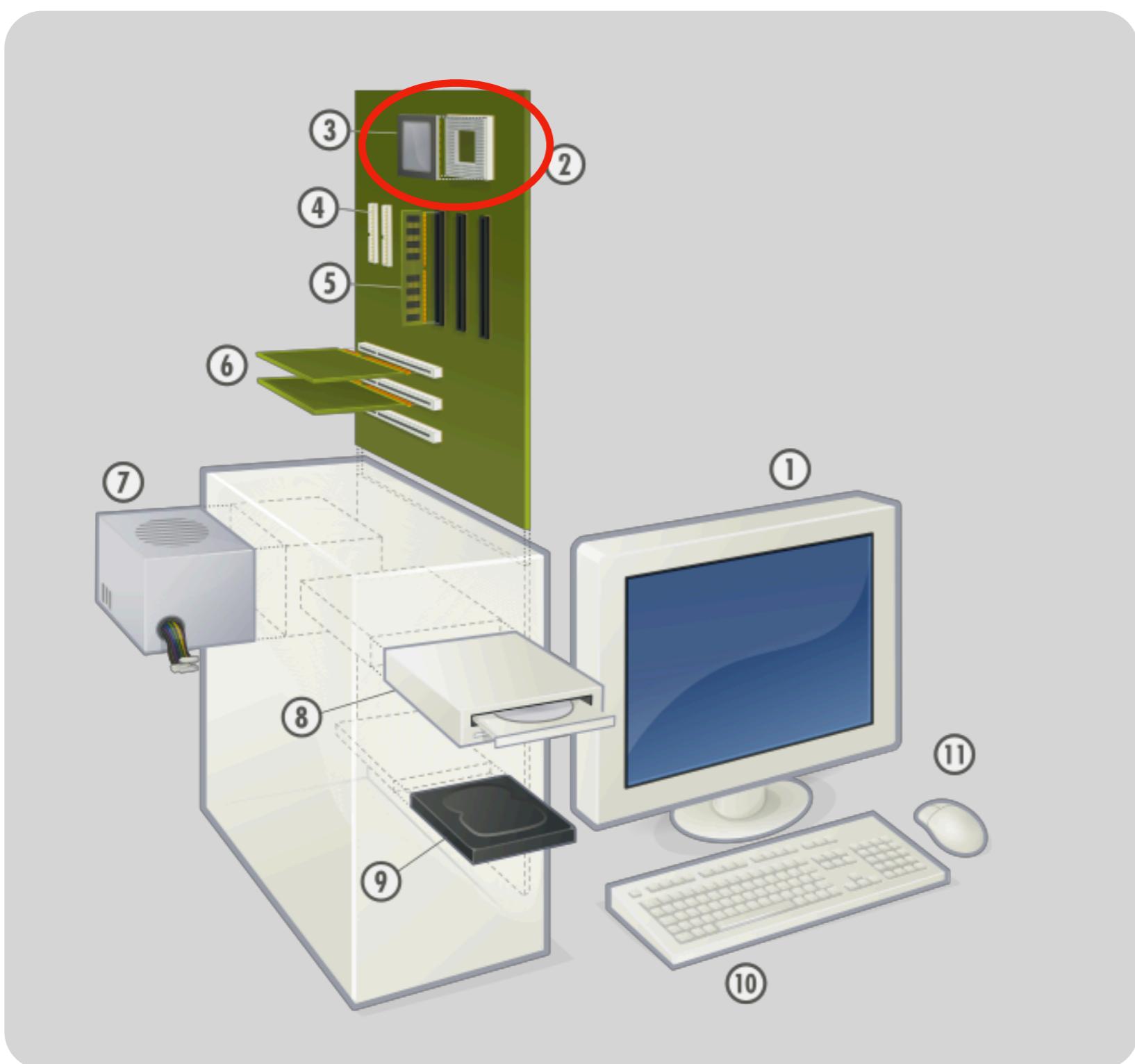
- Input/Output
- CPU
- Memory
- Storage
- Networking

Website

# Central Processing Unit (CPU)



# Central Processing Unit (CPU)



# CPU

Responsible for computation

- Runs code
- Performs addition, other math
- Compares numbers, text
- Receives input, sends output
- Some compare it to a “brain”



Runs on a clock

- Typically a couple GHz (i.e., billions of ticks per second)
- High-speed makes CPUs hot, require fans/cooling
- Overclocking: running on faster clock than CPU is designed for

Computers often have multiple CPUs

- Motherboard may have multiple sockets
- Single chip may contain multiple CPUs
- Allows computers to do more things simultaneously

# Today's Topics

Introductions

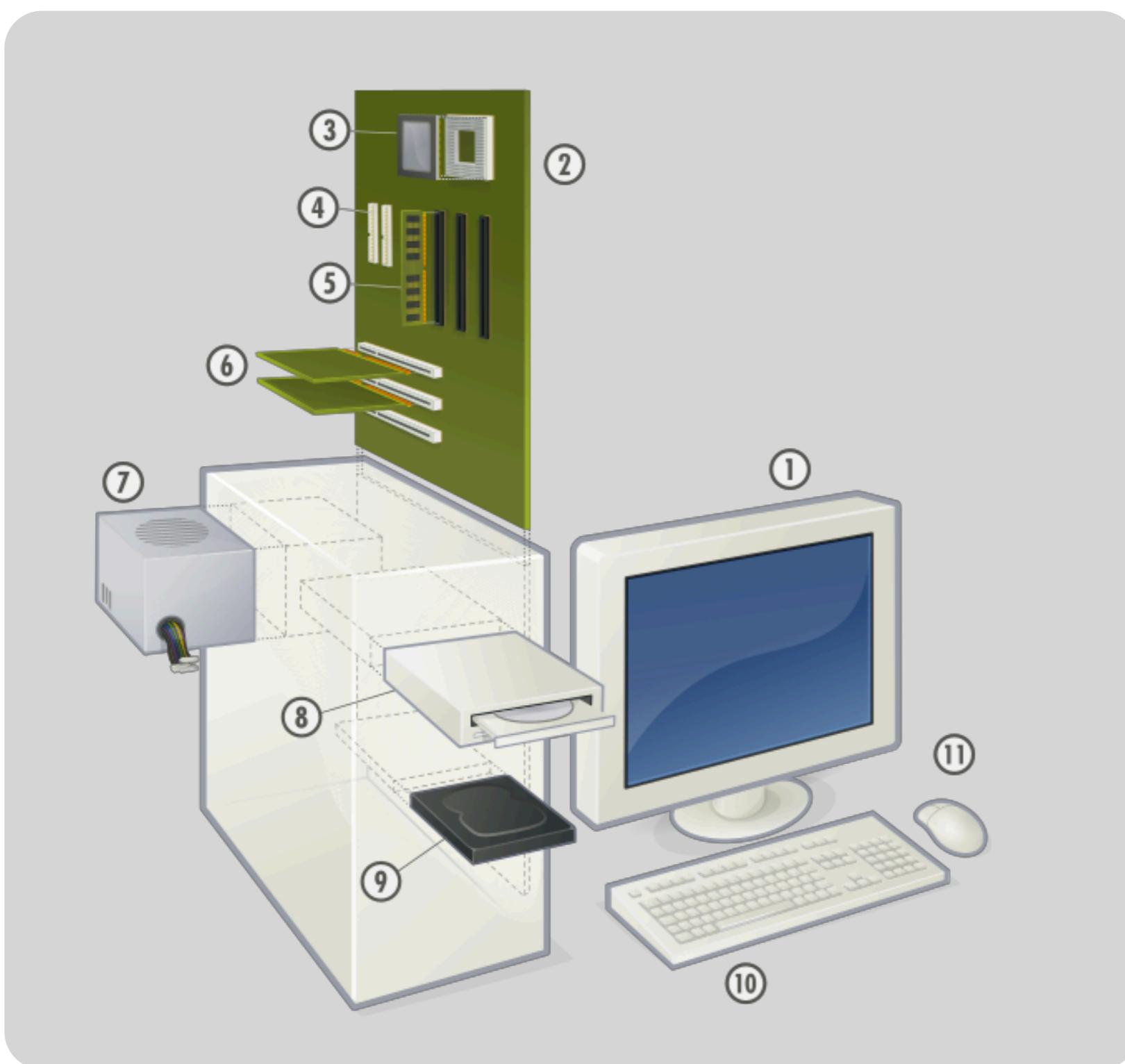
Course overview

Computer hardware basics

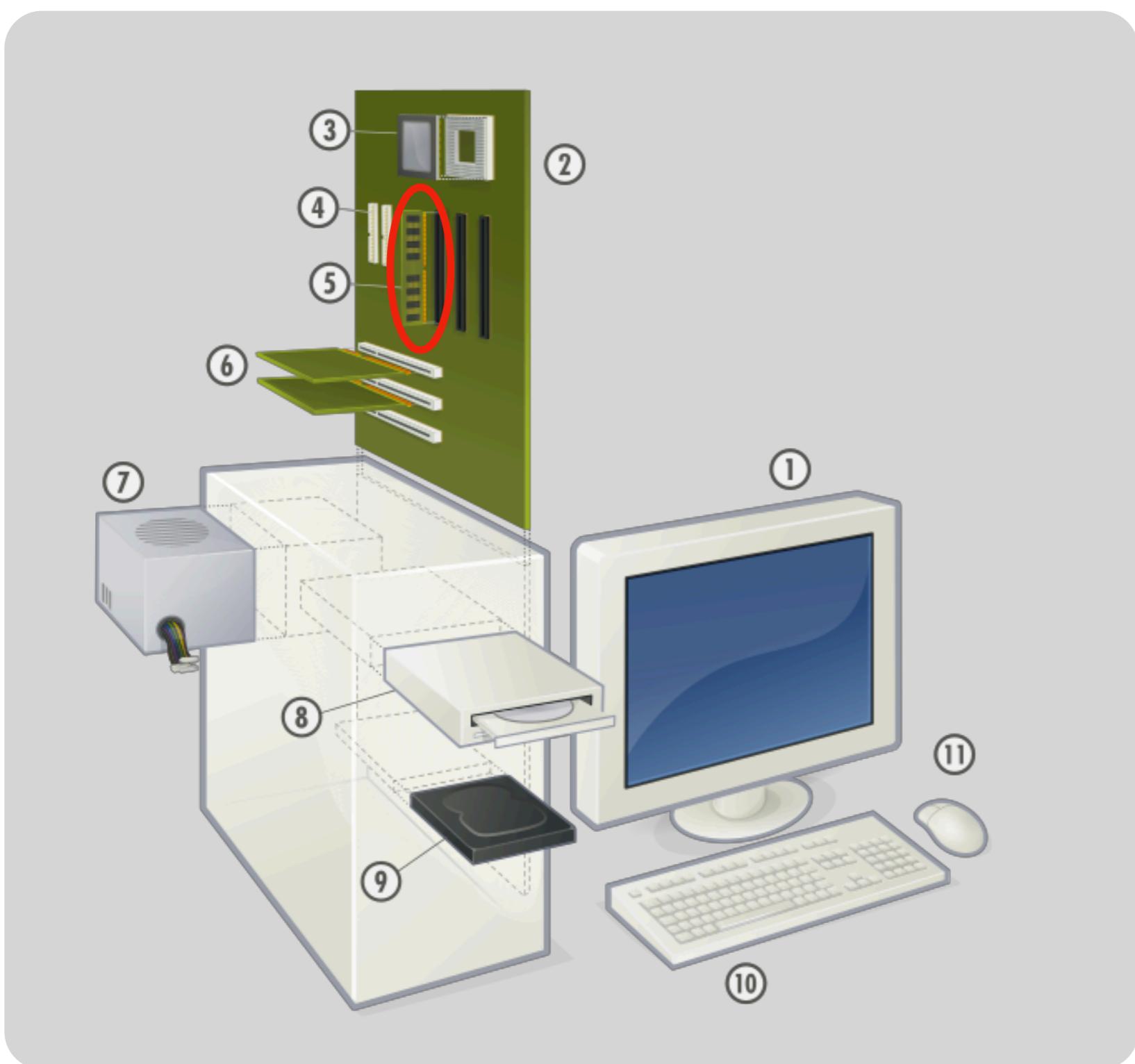
- Input/Output
- CPU
- **Memory**
- Storage
- Networking

Website

# Random Access Memory (RAM)



# Random Access Memory (RAM)



# Memory

Memory stores data for short term

- RAM is most common form today (don't worry about specifics)
- CPU sends data to/from memory
- Accessing it is very fast
- It is “volatile” – meaning you lose this data when you power off your computer
- You don't save “files” in memory, otherwise they would be gone!

Stores bytes of data

- We'll talk about bytes more later; for now, one byte ≈ one letter
- The text “hello” requires 5 bytes
- Typical personal computer has few to tens of gigabytes (billion bytes) of memory



# Today's Topics

Introductions

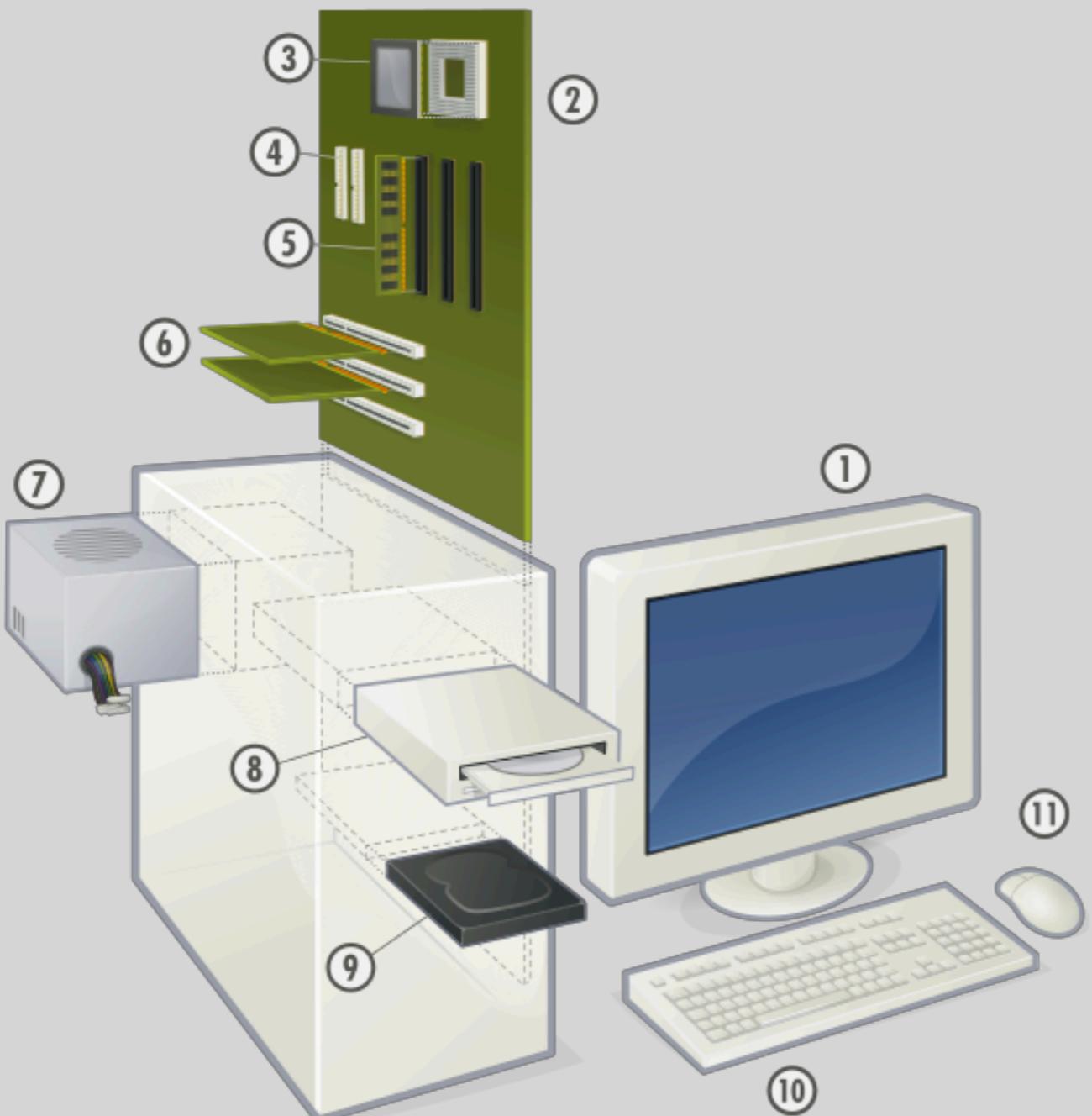
Course overview

Computer hardware basics

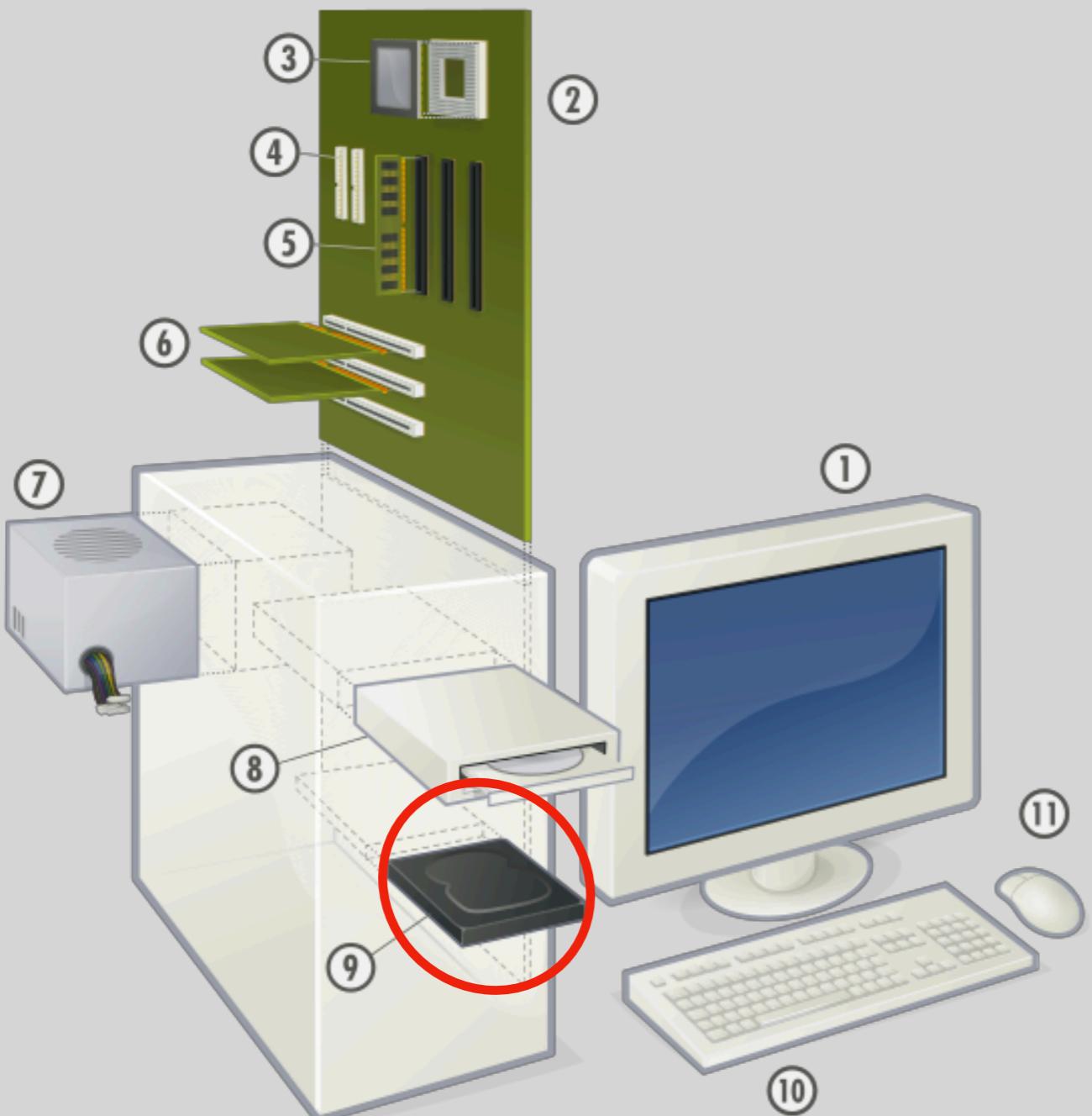
- Input/Output
- CPU
- Memory
- Storage
- Networking

Website

# Storage Drives



# Storage Drives



# Storage Drives

Two common devices

- HDD (hard disk drive), has moving parts, cheap, slow
- SSD (solid state drive), no moving parts, expensive, fast
- RAM is much faster than either HDDs or SSDs

Storage devices used to save data after power down

- **Persistent** medium, in contrast to **volatile** RAM
- Typical capacity: hundreds of gigabytes

When you make a directory/folder or **save a file**, that data is ultimately getting recorded to your storage device

- **Sometimes computers save to RAM first, and only to the device later; power down cleanly to avoid losing your data!!!**

# Today's Topics

Introductions

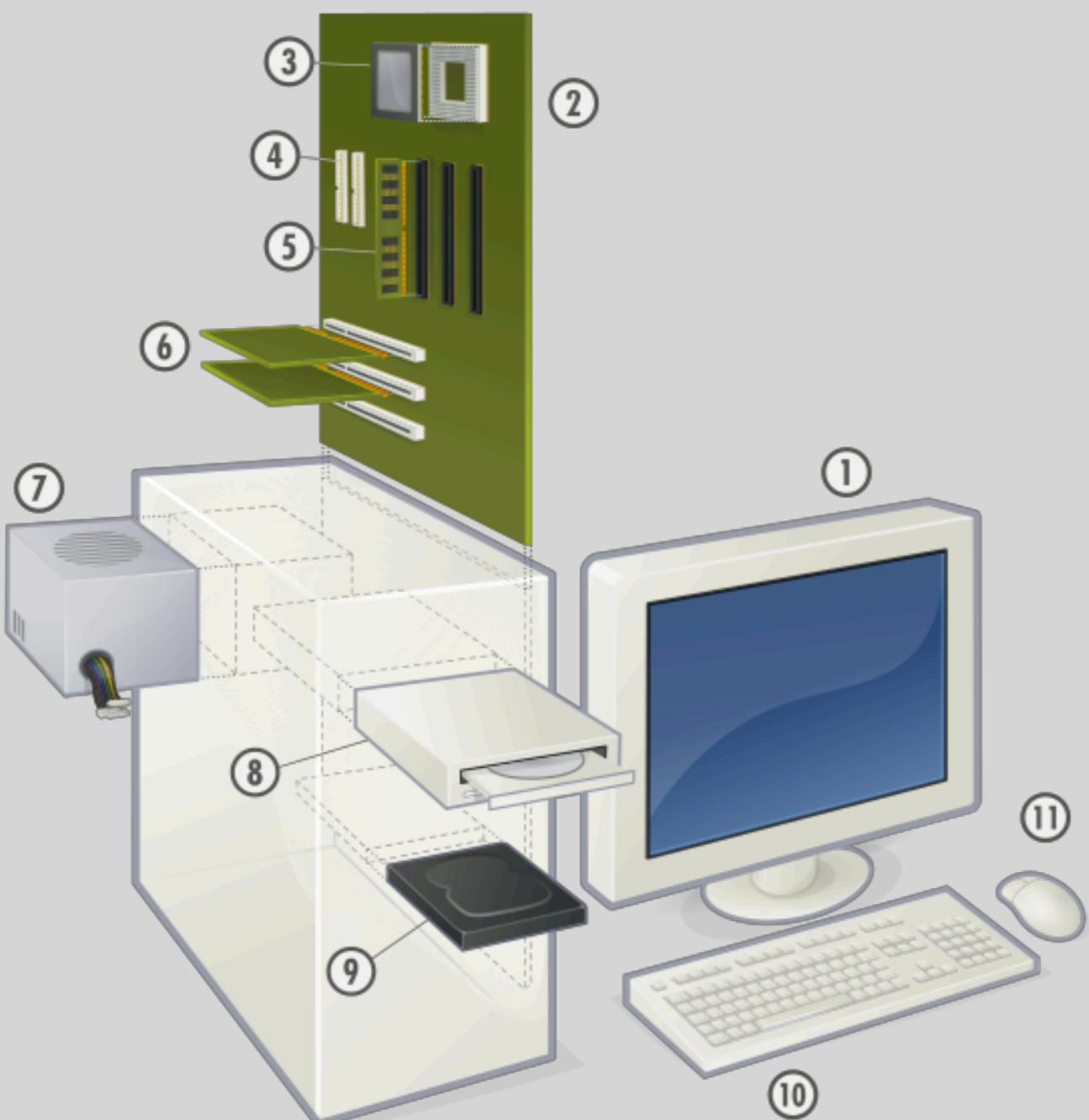
Course overview

Computer hardware basics

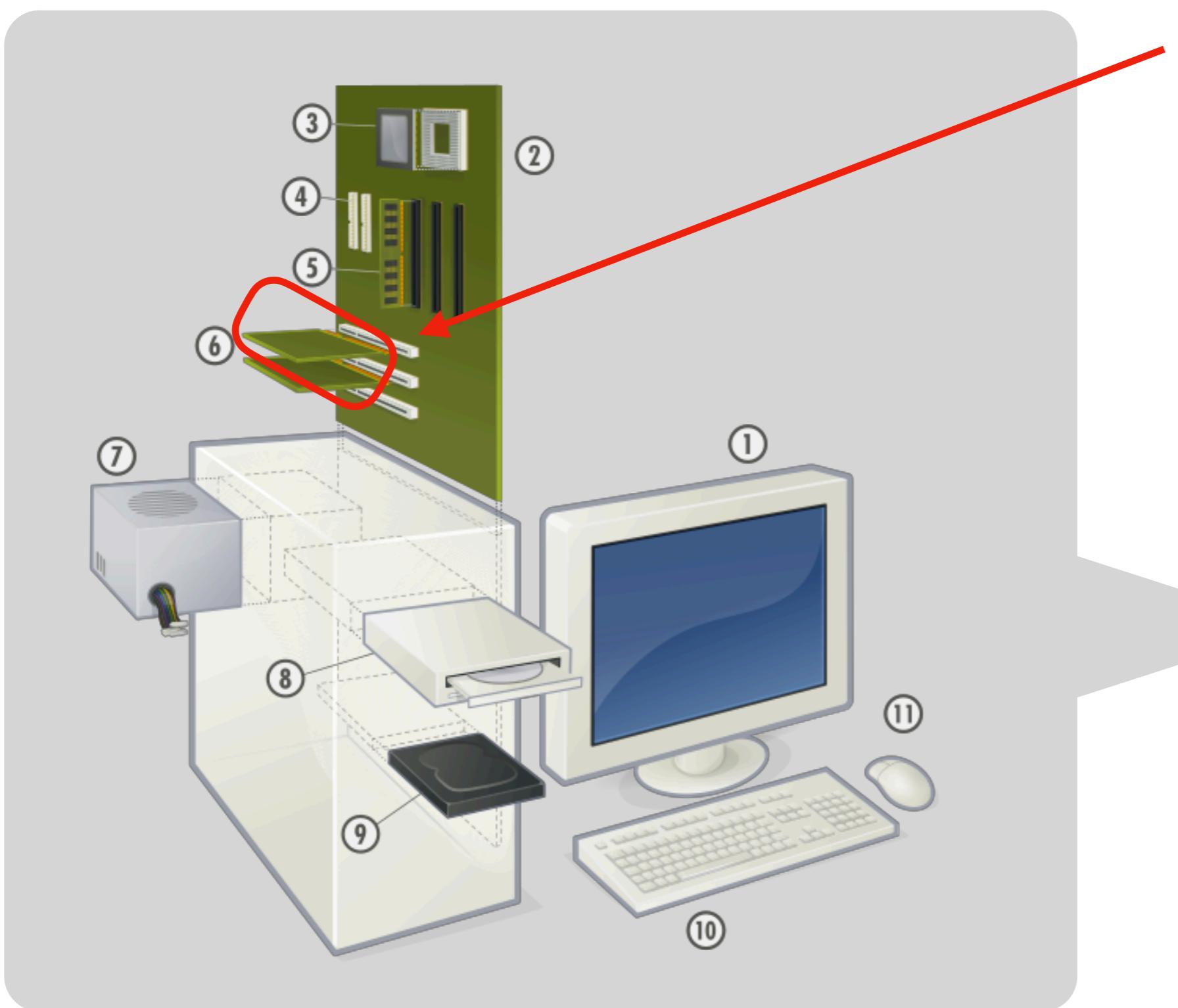
- Input/Output
- CPU
- Memory
- Storage
- Networking

Website

# Network Interfaces



# Network Interfaces



**Network:** often based on extension card or built into the motherboard itself



# Networking

## NIC (Network Interface Controller)

- Provides computer communication to other computers, and the Internet



## Wired vs. Wireless

- Wired ethernet is common for cable-based connection
- Wi-Fi is common for radio-based wireless connection



## Terminology

- **Server:** program/computer that runs, waiting for incoming requests, to which it responds
- **Client:** program/computer that sends requests to a server

# Today's Topics

Introductions

Course overview

Computer hardware basics

**Website**

# Course Website

There are three lecture sections for 301 this spring. I'm teaching sections 2 and 3 and Caroline Hardin is teaching 1.

Shared website:

<https://tyler.caraza-harter.com/cs301/spring19/home.html>

Walk through...

# Conclusion: five action steps for you

- take the "**Who are You?**" survey:  
<https://tyler.caraza-harter.com/cs301/spring19/surveys.html>
- read **syllabus** carefully:  
<https://tyler.caraza-harter.com/cs301/spring19/syllabus.html>
- sign into **Piazza**
- setup **Python** on your computer:  
<https://tyler.caraza-harter.com/cs301/spring19/videos.html>
- start **Project 1** (due next Wed):  
<https://tyler.caraza-harter.com/cs301/spring19/projects.html>