When do you feel the most like a computer scientist?

clicker.cs.illinois.edu

Q1

~Code~

340

for extra credit

CS 340

Debugging + Terminal

From craft night last night

Updates

1. MP 0 is out
   a. MP 1 is out

Feb 3rd

2. HW 0 is out today

next wed

3. Collaboration time today!

2-4 pm

Agenda

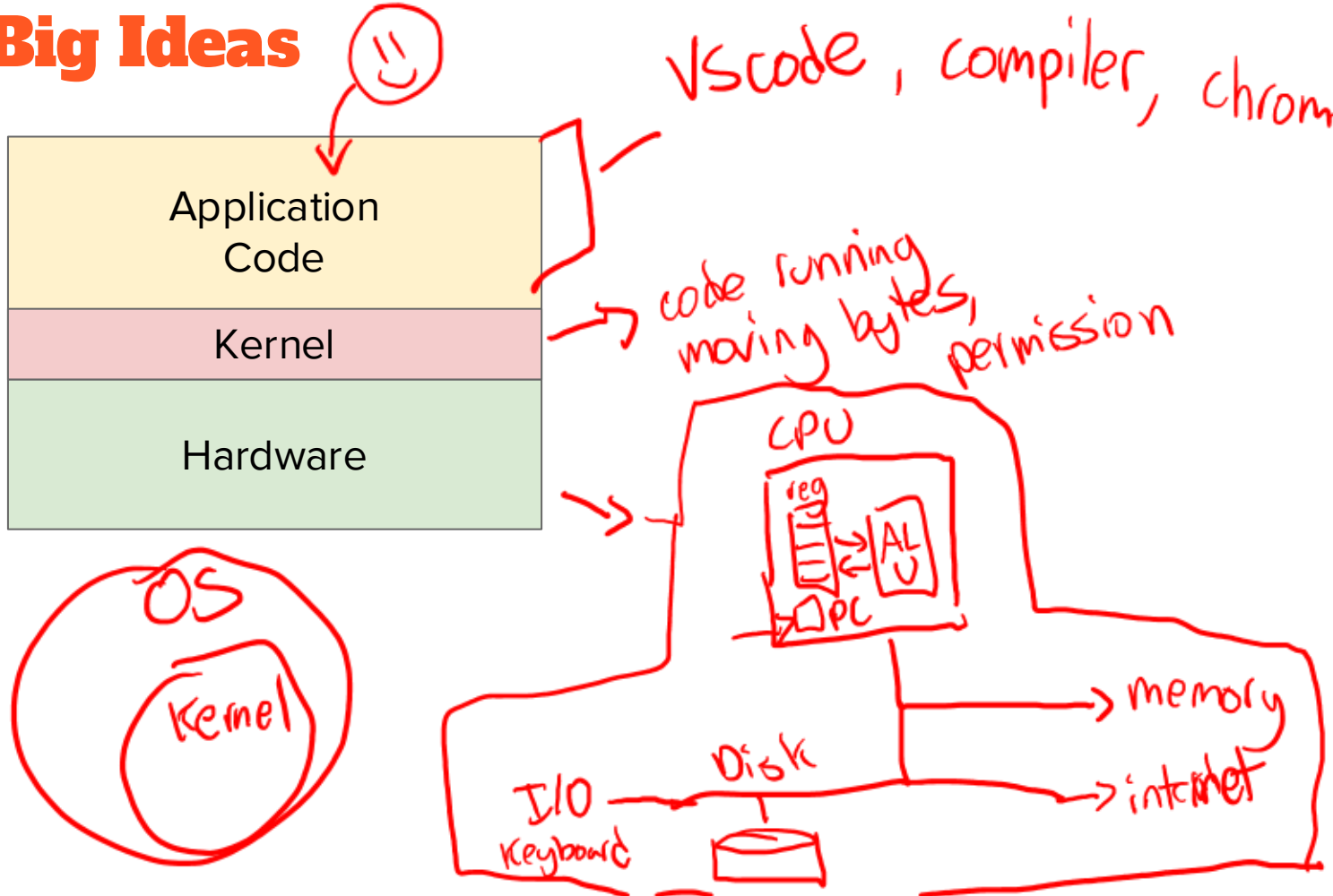
1. Review
2. Terminal/Shell Overview
    a. Basic commands
3. VScode Debugger
4. SSH and your VM

next week
C

gate

mp 0

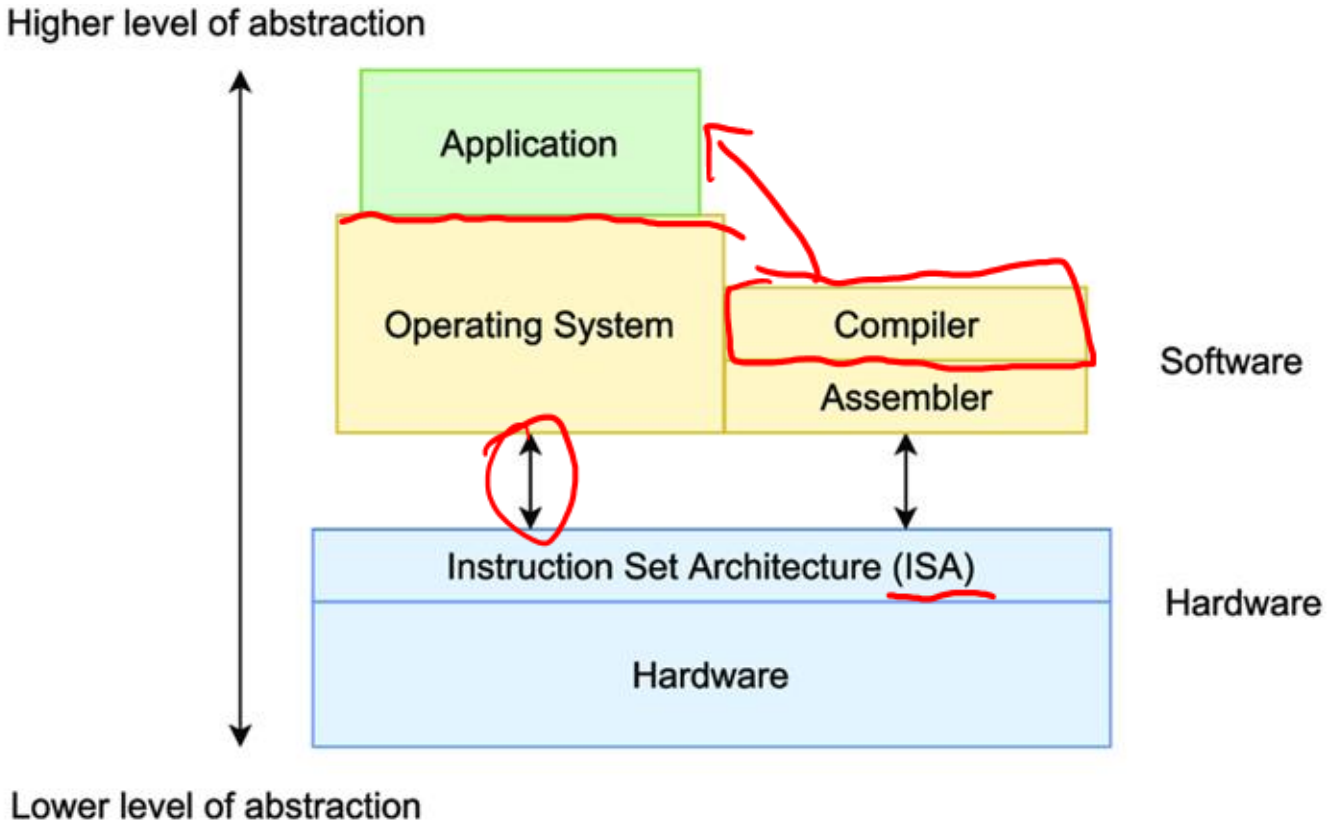
Big Ideas


Application Code
Kernel
Hardware
vscode, compiler, chrome, shell
code running
moving bytes,
permission
CPU
reg
ALU
PC
OS
Kernel
memory
Disk
I/O
keyboard
internet

What should you remember?

- There is a lot you don't YET know and a lot of details you may never care to know. Abstraction allows us to often ignore aspects we don't need to dive into.
- All code is run on hardware.
- The following terms exist - Compiler, application, CPU, PC, ALU, OS, memory, disk, I/O.
- The OS has a major role
  - Switches between running programs
  - Interfaces with many parts of the hardware like memory and other devices

What is misleading about this image?



Terminal/Shell + Debugging

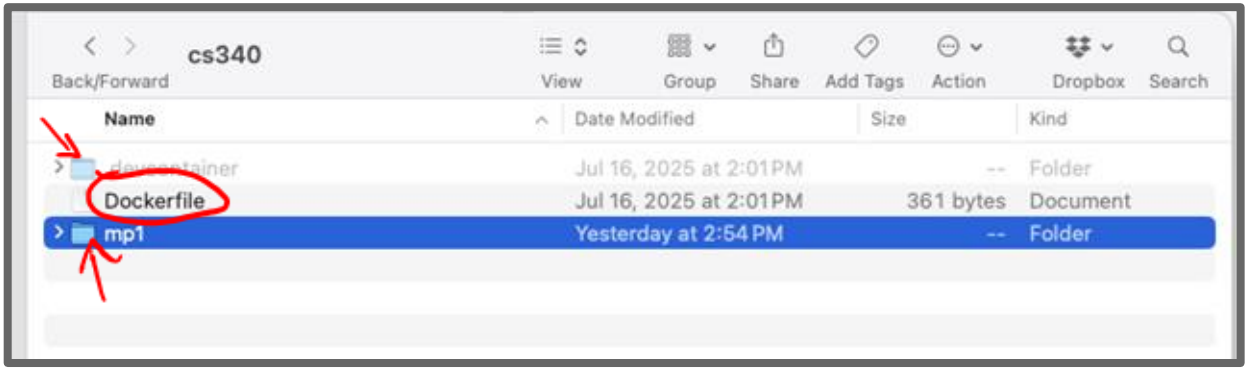
LG: To help you better understand how to run and debug coding projects.

Mental Models Developed

- How a file system works on a computer
- Compile/Execute flow for compiled languages
- What a debugger workflow looks like

Interacting with a computer

option 1 - click things



Option 2

terminal - UI that lets you type

Shell - program that runs the commands you type

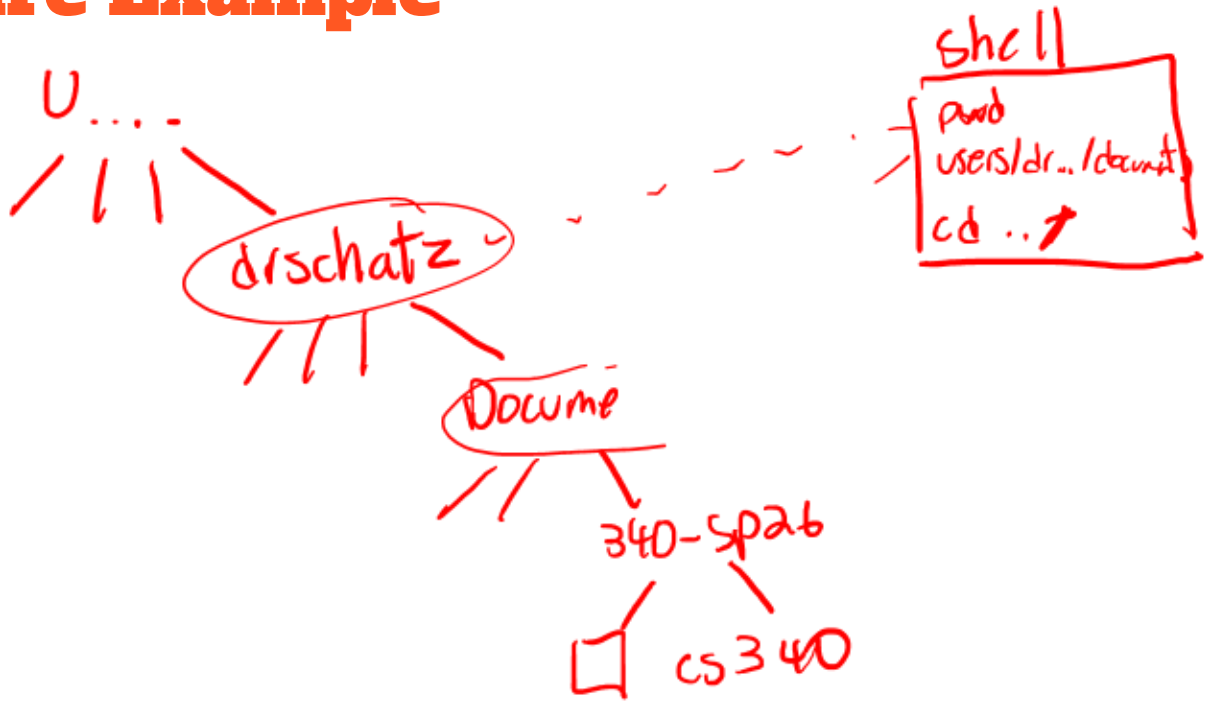
Basic Commands

ls - lists all files and directories in your working directory

cd - moves to a dif. directory

pwd - what file path you are in

File Structure Example



What would pwd print on the next line?

clicker.cs.illinois.edu

Q2

~Code~
340

```
drschatz@cs-drschatz-MBP 340-Sp26 % pwd
/Users/drschatz/Documents/340-Sp26
drschatz@cs-drschatz-MBP 340-Sp26 % cd ..
```

/Users/drschatz

Pwd

...?

What goes in the box to run the python code?

clicker.cs.illinois.edu

Q3

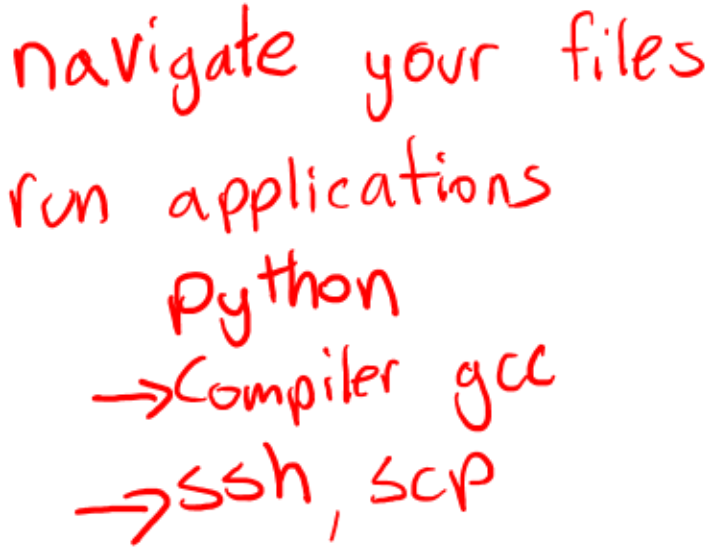
~Code~
340

```
drschatz@cs-drschatz-MBP term-demo % ls
Makefile        test_c.c        test_python.py
test_c          test_c.h        use_test_c.c
drschatz@cs-drschatz-MBP term-demo % pwd
/Users/drschatz/Documents/340-Sp26/cs340/term-demo
drschatz@cs-drschatz-MBP term-demo %
drschatz@cs-drschatz-MBP term-demo % cd ../
drschatz@cs-drschatz-MBP cs340 % ls
Dockerfile      mp1             term-demo
drschatz@cs-drschatz-MBP cs340 % cd mp1
drschatz@cs-drschatz-MBP mp1 % python3 [redacted] test_python.py
```

../term-demo/

Big Idea

You can use a terminal running a shell to -

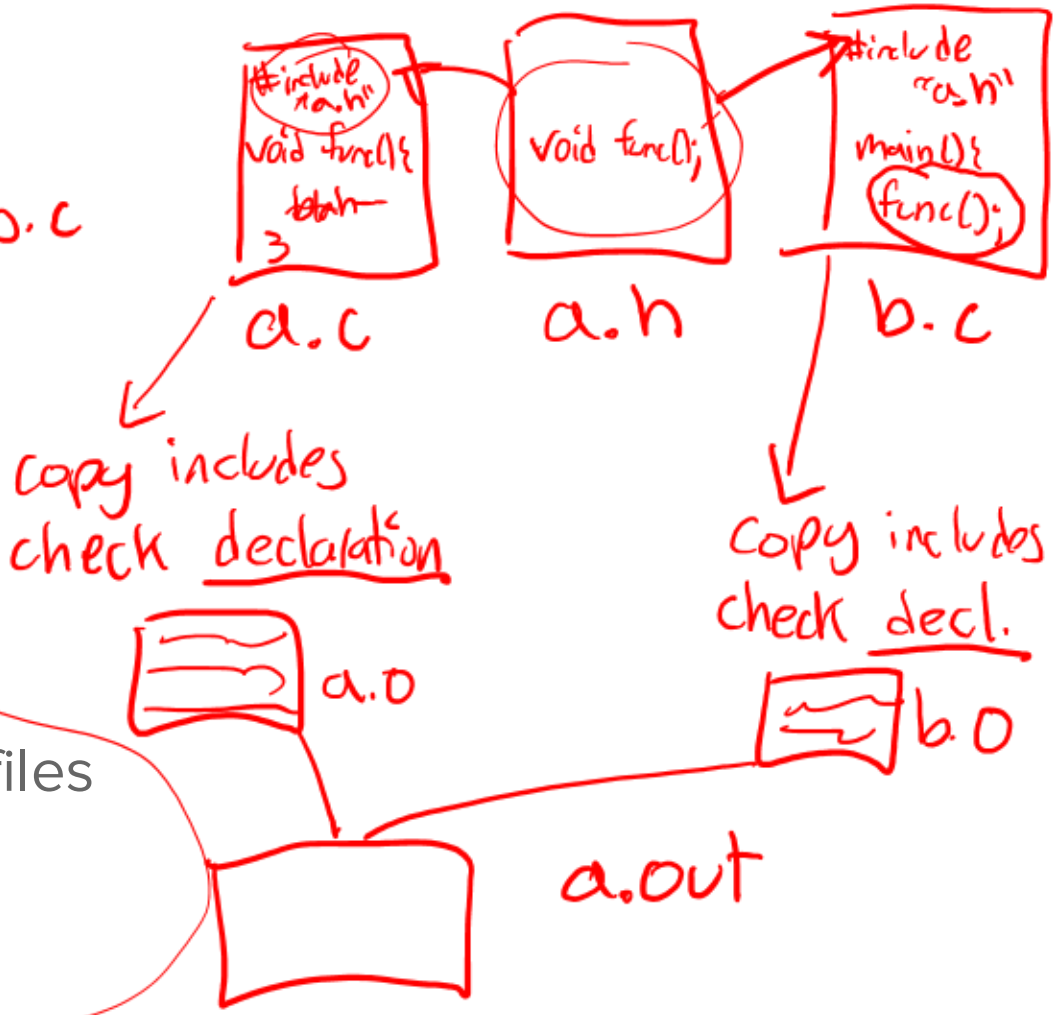


Running a C/C++ file

1. Compile → executable
2. Run executable file — run on CPU please

Compiling Details

1. Compile gcc a.c b.c
   a. Each .c file listed
   b. Link together .o files
      - main()
      - every func has def.



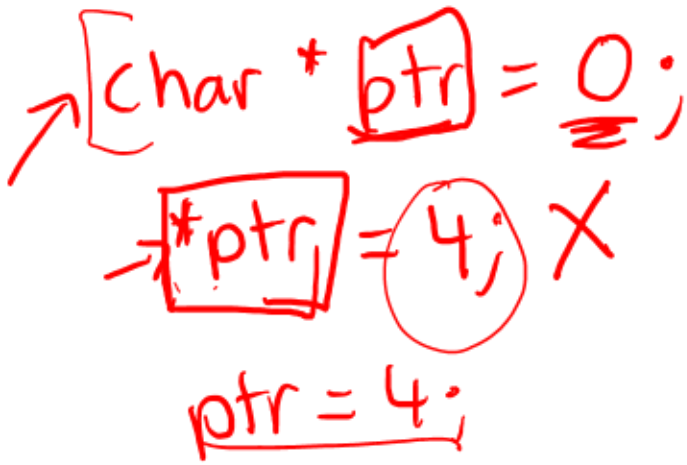
After I update the code, I need to recompile the file into an executable to see the change.

clicker.cs.illinois.edu

Q4

~Code~
340

A) True
B) False
C) Unsure


char * ptr = 0;
*ptr = 4; X
ptr = 4;


ptr
4
char *
0x6780
0 = 4

Docker Vs. VM

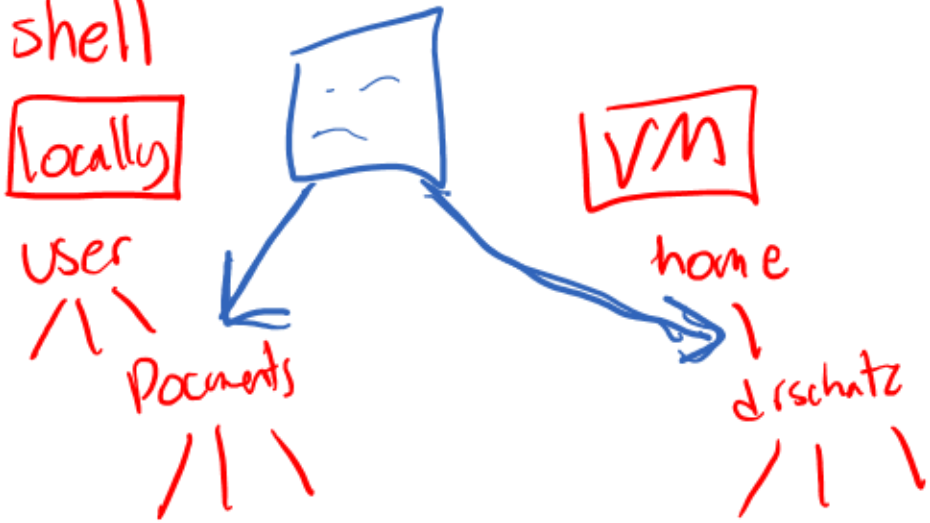
↘ creates an environment
- compiler
- applications

VM - virtual machine
dif computer

- dif files
- dif kernel

SSH and SCP → move files

→ secure shell



1. I make a file test.c on my docker environment
2. I SCP test.c to my VM
3. I edit test.c on my docker environment

True or false, test.c is now updated in both the VM and docker environment?

clicker.cs.illinois.edu

Q5

~Code~
340

Terminal/Shell + Debugging

LG: To help you better understand how to run and debug coding projects.

Mental Models Developed

- How a file system works on a computer
- Compile/Execute flow for compiled languages
- What a debugger workflow looks like