

CSE 374 - Intermediate Programming Concepts and Tools

Lecture 01 - Linux + Administivia
Summer 2026
Instructor: Amber Hu (they/them)

What is this course about?

Official learning objectives:

Linux and automation

Testing and debugging

Resource management

Who are we?

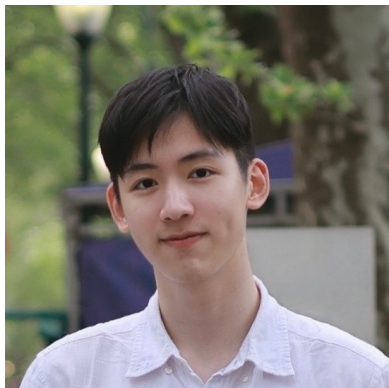
Instructor: Amber Hu (you may just address me as “Amber”!)

- Email: amberhu@cs.washington.edu
- **they/them** pronouns
 - “They sent me an email”
- I did my undergrad at UW, currently finishing up my PhD from Michigan State
- My office is in CSE/Allen 212



Who are we?

TAs:



Read more about us on
the Course Staff page!

How do I succeed in this course?

Lots of material, but straightforward to succeed

- Show up in lecture
- Try the assignments earlier, rather than later
- Ask questions often

Assessments ramp up:

frequent, low stakes → infrequent, high stakes

Little setbacks early are better than big setbacks later

Instructor and TAs are committed to helping every student succeed

Course Requirements & Grading

Lecture: Monday, Wednesday, and Friday: 10:50 am - 11:50 am in [PCAR 395](#)

- Will be recorded via Panopto (uploaded by end of day)
- Slides, code, and ink from lecture will be posted on the course website

Polls: During class, you will respond to poll questions through Gradescope assignments.

participation

Course Tools & Resources

Course website: Schedule, homeworks, slides, and lecture code

Ed discussion board: Get help and help others + some announcements

Gradescope: Assignment submission - exercises and homeworks, lecture polls

Canvas: Panopto lecture recordings, pre-course survey, official gradebook

GitLab: Introduced in Week 3, repository used to store your code for HW3 onward

Office Hours (OH):

- You **must** attend at least one OH in the first 2 weeks
- **1:1 meetings** available by appointment (for occasional support, not regular tutoring)

Poll: Have you read the syllabus, yet?

Please answer in the Poll: Lec 1 assignment on Gradescope!

- Yes I have!
- No – but I will review it after lecture!

Learning Outside of Lecture

There are a lot of *little* things to learn

Our goal in lecture is to teach you the big things

- Programming concepts, important tools, new languages

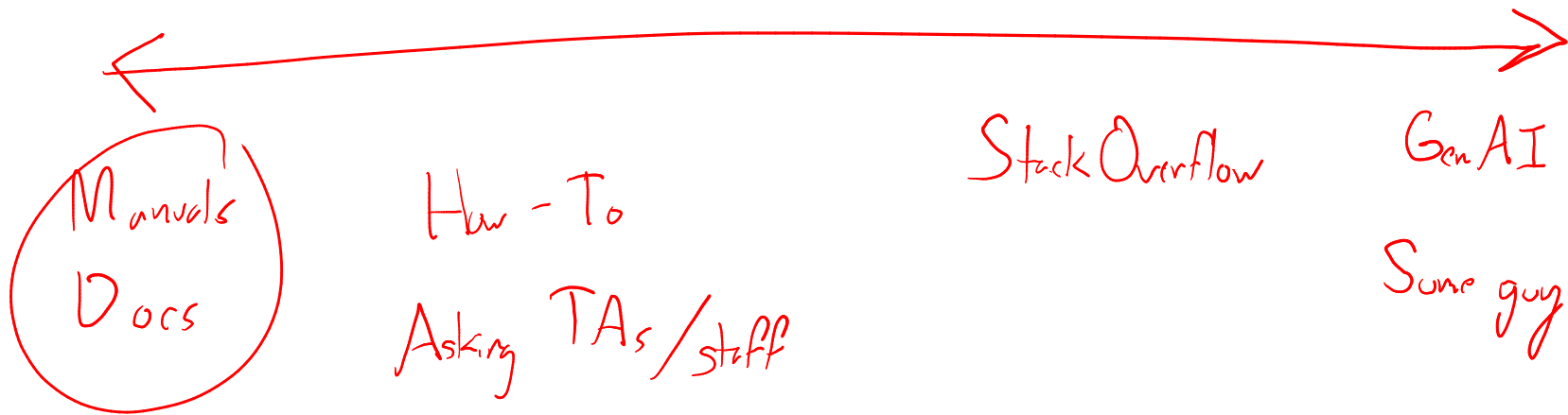
You will need to learn some of the little things on your own

- e.g. How do I create/unpack a .tar archive?
- e.g. How do I read lines of a text file in Bash?

This is similar to what you will experience in your future job!

- Learning how to learn is an *extremely* important skill!

Canon vs. Folklore



Generative AI Policies

ChatGPT and other LLMs are incredibly useful tools, but they can hinder your learning.

It's impractical for us to ask you to stay away entirely.

- Many software engineers and programmers use them daily.

Be mindful of how much you lean on it to learn how to code.

- It's ok to ask questions and learn from it (*i.e.*, use it like web search).
- It's NOT ok to ask it to implement your homework.

You may NOT rely on ChatGPT (or other GenAI tools) to complete assignments!

Academic Integrity

Policy in the course syllabus

- Also refer to: <https://cs.washington.edu/academics/misconduct>

Do your own work, be ready to explain it (Code Interviews!)

Unless otherwise specified, **all work in this course is independent**

- Do not share code; but you can discuss approaches on a high level

When in doubt - ask and be honest

We may run cheating detection software on your code

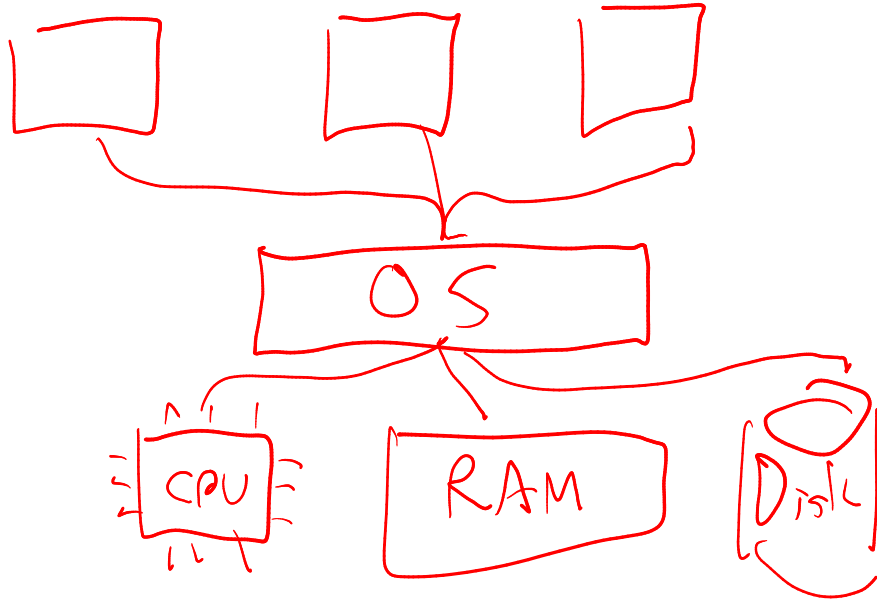
Tour of Course Website:

cs.washington.edu/374

Linux

What is the operating system (OS)?

The OS manages: *Files* *Commands*



Poll: Which operating systems have you used?

Please answer in the [Poll: Lec 1 assignment](#) on Gradescope!

- Desktop Linux
- Windows
- MacOS
- iOS
- Android (Linux)
- Others?

What is Linux?

Linux is an operating system like Windows or MacOS.

We will interact with Linux via a command-line interface, called the **shell**

- The shell is a **text-based** interface to the computer
 - We will use the “**Bash**” shell
- Windows and MacOS also have text-based shells

Linux can also use a graphical user interface (GUI)

- Text interface is sometimes harder to use, but can be more efficient
- Allows for *task automation* in ways that GUI cannot

What is a Command?

A Linux command typically consists of a program name followed by options and arguments, typed within a shell, like this:

program_name [arg1] [arg2] [arg3] ...

Example: wc -l myfile

- Program name (`wc`, short for “word count”) refers to a program somewhere on disk that the shell will locate and run.
- Options (or *flags*), which usually begin with a dash, affect the behavior of the program. In the preceding command, the `-l` option tells `wc` to count lines and not words.
- Argument `myfile` specifies the file that `wc` should read and process.

Options and Arguments

Depends on the command - can have multiple options and arguments.

Options may be given individually or combined after a single dash:

`wc -l -w myfile` is the same as `wc -lw myfile`

Multiple arguments are also OK (depending on the command).

`wc -l myfile myfile2 count` lines in two files

Shell Prompts

Before you can type a command, you must wait for the shell to display a special symbol, called a **prompt**.

A prompt = “I am waiting for your next command”.

Your prompt on your own machine might be a dollar sign \$ or percent sign %

or a complex string containing your computer name C:\Users\chloe>

Once you are on our Linux Machine, you should see [UWNetId@calgary ~]\$

Using Linux in CSE 374

SSH = Securely log into a remote host, or run commands on it.

`ssh` encrypts all data that travels across its connection, including your username and password (which you'll need to access the remote machine).

- **Secure Shell**

`Calgary` run by CSE department

- Connect using an SSH client
- (Optional) Remote SSH through VS Code:

<https://courses.cs.washington.edu/courses/cse374/25su/homeworks/hw0/setup/#optional-vscode-setup>

Use Calgary

In general, don't depend only on running code on your laptop (outside SSH)

- It can sometimes work, but generally Linux is different

What if I have my own Unix setup?

- We're grading you on Calgary, so your code **MUST** work there.
- No “It works on my machine” excuses!

HW0 Demo:

Logging into Calgary + Linux

What now?

Confirm that you are enrolled in Canvas, Gradescope, and Ed and can access Calgary

Try to log into Calgary using your UW NetID credentials

1. Open a terminal: Command Prompt or Powershell in Windows, Terminal in Mac
2. Connect to Calgary Linux server with:

ssh YourNetIDHere@calgary.cs.washington.edu

3. If it says something about authenticity, just say yes
4. Enter your UW NetID password when prompted.

The cursor will not move, but it is accepting input!

Assignment Reminders

EX0: Course Policies due before Friday's lecture

- Exercises are normally due before the next lecture starts, but giving some extra time for EX0 as people are settling into class!

HW0: Shell Access due Friday 6/26 @11:59pm

- Log into Calgary and experiment with the shell!

Pre-Course Survey due on Sunday 6/28 @11:59pm

- We (the staff) want to get to know you better and how we can best support you!

Make sure to submit ~~LP1~~ ^{Poll} before the next class as well.

All of these assignments can be found on our course's Gradescope page.

Extra Notes: