

Welcome to 107A



I hope we have enough room for all of you

Today

- **What is 107A?**
- Course Policies
- Am I enrolled in 107A? Will I be?
- Who am I?
- (if time) Some Unix Content

What is CS107A?

- A class that meets twice a week in which we will do:
 - A small warmup activity (usually incorporating an icebreaker question)
 - A short lecture on the important topics in 107 that week
 - Practice problems either as a class or in small groups

What is CS107A?

- A class that meets twice a week in which we will do:
 - A small warmup activity (usually incorporating an icebreaker question)
 - A short lecture on the important topics in 107 that week
 - Practice problems either as a class or in small groups
- An extra resource for 107 content
 - In addition to 107 Helper Hours, I hold 107A-specific office hours that are only open to you in which we can go over 107 assignments, practice problems, exams, etc
 - All 107 materials including slides and in class exercises + solutions will be posted on the 107A website
 - I will try to record the lecture portion of class, but no guarantees

What is CS107A?

- A class that meets twice a week in which we will do:
 - A small warmup activity (usually incorporating an icebreaker question)
 - A short lecture on the important topics in 107 that week
 - Practice problems either as a class or in small groups
- An extra resource for 107 content
 - In addition to 107 Helper Hours, I hold 107A specific office hours that are only open to you in which we can go over 107 assignments, practice problems, exams, etc
 - All 107 materials including slides and in class exercises + solutions will be posted on the 107A website
 - I will try to record the lecture portion of class, but no guarantees
- A community!
 - 107 is a lot of work, but it's a lot more work when you do it alone

Today

~~— What is 107A?~~

- **Course Policies**
- Am I enrolled in 107A? Will I be?
- Who am I?
- (if time) Short Exercise / Get to know each other

Course Policies

- Attendance is mandatory at **both** sections every week!
- You have to attend all but 2 mandatory sections (mandatory = starting next week)
- You have to pass 107 in order to pass 107A
- If you have to miss class for any reason, let me know within 2 days and we can work something out
 - If you miss more than 2, I will also have to keep ACE leadership in the loop (more on that later)
- That's it!

Today

—~~What is 107A?~~

—~~Course Policies~~

- Am I enrolled in 107A? Will I be?
- Who am I?
- (if time) Some Unix Content

Am I enrolled in 107A? Will I be?

- No one is enrolled yet - you need a permission number from me
- If you previously received an email saying you were pre-approved, you will receive your permission via email by the end of the week
- If you weren't pre-approved, I am still reviewing late applications (we are accepting applications through Friday), and I will accept as many of you as space permits
- If you are accepted in the second batch, you will get your permission number by next Tuesday

web.stanford.edu/class/cs107a/

Today

~~—What is 107A?~~

~~—Course Policies~~

~~—Am I enrolled in 107A? Will I be?~~

- Who am I?

- (if time) Some Unix Content

I'm Frankie, the 107A instructor



- CS undergrad (systems) + CS coterm (security)

I'm Frankie, the 107A instructor



- CS undergrad (systems) + CS coterm (security)
- I love to teach!
 - CS106B section leader for 3 years
 - CS106L lecturer last year

I'm Frankie, the 107A instructor



- CS undergrad (systems) + CS coterm (security)
- I love to teach!
 - CS106B section leader for 3 years
 - CS106L lecturer last year
- I'm from Minnesota

I'm Frankie, the 107A instructor



- CS undergrad (systems) + CS coterm (security)
- I love to teach!
 - CS106B section leader for 3 years
 - CS106L lecturer last year
- I'm from Minnesota
- I love hiking, canoeing, making dinner and eating the leftovers for lunch the next day and *in class participation*

Today

- ~~What is 107A?~~
- ~~Course Policies~~
- ~~Am I enrolled in 107A? Will I be?~~
- ~~Who am I?~~
- (if time) Some Unix Content

Today Part 2 (Unix)

- The command line
- Stanford's Myth machines
- Access the myth machines through ssh
- (if time) Talk about UNIX
 - Learn about file paths
 - Learn some useful commands you will use over and over!

What is the command line?

- The way you use your computer now is really nice!
 - You have a mouse, a pretty desktop, little folder icons representing where your files are
- The command line is a text-based way to use your computer
 - Which is worse in most cases, but better for running code!

```
~ -zsh ***  
→ ~ cd Desktop  
→ Desktop ls  
2021-09-20_AL_Frankie_Cerkvenik.pdf  
2022-09-27_AL_Frankie_Cerkvenik.pdf  
AppIdeasBebbsite  
Frankie_Cerkvenik_Block_Party - Block Party CIIAA - Google Docs.pdf  
Frankie_Cerkvenik_Block_Party - Employee Offer Letter.pdf  
FrankieCerkvenikResume2022.pdf  
JobApps  
Screen Shot 2022-06-22 at 11.27.07 AM.png  
Screen Shot 2022-10-13 at 5.29.07 PM.png  
Screen Shot 2022-10-22 at 12.30.29 PM.png  
→ Desktop █
```



Quick Demo - Terminal app on Mac

What are the Myth machines?

- Computers in Gates basement! Room B08
- That you have access to :)
- And we will be using in this class to run all of our code

The Myth machines ->



ty Andrew Benson's slides for the pic

UNIX and Linux

- UNIX and Linux are both operating systems - programs that run on every computer to define its basic functionality (Macs run MacOS!)
- The Myth machines use **Linux**, which was inspired by UNIX, so you might hear those terms interchangeably
- Operating systems define a set of commands that we can use from the command line (like `cd` and `ls`)
- In addition to using these commands, you will be implementing several of them throughout this quarter! **You** will become the OS-architect

How do I connect to the myth machines? `ssh`

- You don't have to go all the way to Gates to use these computers
- You can connect to them via the internet using a program called `ssh` ("Secure Shell") that we run from the command line
- To connect to your "account" on the myth machines, run

```
ssh <your_SUNetID>@myth.stanford.edu
```

- It should ask for you password, enter it (the cursor won't move, just keep typing)

Quick Demo - Myth

The two most important UNIX commands + one more

- **ls [PATH]**
 - Means “list” - it will list out all the files and folders in the specified directory
 - You can run just **ls** and it will list all the files in the current directory
 - Or run **ls PATH** and it will list all the files in the directory given by **PATH**
- **cd PATH**
 - Means “change directory”,
 - It will move you from your current directory into the one given by **PATH**
- **mkdir [PATH/]name**
 - Means “make directory”
 - Will make a folder called name in the path given by **PATH**, or in the current directory if none is given

File Paths

- When we say “**directory**”, think folder - its a file with other files in it!
- A **file path** specifies an exact route to a specific file or directory
 - `~/Desktop/example_file.txt` says that in the `~` directory, there is a folder called Desktop, which has a file called `example_file.txt` in it
- File paths can be **relative** or **absolute**
- Absolute paths start with special directory names, either `~` or `/`
 - `~` is the home directory, where you usually start in myth, `/` is the root directory
- Relative paths specify paths within the current directory you are in
 - `Desktop/example_file.txt` is a relative path. If we are currently in `~`, it would refer to `~/Desktop/example_file.txt`. If we were currently in `~/Documents`, it would refer to `~/Documents/Desktop/example_file.txt`
- Special relative paths: `.` means “current directory” and `..` means “parent directory”

Demo - draw Frankie's file tree

Demo - make your 107A folder

Exercise:

```
cp -r /afs/ir/class/cs107a/WWW/lecture_code/wk1_1 .
```

Exercise - groups of 5ish

1. Go around in a circle and introduce yourselves and say where you live. For each member of your group, create a textfile called THEIR_NAME.txt in the appropriate subdirectory of the `campus` directory

Example: if Frankie lives in McFarland, create

```
CS107A/wk1_1/campus/east_campus/frankie.txt
```

2. Challenge: if two people live in the same dorm, create a subdirectory on the appropriate place titled `dorm_name` and move their textfiles into that directory.

Example: if Frankie and Jerry live in Wilbur, move

```
CS107A/wk1_1/campus/east_campus/frankie.txt and
```

```
CS107A/wk1_1/campus/east_campus/jerry.txt to
```

```
CS107A/wk1_1/campus/east_campus/wilbur
```

Command-line quick-reference

- **ls [PATH]**
 - Means “list” - it will list out all the files and folders in the specified directory
 - You can run just **ls** and it will list all the files in the current directory
 - Or run **ls PATH** and it will list all the files in the directory given by **PATH**
- **cd PATH**
 - Means “change directory”,
 - It will move you from your current directory into the one given by **PATH**
- **mkdir [PATH/]name**
 - Means “make directory”
 - Will make a folder called name in the path given by **PATH**, or in the current directory if none is given
- **touch [PATH/]name**
 - Will make a file called name in the path given by **PATH**, or in the current directory if none is given
- **cp [-r] PATH_SRC PATH_DEST**
 - Means “copy”
 - Will copy the file specified by **PATH_SRC**, or everything in the directory specified by **PATH_SRC** if -r to the folder specified by **PATH_DEST**