```
$ cat welcome.c
#include <stdio.h>

int main(int argc, char *argv[])
{
   printf(“Welcome to COS 217\n");
   printf("Introduction to Programming Systems\n\n");

   printf("%s %d\n", ”Fall", 2020);
   return 0;
}

$ cat Makefile
CC=gcc217
welcome: welcome.o

$ make
gcc217    -c -o welcome.o welcome.c
gcc217   welcome.o   -o welcome

$ ./welcome
Welcome to COS 217
Introduction to Programming Systems

Fall 2020
```

## Agenda

**Course overview**

- **Introductions**
- Course goals
- Resources
- Grading
- Policies

**Getting started with armlab**

- Brief overview of Linux and bash
- bash walkthrough (separate video)

## Introductions

### Lead Instructor

| | |
|---|---|
| Christopher Moretti | cmoretti@cs.princeton.edu |

### Lead Preceptor

| | |
|---|---|
| Xiaoyan Li | xiaoyan@cs.princeton.edu |

### Preceptors

| | |
|---|---|
| Donna Gabai | dgabai@cs.princeton.edu |
| Scott Karlin | scott@cs.princeton.edu |
| Weicong Dong | weicongd@princeton.edu |
| Juan Duque | duque@princeton.edu |
| Ben Kaiser | bkaiser@cs.princeton.edu |
| Anne Kohlbrenner | akohlbrenner@princeton.edu |
| Dale Lee | dalelee@cs.princeton.edu |
| Huihan Li | huihanl@princeton.edu |
| Pi Songkuntham | pisong@princeton.edu |

## Agenda

Course overview

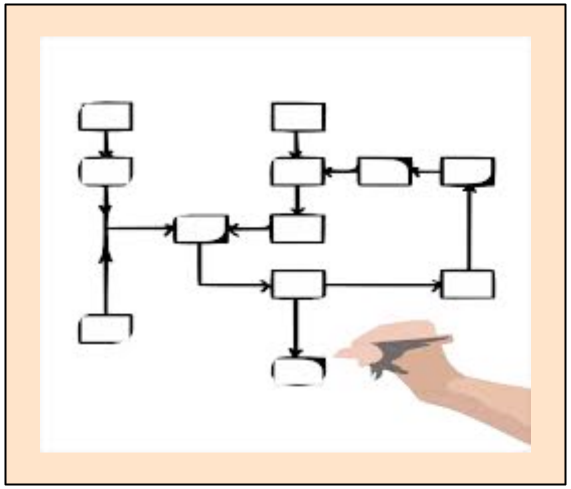
- Introductions
- **Course goals**
- Resources
- Grading
- Policies

Getting started with armlab

- Brief overview of Linux and bash
- bash walkthrough (separate video)

## Goal 1: Programming in the Large

Learn how to compose
large(r) computer programs



Topics

- Modularity/abstraction, information hiding, resource management, error handling, testing, debugging, performance improvement, tool support

# Goal 2: Lower-level Languages

```
int main(void) {
   while ((iChar = getchar()) != EOF) {
      lCharCount++;
      if (isspace(iChar)) {
         if (iInWord) {
            lWordCount++;
            iInWord = FALSE;
         }
      }
```

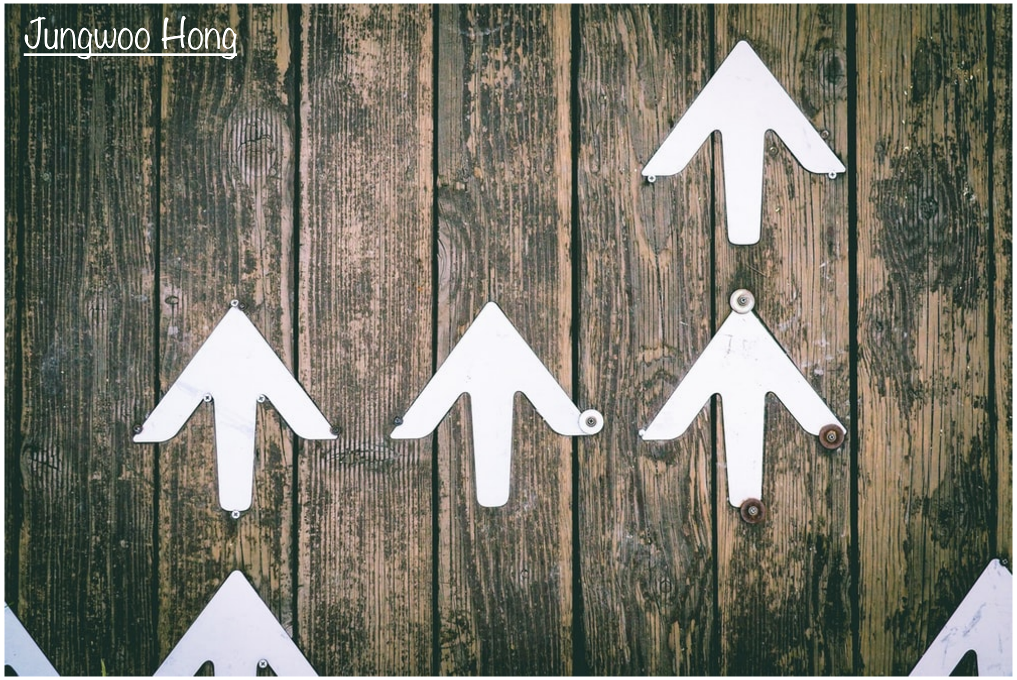
THE C PROGRAMMING LANGUAGE

```
main:
.LFB0:
.cfi_startproc
stp x29, x30, [sp, -16]!
.cfi_def_cfa_offset 16
.cfi_offset 29, -16
.cfi_offset 30, -8
add x29, sp, 0
.cfi_def_cfa_register 29
b .L2
```

```
RELOCATION RECORDS FOR [.eh_frame]:
OFFSET           TYPE              VALUE
000000000000001c R_AARCH64_PREL32  .text

Contents of section .text:
 0000 fd7bbfa9 fd030091 39000014
00000090  .{......9.......
```

## Goals: Summary

Help you to gain ...



***Programming Maturity***

## Specific Goal: Learn C

**Question**: Why C instead of Java?

**Answer 1**: A primary language for "under the hood" programming in real code bases.

**Answer 2**: A variety of experience helps you "program in the large"

THE C PROGRAMMING LANGUAGE

# Specific Goal: Learn Linux

**Question**: Why use the Linux operating system?

**Answer 1**: Linux is the industry standard for servers, embedded devices, education, and research

**Answer 2**: Linux (with GNU tools) is good for programming (which helps explain answer 1)

## Agenda

Course overview

- Introductions
- Course goals
- **Resources**
- Grading
- Policies

Getting started with armlab

- Brief overview of Linux and bash
- bash Walkthrough (separate video)

## Lectures

### Lectures

- Describe material at a mix of levels
    - Some conceptual (high) overview
    - Some digging into details
- Slides available via course website
- Videos released TTh on Canvas
- “Watch Party” on Zoom TTh 10am-11am.

### Etiquette

- Watch the lecture before going to precept, otherwise you may end up lost in precept and slow down the rest of the class
- “Watch Party” office hours are for questions about lecture content topics only, not about help with assignments.

## Precepts

### Precepts

- Describe material at the “practical” (low) level
- Support your work on assignments
- Hard copy handouts distributed during precepts
- Handouts available via course website

### Etiquette

- Attend your precept: attendance will be taken
  - Must miss your precept? ⇒ inform preceptors & attend another
- Use TigerHub to move to another precept
  - Best for this to happen organically (more than 25% move ≥ 1x)
  - Issues ⇒ See Colleen Kenny (info on website)

**Precepts begin Wednesday/Thursday!**

## Websites

https://www.cs.princeton.edu/~cos217 (Course website)

- Home page, schedule page, assignment page, policies page

https://princeton.instructure.com/courses/561 (Canvas)

- Links to Zoom precepts, Ed, recorded lectures and precepts, Library reserves and other readings, NameCoach

# Ed

## Ed

- https://us.edstem.org/courses/2185/discussion/
- Also available as a Canvas link
- Instructions provided in first precept

## Etiquette

- Study provided material before posting question
  - Lecture slides, precept handouts, required readings
- Read / search all (recent) Ed threads before posting question
- Don't reveal your code!
  - See course policies

# Books

***C Programming: A Modern Approach (Second Edition)*** (required)

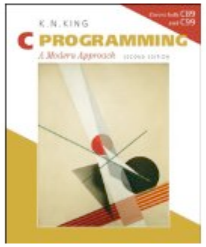
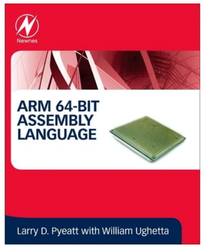
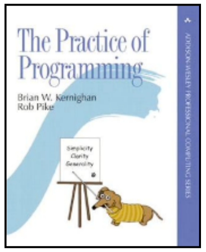
- King
- C programming language and standard libraries

***ARM 64-bit Assembly Language*** (required)

- Pyeatt with Ughetta

***The Practice of Programming*** (recommended)

- Kernighan & Pike
- "Programming in the large"

## Manuals

Manuals (for reference only, available online)

- ***ARMv8 Instruction Set Overview***
- ***ARM Architecture Reference Manual***
- ***Using as, the GNU Assembler***

See also

- Linux ***man*** command

# Help sessions

## Office Hours (starting Wednesday 9/2)

- 4+ hours every weekday + 2 hours Sunday
- Schedule is on the course website
- Links are on Ed

## LabTAs

- Your peers are available 4-6 hours per day, every single day
- These are specific to debugging your assignments, for conceptual help with course materials, go to office hours
- https://labta.cs.princeton.edu/

# Agenda

Course overview

- Introductions
- Course goals
- Resources
- **Grading**
- Policies

Getting started with armlab

- Brief overview of Linux and bash
- bash Walkthrough (separate video)

## Grading

| Course Component | Percentage of Grade |
|---|---|
| Assignments * | 66 |
| Midterm Exam ** | 10 |
| Final Exam ** | 20 |
| Participation *** | 4 |

* 6 assignments * 11% each; penalties for lateness

** During midterms week and final exam period, respectively

*** Did your involvement benefit the course?

- As measured through precept attendance, precept participation, and Ed participation
- Scaled down from prior terms due to being online

# Programming Assignments

## Regular (every 1.5-2.5 weeks) assignments

0. Introductory survey
1. “De-comment” program
2. String module
3. Symbol table module
4. Directory and file trees *
5. Assembly language programs *
6. Buffer overrun attack *

*(partnered assignment)

**Assignments 0 and 1 are available now**

**Start early!!**

## Agenda

Course overview

- Introductions
- Course goals
- Resources
- Grading
- **Policies**

Getting started with armlab

- Brief overview of Linux and bash
- bash Walkthrough (separate video)

## Policies

Learning is a collaborative activity!

- Discussions with others that help you understand concepts from class are encouraged

But programming assignments are graded!

- Everything that gets submitted for a grade must be exclusively your own work
- Don't look at code from someone else, the web, Github, etc. – **see the course "Policies" web page**
- Don't reveal your code or design decisions to anyone except course staff – **see the course "Policies" web page**

Violations of course policies

- Typical course-level penalty is **0**
- Typical University-level penalty is **suspension**

Questions?

## Agenda

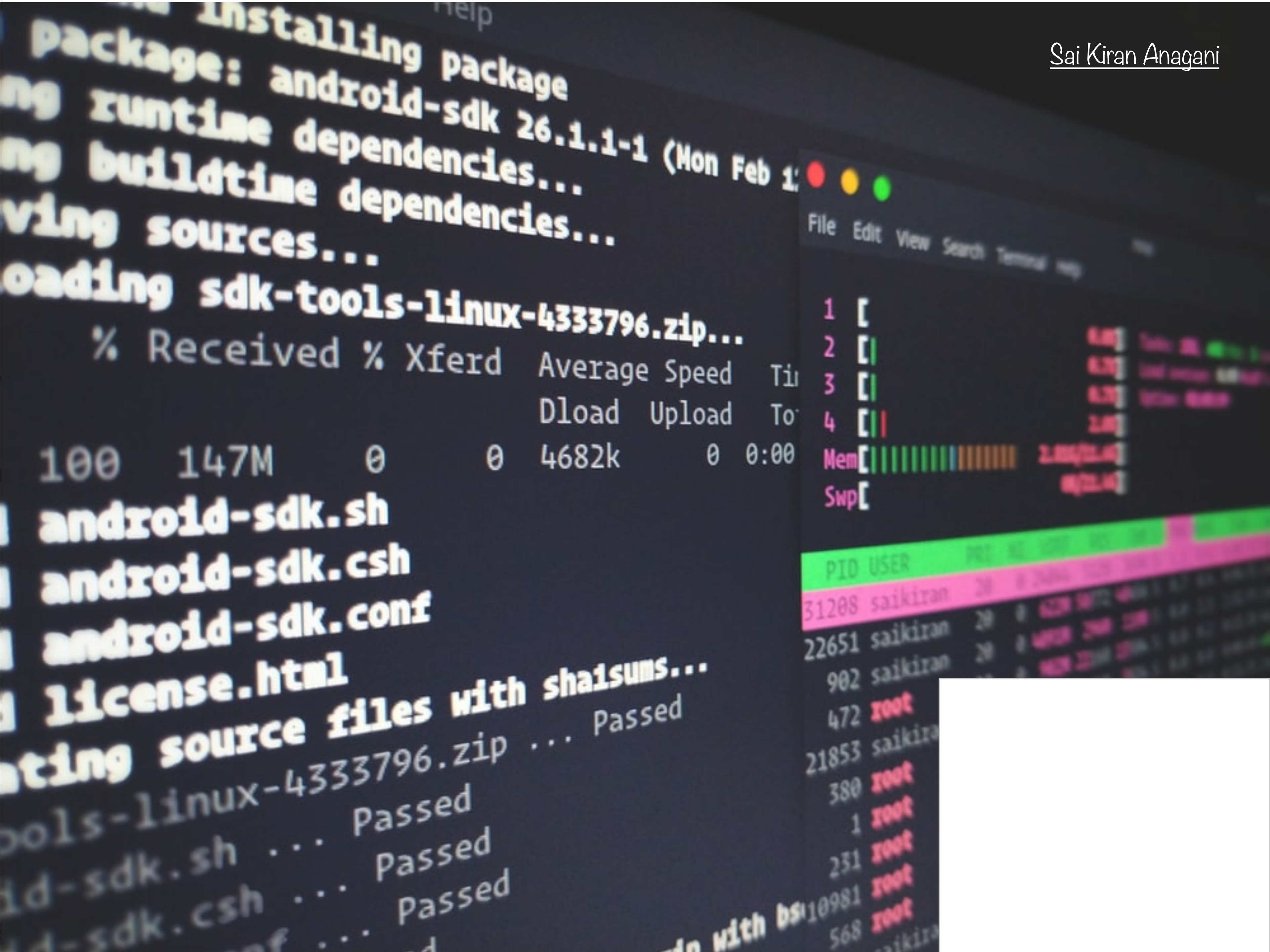
Course overview

- Introductions
- Course goals
- Resources
- Grading
- Policies

Getting started with armlab

- **Brief overview of Linux and bash**
- bash Walkthrough (separate video)

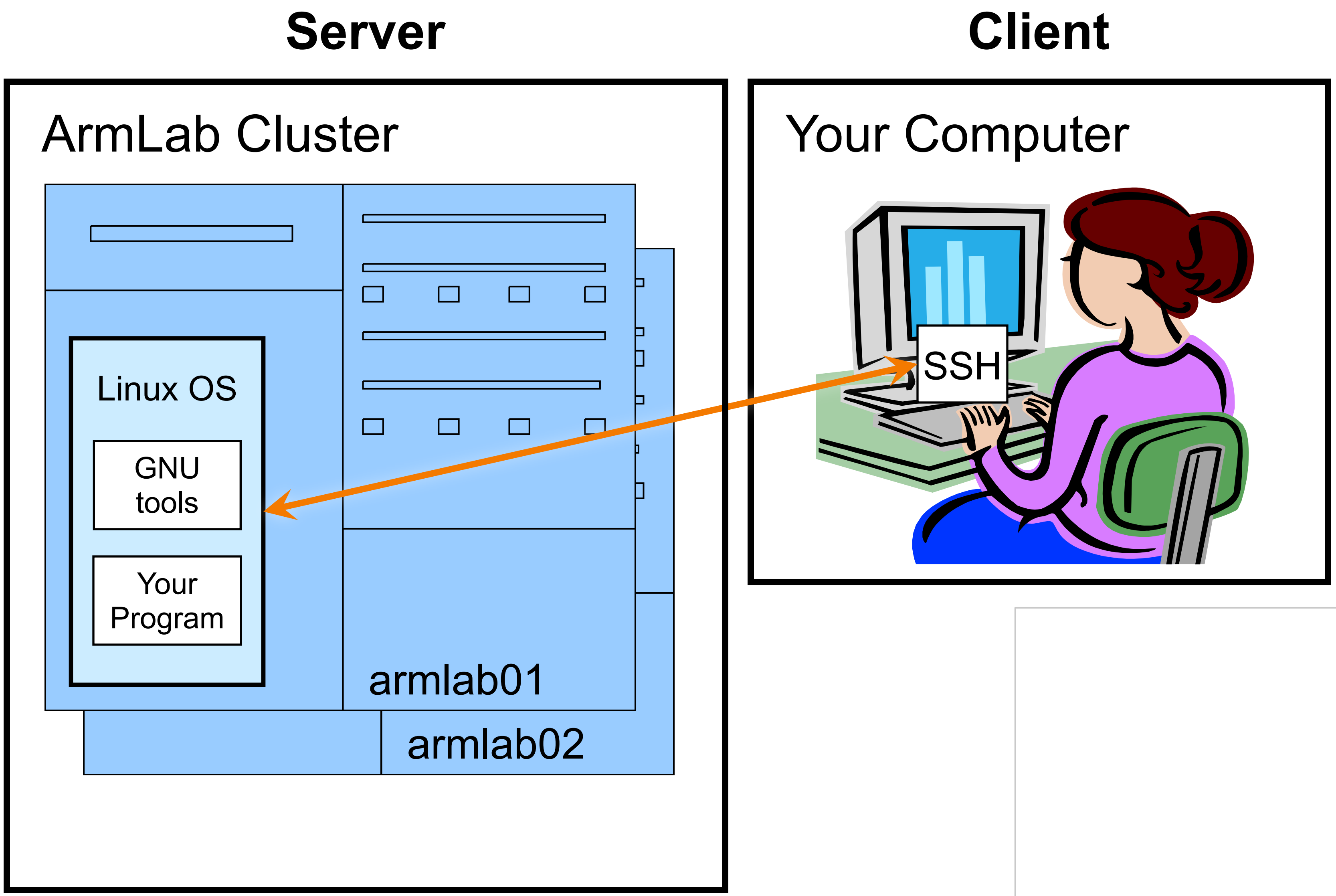
Sai Kiran Anagani



```
    Installing package
Package: android-sdk 26.1.1-1 (Mon Feb 1
ng runtime dependencies...
ng buildtime dependencies...
ving sources...
oading sdk-tools-linux-4333796.zip...
  % Received % Xferd  Average Speed   Ti
                      Dload  Upload   To
100  147M    0     0  4682k      0  0:00
 android-sdk.sh
 android-sdk.csh
 android-sdk.conf
 license.html
ting source files with sha1sums...
ools-linux-4333796.zip ... Passed
id-sdk.sh ... Passed
id-sdk.csh ... Passed
```

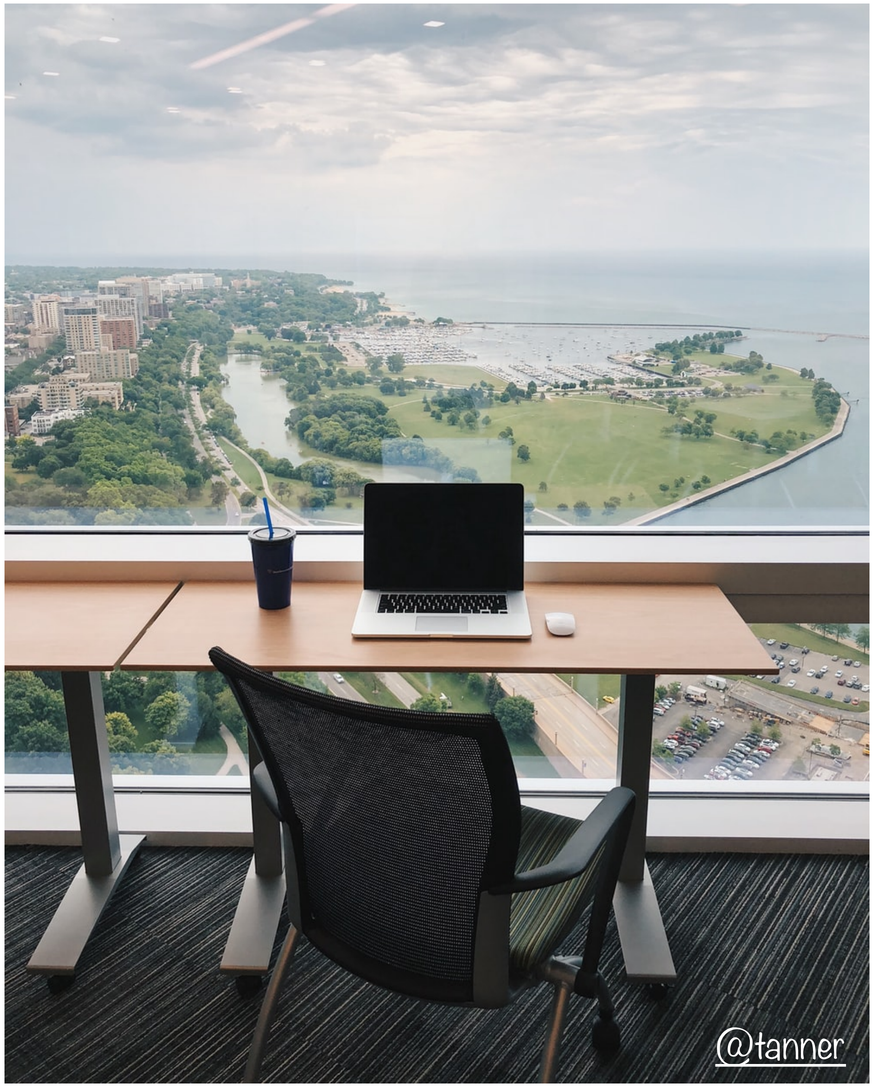
```
  1 [
  2 [|
  3 [|
  4 [||
Mem[||||||||||||||||||||
Swp[

  PID USER
31208 saikiran
22651 saikiran
  902 saikiran
  472 root
21853 saikira
  380 root
    1 root
  231 root
10981 root
  568 root
```

# Programming Environment



## Terminology: Terminal vs Shell

## Client/Server Implication



You can do this course from anywhere in the world!

- Good in general, when compared with being confined to a cluster in the Friend basement.
- Necessary in these times

## Getting Started

Check out course website **soon**

- **Study “Policies” page**
- Assignments 0 and 1 are available

Establish a reasonable computing environment **soon**

- Instructions given in first precept
- Whatever you choose, you’ll need to get up to speed on Linux at least a little bit, so that will be the second part of this lecture.