

CS107e
Computer Systems from
the Ground Up

Julie Zelenski, Pat Hanrahan,
Jesse Doan, Matthew Trost,
Maria Fernandez,
Liana Keesing, Anna Mistelle

Winter 2022
<https://cs107e.github.io/>

Learning Goal 1

**Understand how computers
represent data,
execute programs,
and control peripherals**

Saturn V Kennedy Space Center



Falcon 9





Understanding is Empowering

Understand ...

ARM processor and memory architecture

Peripherals: GPIO, timers, UART, ...

Assembly language and machine code

Low-level representation of information / bits

From assembly language to C

Function calls and stack frames

Serial communication and strings

Modules and libraries: Building and linking

Memory management: Memory map & heap

Learning Goal 2

Master your tools

Software Tools

UNIX command line: bash, cd, ls, ...

Text editor: vim, emacs, sublime, ...

Programming languages: C, ...

Compiler: gcc

Assembler: as

Linker/loader: ld

binutils: nm, objcopy, objdump, ...

make

git and github.com

documentation: markdown

Software Tools

UNIX command line: bash, cd, ls, ...

Text editor: vim, emacs, sublime, ...

Programming languages: C, ...

Compiler: gcc

Assembler: as

Linker/loader: ld

binutils: nm, objcopy, objdump, ...

make

git and github.com

documentation: markdown



Organized Development Environment



<http://amhistory.si.edu/juliachild/>

A close-up photograph of a person's hands using a hand plane to smooth a piece of wood. The person is wearing a maroon apron. The hand plane is held in the right hand, and the left hand is guiding the wood. The wood is resting on a wooden workbench. In the background, there is a metal stand with a blue label that says "Verban Vre-AC" and a clear plastic water bottle.

Practice, Practice, Practice

Debugging and Troubleshooting





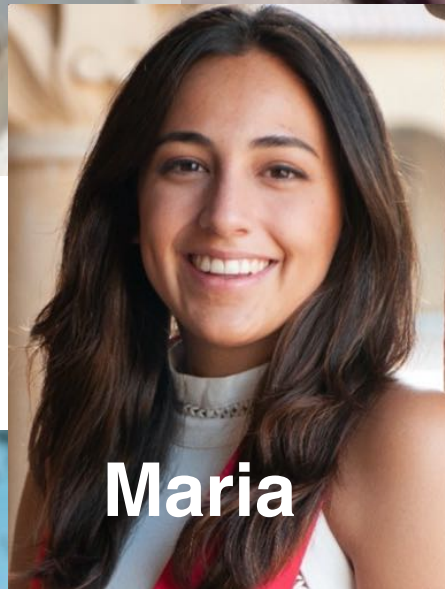
Pat



Julie



Jesse



Maria



Matt



Anna



Liana

Course Schedule

§1 Bare Metal Programming

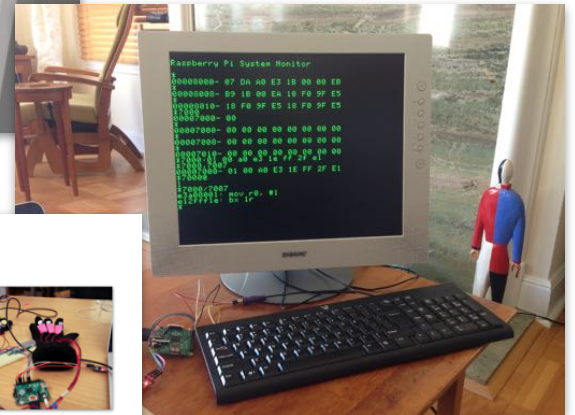
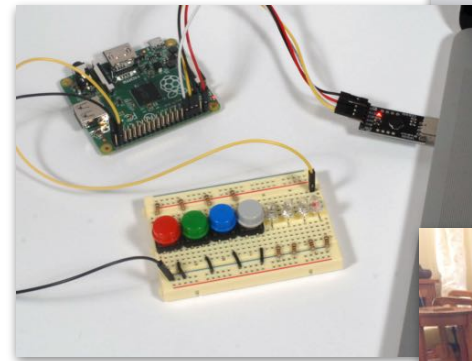
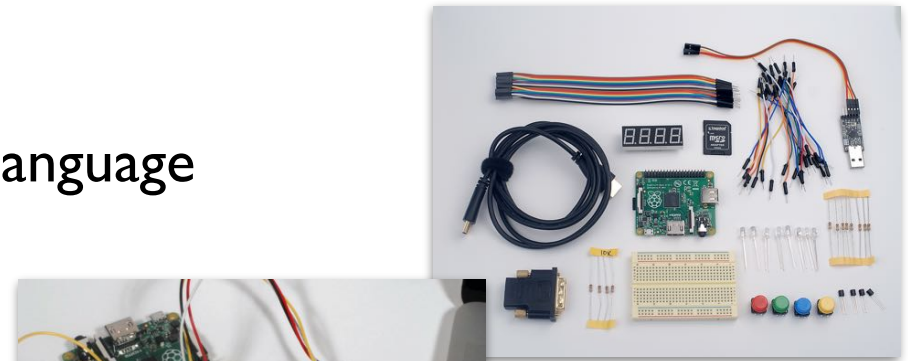
- ARM architecture and assembly language
- C functions and pointers
- Serial communication
- Linking and loading
- Memory allocation

§2 Build a Personal Computer

- Keyboard
- Graphics
- Interrupts

§3 Create Your Own Project

- Sensors
- Performance



<https://cs107e.github.io/schedule/>

Weekly Cadence

Each week has a focus topic

Pair of coordinated lectures on Fri and Mon

Lab sessions Tues or Wed evening

Assignment out Wed after lab, due next Tue at midnight

Labs

Set of guided exercises, follow up on lecture content

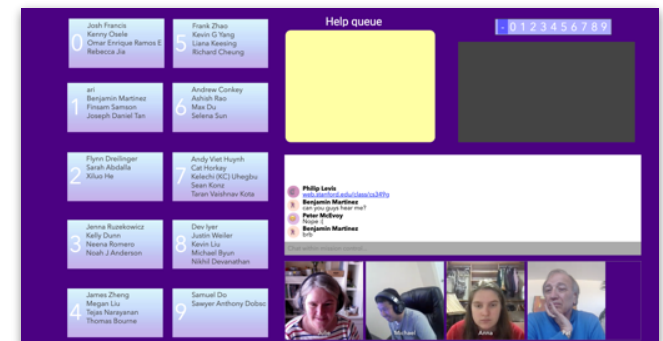
Work in groups

Complete exercises and check in with staff

Leave lab ready to start assignment!

Lab participation is **mandatory**

Philosophy: lab is hands-on, collaborative, supported, **fun!**



Assignments

7 weekly assignments that build on each other
This is where the learning really happens!

Each assignment has

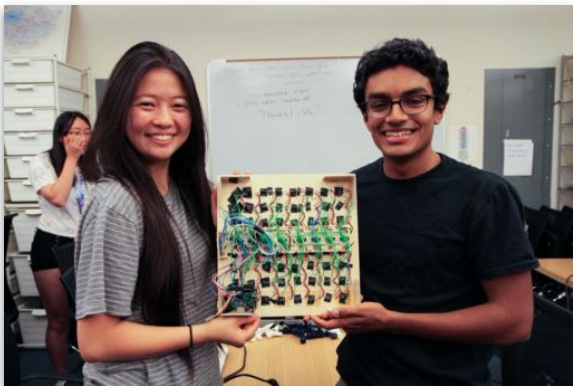
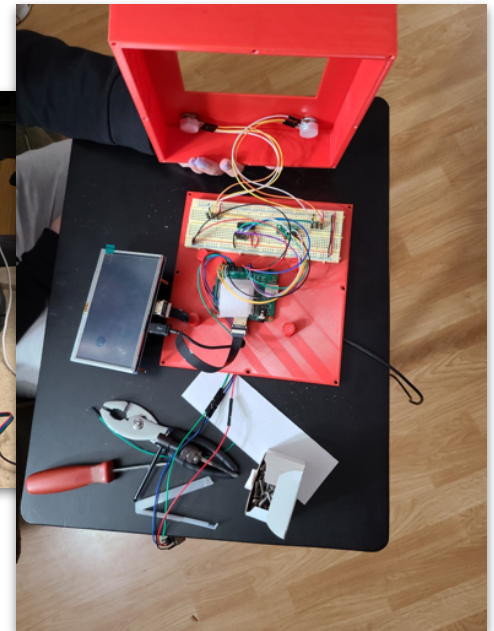
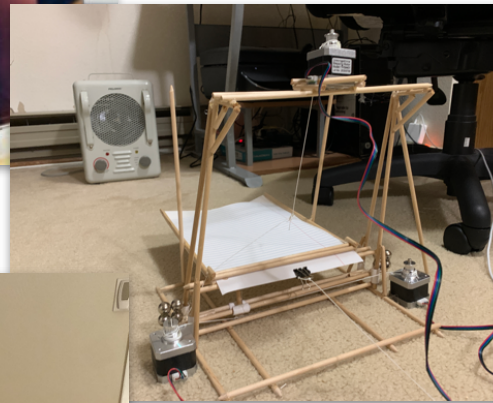
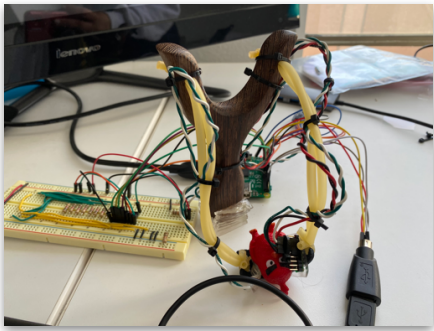
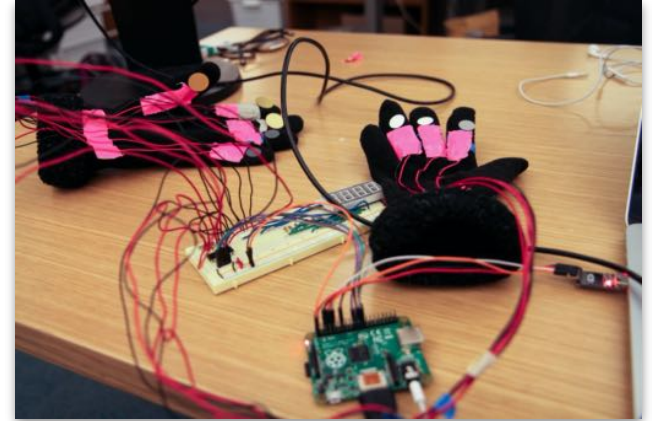
- **Basic** requirement (tight spec, guided steps)
- Optional **extension** (opportunity for exploration/creativity)

Encouraged to resubmit and correct issues in basic

End goal is complete working system of your own

Project!

project_fair.mp4



Markers for success

- Solid prerequisites: CS106B, C++, debugging
- Curiosity
- Perseverance
- Motivation

How to thrive in this course

- Consistency, follow through
- Leverage our resources, support, feedback
 - Ask questions, reach out when you need help

Interested?

FAQ cs107e.stanford.edu

Submit student questionnaire by Dec 28th

Our decisions by Dec 30th, you commit by Jan 2

Kits pickup on campus by Jan 11th