

A WEB-BASED SYSTEM PROGRAMMING LEARNING ENVIRONMENT

ECE UNDERGRADUATE RESEARCH SYMPOSIUM SPRING 2015

Neelabh Gupta
Computer Engineering

April 8, 2015



ADVISORS



- **Roy H. Campbell,**
Computer Science
<http://cs.illinois.edu/directory/profile/rhc>



- **Lawrence C. Angrave,**
Computer Science
<https://cs.illinois.edu/directory/profile/angrave>

SYSTEM PROGRAMMING

- An operating system is a piece of software to manage a computer's resources and provide support for common functions such as accessing peripherals
- System programming refers to writing code that takes advantage of operating system support for programmers to provide users and other programmers useful interfaces for accessing the hardware and other resources
- Traditionally performed and taught using the C language

TEACHING OF SYSTEM PROGRAMMING: COURSE MATERIAL

- Core part of most undergraduate computer engineering/science courses
- Teaching to beginners is primarily limited to University classrooms
- Wide gap between classroom learning and practical assignments. For example in CS 241 (System programming) course at Illinois, most students are using C for the first time. However, the first assignment gets started with relatively advanced concepts - students are expected to learn C and navigate “trivial” programming issues by themselves.
- Need C tutorials, short video lectures, bite-sized challenges/exercises

TEACHING OF SYSTEM PROGRAMMING: DEVELOPMENT ENVIRONMENTS

- Students are usually required to setup a complicated development environment
- Either compatible hardware and software required, or virtual machines used
- Setup can be daunting for learners
- Not viable for modern large scale online courses (such as MOOCs)
- Natural solution is to deliver the programming environment in the web browser
 - Simple
 - Scalable
 - Consistent

BROWSER-BASED PROGRAMMING ENVIRONMENTS

- Many in-browser programming and learning environments
- Almost all of them teach scripting/web languages
- Usually rely on a server infrastructure for compiling and running user programs
 - Hard to scale
 - Prone to security concerns
- Prior to this project, no viable implementation for system programming

THE PROJECT

- A C programming environment running in the browser
- Purely client code, no compilation or running in the server
- A full-fledged Linux virtual machine in JavaScript
- Short lecture videos
- Simple challenges and exercises
- Quick access to manuals and documentation
- Complements in-class learning

TRY IT OUT!

- Live: <http://cs-education.github.io/sys/>
- Open source: <https://github.com/cs-education/sysbuild>

System Programming

Lessons

Playground

Code

Man page search

monokai

12

Indent code

Download file

Settings

```
1- //Write your C code here!
2- #include <stdio.h>
3-
4- int main() {
5-     printf("Hello world!\n");
6-     return 0;
7- }
8-
```

gcc opts

-lm -Wall -fmax-errors=10 -Wextra

args

Program arguments

Run It

VM state: Running 0.2 MIPS

Compiler: Ready

Hello Standard Error Stream

- Spot the mistake(s) in the following code:

```
for(len = 0; len < 5 ; len++) {
    write(STDOUT_FILENO,"I think", 6);
    write(STDOUT_FILENO,"\n", 6);
}
```
- And use your corrected version in a complete C program to print the following seven lines.

```
I think
I thin
I thi
I th
I t
I
I
```
- Return the value 7. Verify that your program returns 7 typing `echo $?` in the terminal window after your program has just run.

```
NET: Registered protocol family 17
ata1.00: ATA-2: jorlk-disk, , max PIO2
ata1.00: 61440 sectors, multi 0: LBA
ata1.00: configured for PIO
scsi 0:0:0:0: Direct-Access      ATA          jorlk-disk      n/a   PQ: 0 ANSI: 5
sd 0:0:0:0: [sda] 61440 512-byte logical blocks: (31.4 MB/30.0 MiB)
sd 0:0:0:0: [sda] Write Protect is off
sd 0:0:0:0: [sda] Write cache: disabled, read cache: enabled, doesn't support DP
O or FUA
sda: unknown partition table
sd 0:0:0:0: [sda] Attached SCSI disk
VFS: Mounted root (ext2 filesystem) readonly on device 8:0.
devtmpfs: mounted
Freeing unused kernel memory: 128K (c03fa000 - c041a000)

Jan  1 00:00:02 login[43]: root login on 'ttyS0'
stty -clocal crtscts -ixoff
echo boot2ready-$?
Jan  1 00:00:02 login[44]: root login on 'ttyS1'
Aug Wednesday  1 00:00:00 UTC 2014
~ # stty -clocal crtscts -ixoff
~ # echo boot2ready-$?
boot2ready-0
~ #
```

Copyright © 2015 The Board of Trustees at the University of Illinois. Attribution and legal information

PROJECT TIMELINE

- Started out as an experiment in Summer 2014
- CS 241 Fall 2014 students start using it
- Took on as research project in Fall 2014
- Project becomes part of the Department of Computer Science's Senior project class
- Senior project team of 7 students start working on the project

RESULTS AND FUTURE

- Used by 400+ CS 241 students in Spring 2015 and Fall 2014 semesters
- To be built into a full fledged course for introduction to system programming
- Part of CS Senior project, so future senior project batch of students will continue growing the platform
- Envisioned to be the go-to platform for teaching system programming at Universities and online course
- Illinois as a leading brand for system programming instruction

THANK YOU!

neelabhgupta.com

 @neelabhg

Q & A