CS 354 - Machine Organization & Programming
Thursday, September 5, 2019

Waitlisted? Complete the form at: https://forms.gle/CRvL1oR8i9Bymvyo6

Instructor: Jim Skrentny, skrentny@cs.wisc.edu

Lectures

- Lecture 1: 132 Noland, TR: 2:30 PM - 3:45 PM
- Lecture 2: 132 Noland, TR: 4:00 PM - 5:15 PM

Description

An introduction to fundamental structures of computer systems and the C programming language with a focus on the low-level interrelationships and impacts on performance. Topics include the virtual address space and virtual memory, the heap and dynamic memory management, the memory hierarchy and caching, assembly language and the stack, communication and interrupts/signals, assemblers/linkers and compiling.

Today

Course Info and Coursework
Java vs. C
Coding in C Remotely
    Get Connected to CS
    Edit your Source
    Compile/Run/Debug

Next Time

C Program Structure and Control
Review:
    K&R Ch. 2: Types, Operators, and Expressions
           variable names, data types, constants, declarations
           arithmetic/relational/logical operators, assignment, precedence
    K&R Ch. 3: Control Flow
           statements & blocks, if-else & else-if, switch, while, for, do-while
    K&R Ch. 4: Functions & Program Structure
           basics, parameters, return values, scope rules
Do: read course "Information and Policies" pages linked to course website
    access CS Linux lab computers, try Linux commands and tools (vim, gcc)
    check out course Piazza site

Course Information

✳ ***Course Website:*** https://canvas.wisc.edu/courses/154937

Textbooks
- ◆ The C Programming Language, Kernighan & Ritchie, 2nd Ed., 1988
- ◆ Computer Systems: A Programmer's Perspective, Bryant & O'Hallaron, <u>2nd Ed, 2010</u>

Piazza
- ◆ is used for online course discussions and questions with classmates and the TAs about homeworks, projects, exams, and course concepts as well as course logistics

CS Account
- ◆ provides access to CS Linux Computers with C tools (rooms **1366**, 1355, 1358, 1368)
- ◆ should be same user name/password as your prior CS 200/300 accounts
- ◆ must be activate your CS account IF YOU ARE NEW TO CS (see TA during lab consulting)
- ◆ is needed to access your CS 354 student folder used for some course projects

TAs
- ◆ are graduate students with backgrounds in computer architecture and systems
- ◆ help with course concepts, Linux, C tools and language, homeworks and projects
- ◆ do consulting in 1366 CS Linux Computer Lab during scheduled hours, which are posted on course website's "TA Consulting" page

Peer Mentors
- ◆ are undergraduate students that have recently completed CS 354
- ◆ hold drop-in hours and do a variety of activities to help students succeed, which are posted on course website's "PM Activities" page

Coursework
The first page of each lecture outline will have reminders of coursework deadlines.

Exams (55%)
- ◆ Midterm (~15%): Thursday, October 3rd, 7:15 PM to 9:15 PM
- ◆ Midterm (~18%): Thursday, November 7th, 7:15 PM to 9:15 PM
- ◆ Final (22%): Monday, December 16th, 5:05 PM to 7:05 PM

Conflict with these times? Complete the form at: https://forms.gle/6TwXssFmUCh7o8GS8

Projects (30%): 6 projects, 3-8% each, posted on course website

Homeworks (15%): 10 homeworks, 1.5% each, posted on course website

Java vs. C

Java	C
designed for safety	
object oriented	
interpreted	
high level	
references	
garbage collector	
exception handling	

Coding in C Remotely - Get Connected to CS

✳ *The CS Linux lab computers are to be used for CS 354 programming.*

Access CS Linux Computers

Windows: get ssh program and configure to connect to CS machines
Macs: open terminal and enter ssh <cs_account>@<machine>

machine names:
    best-linux.cs.wisc.edu (might cause issues with security certificates on Macs)
    emperor-01.cs.wisc.edu through emperor-07.cs.wisc.edu
    rockhopper-01.cs.wisc.edu through rockhopper-09.cs.wisc.edu
    royal-01.cs.wisc.edu through royal-30.cs.wisc.edu
    snares-01.cs.wisc.edu through snares-10.cs.wisc.edu

Use some Linux

command shell

→ How do you:
    list the contents of a directory?   Show details?

    display what directory you're currently in?

    copy a file?

    remove a file?

    move to another directory? Up a directory?

    make a new directory?

    rename a file or directory?

    remove a directory?

    get more information about commands?

Coding in C Remotely - Create your Source

1. Edit your Source File

```
$vim prog1.c
$vimtutor
```

→ Why vim?

```c
/* title:  First C Program
 * file:   prog1.c
 * author: Jim Skrentny
 */


#include <stdio.h>
#include <stdlib.h>
#include <string.h>


int main()
{

   char *str = malloc(50);

   printf("Enter your CS login: ");

   if (fgets (str, 50, stdin) == NULL)
      fprintf(stderr, "Error reading user input.\n");


   int len = strlen(str);
   if (str[len – 1] == '\n') {
      str[len – 1] = '\0';
   }


   printf("Your login: %s\n", str);



}
```

2. Compile

```
$gcc prog1.c
```

OR
```
$gcc prog1.c –Wall –m32 –std=gnu99 –o prog1
```

3. Run

```
$a.out
```

→ Why a.out?

OR
```
$prog1
```

4. Debug

5. Submit

- ◆ Download your source from the lab computer to your local machine
- ◆ Upload your source from your local machine to the course website