**CSCI 3753: Operating Systems**

**Fall 2015**

**Instructor:** Dr. Frank W. Miller

**Office:** ECOT 734

**Email:** [frank.miller@colorado.edu](mailto:frank.miller@colorado.edu)

**Office Hours:** MWF 11:00a-1:00p (2 hours after lectures)

Available other times as needed, send me an email or stop by

**Teaching Assistants:** Rahat Rafiq ([rahat.rafiq@colorado.edu](mailto:rahat.rafiq@colorado.edu)), Zhiyuan Liu

([zhiyuan.liu@colorado.edu](mailto:zhiyuan.liu@colorado.edu))

**Class Time:** MWF 10:00a-10:50a

**Location:** ECCS 200

**Recitation Sections:** 101: M 12:00p-12:50p STAD 112

102: M 1:00p-1:50p ECCR 139

103: M 4:00p-4:50p ECCR 110

104: M 3:00p-3:50p ECCR 110

**Description:**

This course introduces students to important Operating Systems concepts. The course will cover key aspects of how an operating systems functions on a modern computer.  The following major topics will be covered:

* Basic System Structure
* Processes and Threads
* Virtual Memory
* File Systems
* Security
* Networking and Distributed (File) Systems
* Virtual Machines

In addition, the class will gain familiarity with important software tools such as debuggers, compilers, editors, kernel modules, and virtual machines.

**Prerequisites:** CSCI 2400

**Textbook:**

*Operating System Concepts, 9th Edition,* Abraham Silberschatz, Peter Galvin, Greg Gagne (electronic or hardcover)

**Suggested: http://www.intel.com/content/www/us/en/processors/architectures-software-developer-manuals.html**

**Course Outline:**

|  |  |  |
| --- | --- | --- |
| **Week** | **Topics** | **Reading** |
| **1** | Intro, Kernel Mode, System Calls | Ch. 1, 2 |
| **2** | I/O, Interrupts, DMA | Ch. 2, 13 |
| **3** | Processes, IPC, Synchronization | Ch. 3, 5 |
| **4** | Mutual Exclusion, Sychronization | Ch. 5 |
| **5** | Monitors, Cond Vars, Deadlock | Ch. 5, 7 |
| **6** | Deadlock Detection, Avoidance, Scheduling | Ch. 6, 7 |
| **7** | Scheduling | Ch. 7 |
| **8** | Scheduling, Memory Mgmt | Ch. 6, 8 |
| **9** | Paging | Ch. 8 |
| **10** | Paging, Page Replacement | Ch. 8, 9 |
| **11** | Memory Allocation, File Systems | Ch 9, 10-12 |
| **12** | File Allocation, Free Space Mgmt | Ch. 10-12 |
| **13** | Disk Scheduling, Flash File Systems, RAID | Ch. 10-12 |
| **14** | Authorization, Symmetric Key Crypto, Public Key Crypto | Ch. 14, 15 |
| **15** | Networking, Virtual Machines | Ch. 16, 17 |

**Course Website:**

Please enroll ASAP in the Moodle course web page. Nearly all your class interactions will be available through Moodle. The enrollment key is “StorageHierarchy”

**Grading and Policies:**

25% Final Exam  
15% Midterm  
50% Programming Assignments  
10% Problem Sets

**Grading for Programming Assignments:**

Grading for programming assignments (PAs) will be based on interview-style grading, where about 40% of the grade will be based on the code submitted (does it compile, does it execute the required functions) and about 60% of the grade will be based on answering questions from the TA in the interview (questions may be based on explaining the code and/or explaining software concepts that the PAs cover), unless otherwise noted.

*Teams*: Students may work in teams of up to two, but each student will still be responsible for scheduling their own grading meeting with the TA for each PA.

*Code*: Programming assignments must be submitted by [uploading them to the moodle class Web page](http://moodle.cs.colorado.edu/) by their due date, unless otherwise noted.  All PAs must be written in C and compiled for execution in the specified environment for that PA, unless otherwise noted.  The late policy for submitting PAs is as follows: maximum 30 points out of 40 on the code grading portion of the assignment if submitted within 24 hours of the due date.  If you submit your assignment over 24 hours after the due date, we reserve the right to assign you a zero for the code portion of the assignment.  *We highly encourage you to submit even partially completed programs for partial credit.* Extensions will not be granted except at the instructor's discretion in documented cases of excused absences (extreme hardship, emergencies, etc.).

*Interview*: Each student must arrange an individual grading interview with a TA for each PA, even if they are part of a team.  Interview time slots will be posted on the moodle.  You may bring your own laptop to these grading sessions to demonstrate your program to the TA and answer questions.  Even if your PA code is submitted in time, you must attend your grading meeting to be given a grade for the interview portion of the PA. Any missed meetings (without notifying your TA ahead of time with a suitable reason) may result in a zero grade for the interview portion of the assignment.  The TA is under no obligation to reschedule your appointment if you miss your meeting, so write down your meeting times, and don't forget them!  
  
All programming assignments should be your original work, unless otherwise noted.  You may help others only to the extent of answering typical questions that arise during compiling, debugging, and executing your programming assignments.  The code that you write should be your own (or your team's own).  Remember that *plagiarism only hurts you* and is not an excuse that Google accepts during your Google inteview when you can't explain fundamental OS concepts!

**Grading for Problem Sets:**

All problem sets must be submitted by the deadline.  No late submissions are allowed except in documented excused cases.