

CS5600 – Computer Systems (MS Core)

Semester: Fall 2017

Instructor: Peter Desnoyers, room 334 West Village H

Class: Wed 6:00-9:00 PM, 325 Behrakis

Office hours: Wed 10:00 AM – 12:00 noon, or by appointment, Rm. 334 West Village H

Textbook: “How Operating Systems Work”, Desnoyers 2017 (download from Blackboard)

Studies the internal operation of computer systems, using traditional operating systems as a model. Briefly reviews computer hardware and architecture. Covers memory organization, processes and context switching, concurrency and parallel programming, disks and external storage systems, file systems, memory management and virtual memory, security, and virtual machines. Exposes students to these concepts through both paper-based and programming exercises. Assumes prior knowledge of computer architecture and C programming.

Prerequisites

Students should be able to write programs in C in order to complete the programming assignments. Some knowledge of computer architecture is assumed.

Course Topics

This class covers the following areas:

- OS basics - context switching and program loading
- Virtual memory at the hardware and operating system level
- Hardware virtualization
- File and block I/O, DMA
- Basic OS security theory and mechanisms

Course Outcomes

By the end of this class students will be able to demonstrate mastery of the following topics:

- OS basics – trace the operation of context switching and program loading, complete a programming assignment demonstrating understanding of these principles.
- Synchronization – use synchronization primitives (mutexes, condition variables) to synchronize threads as demonstrated in a programming assignment.
- Virtual memory – describe operation of page tables and OS page faulting mechanisms for implementing demand allocation, demand loading, and copy-on-write.
- Block storage – understand performance characteristics of hard drives, demonstrate knowledge of RAID configurations via a programming assignment.
- File systems – describe operation of journaled and log-structured file systems, demonstrate knowledge by writing a simple FUSE-based file system for Linux.
- Security – understand permissions and access control lists as implemented in Linux and Windows, describe common software exploits.
- Virtual machines – understand and describe the three common virtualization mechanisms in use.

Format and Logistics

This class meets once a week in the evening; students are expected to attend lecture. There are reading assignments for each week; students are expected to do the assigned reading before lecture, as it is an

integral part of the course. Lectures will be recorded and videos made available within a day or two; these may be used for reference or in the case of an unavoidable absence. (note that my recording equipment has been less than reliable lately, so please plan accordingly).

There will be short homework assignments each week, due on the Monday after the corresponding material is covered in lecture. The homeworks will be made available before lecture, so that students may ask questions on them during class; in addition you are encouraged to discuss the assignments (and readings) on Piazza.

There will be four programming assignments, completed in teams of two, which will involve significant programming and testing effort.

Assignment submission and late policy

Weekly short homework assignments will be submitted through Blackboard; late assignments will not receive credit. Submitted work should be in PDF or text format.

Programming assignments will be distributed and submitted via the source-code control system. Each student will be given 4 slip days, which you can use on any project in increments of one day – e.g. you can hand in one project 4 days late, or all projects 1 day late. You do not need to ask for permission to use slip days – if you have remaining slip days your late assignment will automatically receive its graded mark, while if you have used them all your assignment will be assessed a penalty of 20% per day or fraction thereof.

Submission time will be automatically determined from the time of the last checkin to the corresponding homework repository. If you are done on time, but accidentally submit changes after the due date, please contact the instructor to fix things.

Slip days will be deducted from *each group member's remaining slip days*. This means that if you change groups and your partner does not have any slip days remaining, then you don't either. If this seems grossly unfair (e.g. your previous partner dropped the class, leaving you little choice in partners) please contact the instructor.

Online discussion

Although Blackboard is used for most of the online aspects of this course, Piazza (piazza.com) will be used for online discussion. You will receive an online invitation to join the CS5600 class, and will be responsible for responding to it. If you do not receive an invitation, please go to <http://piazza.com/northeastern/fall2017/cs5600desnoyers> to subscribe. **Participation in Piazza discussions on class material and assignments will be a significant factor in your class participation grade.**

Computing resources

You will need a CCIS account in order to complete the assignments for this class. If you are a CCIS student you should get an account automatically; otherwise go to <https://my.ccs.neu.edu/> and click “Apply for account” to get one.

Certain parts of the programming assignments are machine-specific. A standard virtual machine image is used to provide a consistent programming environment; instructions are in Homework 0. This is the environment which will be used for testing submissions for grading purposes; you are not required to use it, but may be asked to replicate problems on it if you seek help from the instructor or TA.

Schedule

The schedule of lectures, exams, and programmings assignments is shown below. In addition to the assignments shown here, there will be a short homework due most weeks.

Sept. 6	Operating system organization, context switching and scheduling	
Sept. 13		
Sept. 20	Virtual Memory	
Sept. 27		Programming assignment 1 due 9/29
Oct. 4	Synchronization and parallel programming	
Oct. 11		
Oct. 18	Midterm exam (in class)	Programming assignment 2 due 10/23
Oct. 25	Hard Drives, Block I/O, RAID	
Nov. 1 (video lecture)		
Nov. 8	File Systems	Programming assignment 3 due 11/17
Nov. 15		
Nov. 22 *no class*	-----	
Nov. 29	Security	
Dec. 6	Hardware Virtualization	Programming assignment 4 due 12/11
Dec. 13	Final Exam	Final exam (in class)

Grading

Your final grade will be calculated as follows:

Class Participation	5%
Short homeworks	10% (1% each)
Programming assignments	40% (10% each)
Mid-term exam	20%
Final exam	25%

Class participation

Class participation is a more significant factor in your grade than its percentage would seem to suggest, as it is graded on a scale from 0 to 5 points, and students with minimal online or in-class participation will receive a 0.

You are expected to participate in class discussions, both online and in the classroom. Students who do not know answers should ask questions, and students who do not have questions should offer answers.

Policies

Exams are not open-book, but you may bring **your own** hand-written notes (on paper) and refer to them during the test. Calculators are allowed (although not needed), but internet-connected devices of any form may not be used.

Collaboration: Students are expected to complete programming assignments as part of a 2-person team which will submit a single copy of an assignment, with both students receiving the same grade. If you feel that your partner is not doing their fair share, please let me know. Discussion of the homework between members of different groups is allowed, but each group must always submit their own independently-written work. Plagiarism-detection software will be used to compare submissions from different teams and across semesters.

Computer use during lecture is discouraged, except in the case of in-class exercises. Note that use of a laptop for note-taking is not very useful, as you will not be able to use typed notes during an exam.

Online communications are an important part of this class. In particular, students are responsible for receiving email at their husky.neu.edu address, and successful delivery of an email to this address will be considered sufficient notice for any group or individual communications from the instructor.

Academic Integrity

Unattributed submission of another person's work, whether a classmate's or material you found on the Internet, use of unauthorized materials or communication with other students during an exam, and other forms of cheating will result in referral to OS-CCR and/or serious penalties. Although discussion of programming projects between groups is allowed and encouraged, submission of code written by anyone other than yourself or another group member is not allowed, and will be automatically checked for. All students must adhere to the Northeastern University Academic Integrity Policy available on the Northeastern web site (see <http://www.northeastern.edu/osccr/academic-integrity-policy/>) and the graduate student handbook.

Remember – your work = your fingers on your keyboard.

If you are cutting and pasting code (instead of typing it yourself) you're probably doing something that could earn you a 0 for the assignment or an F in the class. If you're renaming variables to hide what you did, you are almost certainly risking a 0 or an F.

Special accommodations

If you have any specific disabilities which may require accommodation for this course, please contact the Disability Resource Center at (617) 373-2675.