Computer Science 3220 / Electrical and Computer Engineering 3220 - Section 2
Introduction to Operating Systems
Spring 2018

This file can be found in http://www.cs.clemson.edu/~cmarti2/3220/3220.18.spring.html

---

Course Description: Detailed study of management techniques for the control of computer hardware resources. Topics include interrupt systems, primitive level characteristics of hardware and the management of memory, processor, devices, and data.

Key Topics:

- hardware support of operating systems
- synchronization of concurrent threads
- scheduling of threads
- allocation of physical and virtual memory
- storage of data in files

Learning Outcomes: Students who successfully complete this course should be able to:

- Explain the objectives and functions of a modern operating system.
- Contrast kernel and user mode in an operating system.
- Explain the actions of hardware in response to an interrupt.
- Describe the difference between processes and threads.
- Demonstrate the potential run-time problems arising from the concurrent operation of many separate threads.
- Use a synchronization technique to control concurrency among multiple threads.
- List the four necessary conditions for deadlock to occur.
- Compare and contrast the common algorithms used for both preemptive and non-preemptive scheduling of threads.
- Describe the data structures needed to support thread management and thread scheduling.
- Explain the need for dynamic priority adjustments in thread scheduling (e.g., priority aging and priority inheritance).
- Implement a policy for variable-length memory allocation.
- Describe how a virtual memory address is mapped into a physical memory address in a computer system with paged virtual memory.
- Describe how files are stored in secondary storage.
- Explain how an access control list protects files.

Prerequisites: CPSC 2120 and CPSC 2310 with a C or better; or ECE 2230 and ECE 2720 with a C or better.

Meetings: 4:00-5:15 MW in Jordan G033.

Instructor: Carl Martin, 114 Ravenel Center, cmarti2@clemson.edu, (864) 656-8068

Office Hours: 5:15-6:15 MW

Required Textbook: Thomas Anderson and Michael Dahlin, Operating Systems: Principles & Practice, 2nd edition, Recursive Books, 2014. companion web site

Grading:

- 1000 points in the semester will be awarded
- 200 points - Final Exam - 200 points x1
- 300 points - In-class Exams - 100 points x3
- 400 points - Papers/Programs - 100 points x4
- 100 points - Chapter Quizzes - 10 points X10
- **A**:1000-900, **B**:899-800, **C**:799-700, **D**:699-600, **F**:599-0
- Attendance at the scheduled exams is required; an absence will be counted as a zero unless you have an excused absence.
- Regular attendance is not graded; but, chapter quizzes will only be given one time and missing them will be counted as a zero unless you have an excused absence.
- Any exam that was scheduled at the time of a class cancellation due to inclement weather will be given at the next class meeting unless contacted by the instructor. Any assignments due at the time of a class cancellation due to inclement weather will be due at the next class meeting unless contacted by the instructor. Any extension or postponement of assignments or exams must be granted by the instructor via email or Canvas within 24 hours of the weather-related cancellation.
- Please wait up to 10 minutes if I am late to class.

Drops Days: Drop w/o record by Wednesday, January 24, 2018; drop w/o grade by Friday, March 16, 2018.

Projects:

- Project1: TBD
- Project2: TBD
- Project3: TBD
- Project4: TBD

Fall 2017 projects were a paper on OS security flaws, a threaded-merge sort, and a simulator for scheduling algorithms.

Summer 2017 projects were a paper on OS security flaws, a program for bit-mapped memory allocation, and a pthread program for priority-based locks.

Spring 2017 projects were a paper comparing microkernel and monolithic kernel designs, a pthread program to implement RC4000-like message passing primitives, a slab memory allocator, and a paper on the NTFS and ReFS file systems.

Fall 2016 projects were a paper on interrupt handling, a pthread program to implement a FCFS reader/writers lock, a program for boundary-tag memory allocation, and a program to display LRU page replacement actions.

Programs:

- Programs must run on School of Computing servers. (Your laptop may have applications and/or libraries not available on the servers; thus, if your program makes use of these, your program cannot be run and graded on the servers.)
- Programs are due by 11:59PM on the due date.
- Programs that work correctly for a few test cases but not for all reasonable inputs will not be given full points. It is your responsibility to code correctly for reasonable inputs.

- Late programs will be accepted but 10 points will be deducted for each day late, up to five days. Thus a program that is 4 days late will have a maximum possible point total of 60.
- Unless otherwise noted, programming is to be individual work. Do not discuss the solution to an assignment with anyone else (either a class member or someone else). If you have questions, ask me in person or by email.

Clemson Statement of Academic Integrity: "As members of the Clemson University community, we have inherited Thomas Green Clemson's vision of this institution as a 'high seminary of learning.' Fundamental to this vision is a mutual commitment to truthfulness, honor, and responsibility, without which we cannot earn the trust and respect of others. Furthermore, we recognize that academic dishonesty detracts from the value of a Clemson degree. Therefore, we shall not tolerate lying, cheating, or stealing in any form."

Work turned in for credit in a previous class: When appropriate and when fully documented, you may reuse your own work from a previous class. For example, if you are reusing parts of a program written for a previous course, add a notice to your program header comments about the scope of the reuse and the course in which the work was previously submitted.

Accessibility Statement: Clemson University values the diversity of our student body as a strength and a critical component of our dynamic community. Students with disabilities or temporary injuries/conditions may require accommodations due to barriers in the structure of facilities, course design, technology used for curricular purposes, or other campus resources. Students who experience a barrier to full access to this class should let the professor know, and make an appointment to meet with a staff member in Student Accessibility Services as soon as possible. You can make an appointment by calling 656-6848, by emailing studentaccess@lists.clemson.edu, or by visiting Suite 239 in the Academic Success Center building. Appointments are strongly encouraged - drop-ins will be seen if at all possible, but there could be a significant wait due to scheduled appointments. Students who receive Academic Access Letters are strongly encouraged to request, obtain and present these to their professors as early in the semester as possible so that accommodations can be made in a timely manner. It is the student's responsibility to follow this process each semester. You can access further information here: http://www.clemson.edu/campus-life/campus-services/sds/.

Title IX (Sexual Harassment) Statement: Clemson University is committed to a policy of equal opportunity for all persons and does not discriminate on the basis of race, color, religion, sex, sexual orientation, gender, pregnancy, national origin, age, disability, veteran's status, genetic information or protected activity in employment, educational programs and activities, admissions and financial aid. This includes a prohibition against sexual harassment and sexual violence as mandated by Title IX of the Education Amendments of 1972. This policy is located at http://www.clemson.edu/campus-life/campus-services/access/title-ix/. Ms. Alesia Smith is the Clemson University Title IX Coordinator. She also is the Director of Access and Equity. Her office is located at 110 Holtzendorff Hall, 656-3311 (voice) or 656-0899 (TDD).

Note: The instructor for this course reserves the right to change this syllabus. Announcements will be made in class if and when such changes occur.

Topical Schedule:

- OS definition, evaluation, and history (Chapter 1, slides)
- Kernels, interrupts, and processes (Chapter 2, slides)
- Programming interface (Chapter 3, slides)

- Threads and concurrency (Chapter 4, slides)

- Exam 1 - **Wednesday, February 14, 2018**

- Synchronized access to shared objects (Chapter 5, slides)
- Multiprocessor locking (Chapter 6, slides)
- Scheduling (Chapter 7, slides)
- Address translation (Chapter 8, slides)

- Exam 2 - **Wednesday, March 14, 2018**

- Virtual memory and replacement policies (Chapter 9, slides)
- Advanced memory management (Chapter 10, slides)
- File systems (Chapter 11, slides)
- Storage devices (Chapter 12, see above)

- Exam 3 - **Wednesday, April 18, 2018**

- Files and directories (Chapter 13, slides)
- (Time permitting) Reliable storage (Chapter 14, slides)

- Final Exam - **Tuesday, May 1, 2018, 7:00 pm - 9:30 pm**

Online Resources for Similar Courses:

- CS 140: Operating Systems, John Ousterhout, Stanford
- CS 162: Operating Systems and Systems Programming, Ion Stoica, UC Berkeley
- CSE 451: Operating Systems, John Zahorjan, UW
- CMPT 300 videos, Introduction to Operating Systems, Arvindh Shriraman, Simon Fraser University

- Allen Downey, The Little Book of Semaphores, 2nd ed., version 2.2.1, 2016

- pthread library links
  - Alfred Park, Multithreaded Programming (POSIX pthreads Tutorial)
  - Remzi Arpaci-Dusseau and Andrea Arpaci-Dusseau, Operating Systems: Three Easy Pieces, on-line book
  - Oracle, Multithreaded Programming Guide, Programming with Synchronization Objects
  - Helgrind tool for detecting synchronization errors in pthread code (part of Valgrind tool set)

[CPSC homepage] [Clemson University homepage]

cmarti2@clemson.edu