
Instructor Sanjiv K. Bhatia
Office 317 CCB (Phone 314.516.6520)
e-mail sanjiv@acm.org
url <http://www.cs.umsl.edu/~sanjiv/classes/cs4760>
Office Hrs M 2:00pm – 3:30pm
W 3:30pm – 5:00pm
Any other time by appointment
Prerequisites CS 2700, CS 2750, Stat 1320, Math 2450
Textbooks Stallings. *Operating Systems*. Prentice-Hall, Englewood Cliffs, NJ, 2005.
Kifer and Smolka. *OSP: An Environment for Operating Systems Projects.*, Addison-Wesley, 1991.
Robbins/Robbins. *Unix Systems Programming – Communication, Concurrency, and Threads*.
Prentice-Hall, 2003.

/objectives

- History, Function, Types, and Structure of Operating Systems.
- Concurrent Programming, Process Interaction and Synchronization.
- Process Management and Scheduling.
- Deadlock Problem.
- Memory Management.
- File System and I/O management.
- Security (if time allows).

/etc/policy I'll expect you to be present in most of the classes. I will not be taking attendance but if you start missing too many classes, please take responsibility for your absence, specially when it concerns tests and homeworks. When you come to class, you must change your cell phones to silent mode. In the tests, the cell phones must be completely switched off.

The grade will be based on programming assignments and three tests. All tests will be open book and open notes but no electronic devices will be permitted. Each assignment must be meticulously documented and clearly identify its purpose, author, and date. If you miss any test or assignment without making prior arrangements, you will have a zero. I will not give any make up tests. The distribution of grades will be as follows:

Programming Assignments and Projects	40%
Three non-comprehensive tests	20% each

/etc/exam_dates

Test 1	September 30, 2009
Test 2	November 04, 2009
Test 3	December 09, 2009

There is no final exam. In case of class cancellation due to weather-related or other campus emergency, the test will be held in the following class period. For the third test, if the campus is closed on December 09, the alternative date will be the December 10, possibly in the conference room of the department.

/etc Failure to hand in any assignment will result in an automatic zero for that assignment. If some student is unable to hand in an assignment by the deadline, he/she must discuss it with me *before* the deadline. I'll encourage you to talk to other students regarding homework but you should not collaborate to the extent that two submissions are copies of each other. If you are found copying an assignment (from another student or internet), or if your submission has *unreasonable similarity* to another submission, you get a zero for that assignment automatically. A second offense will be reported to the University

officials and students involved will face serious consequences. I may ask you to come to my office and explain your code to me; in case you are not able to explain the code to my satisfaction, I'll assign you a zero in that project.

The projects in this class will take up a lot of your time. So, you should start working on those as soon as they are assigned. In the past, the students who have asked a lot of questions have scored better grades. Do not hesitate to ask a question in class, in my office, or over email, especially if you do not have an idea on how to start working on the project.

If you have any disability that requires an accommodation (as per UMSL policy), you must notify me in advance. If you cannot attend the class due to a religious holiday or a university-sanctioned event, please let me know in advance as well. In case you are down with the flu, especially H1N1, please stay absent from the class till you recover, and contact me via phone or email. I'll try my best to make accommodation for you in that case.

/etc/notes You will have an account on the Unix machine `hoare.cs.ums1.edu` in the CS department and you should use it for all assignments. All programs must be done in C. The class notes and old tests are available on the class web page in PDF format. The class email list will be announced on the web page when it is set. You are responsible to sign up for it and can do so from an email account of your choice.

Any unsigned email and email not in plain text will go unanswered by me. Please do not send me any attachments without talking to me first.

Anyone desiring an EXC grade after October 31, 2009 must be passing the course at that point to get EXC.

Calendar

Week 1	Overview of operating systems and concepts
Week 2	Overview of operating systems and concepts; Interprocess Communications
Week 3	Labor Day Holiday; Interprocess Communications
Week 4	Interprocess Communications
Week 5	Interprocess Communications
Week 6	Process Management; Test 1
Week 7	Process Management
Week 8	Process Management
Week 9	Resource Management
Week 10	Resource Management
Week 11	Memory Management; Test 2
Week 12	Memory Management
Week 13	Memory Management
Week 14	Thanksgiving
Week 15	File Systems
Week 16	I/O Management; Test 3

Detailed course outline:

1. History, Evolution, and Philosophy

- Historical information about the development of architectural support for changes in software, and the economic and technical forces behind OS development
- Overview of structuring methods like monolithic, layered, and object-oriented
- Application needs and significant case histories (OS/VM, Unix, MS-DOS)

2. Process Coordination and Synchronization

- Process concept
- Overview of multitasking and role of interrupts
- Concurrent execution and precedence graphs (process flow diagrams)
- Race conditions, Sharing access
- Critical Section Problem
 - Models for two processes, including Dekker's Algorithm

- Models for multiple processes, including Bakery Algorithm
- Semaphores, mutex locks, and monitors

3. Process Scheduling and Dispatch

- Preemptive and non-preemptive scheduling strategies
- Short term, medium-term, and long-term scheduling

4. Deadlocks – Causes, conditions, avoidance, prevention, and resolution

5. Physical and Virtual Memory Organization

- Offset registers, partitions, pages, segments, swapping, overlays
- Caching and associative buffers
- Fetch, placement, and replacement policies
- Thrashing and working sets

6. Device and File Management

- File systems
- File types
- File access and directory management
- Distributed file systems
- Equivalence of files and devices
- Organization of I/O subsystems, free space management, and file layout
- Disk scheduling algorithms

7. Security and Protection

- Overview of system security
- Security methods and devices – protection, access, and authentication
- Memory protection
- Recovery management

Note: I am planning on covering the first 6 sections in detail, and section 7 to some extent if we have time. However, we'll discuss security issues from time to time in earlier sections as well.