

CS330 – Operating Systems Syllabus

<i>Semester</i>	Fall 2006
<i>Class Sessions</i>	Tuesday and Thursday, 2:00 - 3:15
<i>Class Location</i>	Gullickson Hall - Room 206A
<i>Catalog Description</i>	Modern operating systems design and implementation: multi-tasking and time sharing, concurrency and synchronization, interprocess communication, resource scheduling, memory management, deadlocks, I/O, file systems, and security.
<i>Prerequisite</i>	CS 210 – Algorithm Analysis and Design
<i>Credit Hours</i>	3
<i>Textbook</i>	Silberschatz, Abraham, et. al., <i>Operating System Concepts 7e</i> , John Wiley
<i>ISBN</i>	0-471-69466-5

<i>Instructor</i>	Jonathan Thompson
<i>Office Phone</i>	304-696-6349
<i>Office Fax</i>	304-696-5454
<i>EMail</i>	ThompsonJ@marshall.edu
<i>Office Location</i>	Gullickson Hall Room 205C
<i>Office Hours</i>	Posted on office door

Grade Components

10%	In-class quizzes
30%	Programs and Homework
10%	Research Report
20%	Mid-term Exam
25%	Final Exam
5%	Class Attendance & Participation

Grade Scale

A	90-100
B	80-89
C	70-79
D	60-69
F	0-59

Schedule of Topics

This is the approximate schedule of topics, subject to change as we progress through the semester.

Week	Tue	Thu	Reading Assignment
1	22-Aug Course Introduction Overview of OS's	24-Aug	Ch 1
2	29-Aug	31-Aug	Ch 2
3	05-Sep Process Management	07-Sep	Ch 3
4	12-Sep	14-Sep	Ch 4
5	19-Sep	21-Sep	Ch 5
6	26-Sep	28-Sep	Ch 6
7	03-Oct Mid-term Exam	05-Oct	
8	11-Oct	12-Oct	Ch 7
9	17-Oct Memory Management	18-Oct	Ch 8
10	24-Oct	26-Oct	Ch 9
11	31-Oct Storage Management	02-Nov	Ch 10 Ch 11
12	07-Nov	09-Nov	Ch 12 Ch 13
13	14-Nov Protection and Security	16-Nov	Ch 14
-	21-Nov Thanksgiving Break	23-Nov Thanksgiving Break	
14	28-Nov	30-Nov	Ch 15
15	05-Dec Last Class	07-Dec Final Exam 2:00 – 4:00	
-	12-Dec	14-Dec	

Measureable Student Learning Outcomes

A high course grade in CS 330: Operating Systems requires that the student demonstrate most or all of the following:

- Explains the basic components of a computer and how they interact with an operating system.
- Describes classes of operating systems, roles of an operating system, and the services an operating system provides under each role to applications and users.
- Understands process and thread concepts and how an operating system manages processes.
- Explains issues in concurrency and mutual exclusion, process synchronization, and interprocess communication; and demonstrates (by writing and debugging concurrent programs) how the solutions provided by an operating systems to these issues can be utilized by applications and users.
- Describes how input/out devices interface with an operating system, and is knowledgeable about the process for developing device drivers.
- Understands why deadlocks arise in operating systems and their implications, and explains algorithms for avoiding, detecting, and recovering from deadlocks.
- Understands, compares, and contrasts various processor scheduling algorithms.
- Understands, compares, and contrasts various disk scheduling algorithms.
- Articulates the motivation for memory hierarchy, explains techniques for implementing virtual memory, and illustrates how application performance can be improved via reducing virtual memory related overhead.
- Explains file system and directory structure concepts, understands tradeoffs in their design and implementation, and demonstrates how applications interface with a file system.
- Explains operating system security mechanisms and its vulnerabilities, demonstrates how applications and users can make use of the operating systems security.
- Explore current trends in operating systems by performing independent research on a focused technical topic; and present the research to the class.
- Demonstrates insight into the design and implementation of two major operating systems.
- Gained hands-on project experience in designing and implementing a subset of operating system functions.

Course Policies

Attendance

Students are expected to attend all class sessions and participate in class activities. Students are required to take exams at the scheduled class period. Students may take an exam at a different time under one of the following conditions:

- They present a University Excused Absence
- They present a valid medical excuse
- Other extraordinary circumstance as determined by the instructor

Academic Conduct

Learning about programming is a hands-on activity, not something that you can pick up by just reading a book or listening to a lecture. It is important that you do the work yourself to gain this experience. To that end, you may discuss programming concepts and techniques with others, consult the web or other textbooks, or study code that is available from various sources but the work you submit must be your own. Here are some examples of appropriate and inappropriate conduct:

- You need to insert an IF statement in your program and you can't remember whether or not parentheses are required. You ask a friend who says, yes, they are required. This is acceptable.
- You're running late on an assignment and in order to hand a program in on time, you copy ten lines of code from a classmate. This is NOT ACCEPTABLE conduct by either student: you must neither directly copy code from someone else nor offer your code to another student or allow it to be copied.
- After struggling for some time, you do a search on the internet and find a snippet of code that you adapt to your problem and insert into your program. You comment your code to acknowledge the source. This is acceptable.
- After struggling for some time, you do a search on the internet and find a program that does exactly what you need. You submit it as your own work. This is NOT ACCEPTABLE.

It is your responsibility to satisfy the spirit of this conduct. If you have any questions, please ask one of the instructors for clarification. Depending on the severity of the offense, the instructors may:

- Take no action
- Penalize the student on the assignment in question
- Assign the student a failing grade in the course

Communication

Assignments, lecture notes, class communications (e-mail), etc, are all handled via the course WebCT/Vista site. Be sure to log into WebCT/Vista regularly to check for course news.

Late Homework

Homework and programming problems will be assigned during the semester. Assignments submitted after the due date will be assessed a late fee.

Modified 20-Aug-06