CSCI 430/530 **Operating Systems**
Spring 2005

Instructor: Derek Harter
Office: JOUR 208
Phone: 903-886-5402
Email: dharter@online.tamu-commerce.edu
Web Page: http://faculty.tamu-commerce.edu/dharter/tamu/classes/2005spring/csci430
Office Hrs: M-W 9:00am – 11:00am
or by appointment
Class: Jour 234, W 4:30pm – 7:10pm

Course Description: This course focuses on the basics of designing and constructing an operating system (OS), from the basic level of the computer hardware, memory and peripheral devices, up to process management, file systems, etc. We will be using the XINU operating system, developed at Purdue University by Douglas Comer and associates for teaching and learning about OS, as our focus for the class. We will walk through the implementation of this small but complete OS from the hardware up, examining closely the C and Assembly source code that makes it all work. The source code for XINU will be discussed in the class and projects will involve design and implementation of UNIX like features into XINU. The programming will all be in plain C (no object-oriented features will be used or discussed in the class) with OS system calls. The student will have to read, understand, and modify existing C code. Students taking this course for graduate level credit will be assumed to have some familiarity with basic OS concepts and will be given extra assignments and readings into advanced OS concepts.

Goals: After completion of this course you will be familiar with the basic elements of an OS and how they may be implemented. These elements include: scheduling and process management, process coordination, message passing, process creation, threading and forking, memory management, interrupt processing, device I/O and device drivers, file systems and shells. Other goals include learning the UNIX programming environment and simple system administration on UNIX systems.

Prerequisite: CSci 270
CSci 241 (Machine Language and Computer Organization) recommended

Text: Operating System Design: Vol I The XINU Approach
by Douglas Comer and Timothy Fossum
(ISBN: 0136381804)

Supplemental Texts:

Operating Systems: Design and Implementation
by Andrew S. Tanenbaum

Modern Operating Systems (2nd Edition)
by Andrew S. Tanenbaum

The Design of the UNIX Operating System
by Maurice J. Bach

Course Materials:

Source code for the XINU Operating System, a simple C compiler for MS-DOS and other course materials are available on the class home page:
http://faculty.tamu-commerce.edu/dharter/tamu/classes/2005fall/csci430/

Evaluation:
Your grade for the course will be based on the following approximate percentages:

30% Exams (2 exams, one at mid-term and a non-cumulative exam during finals week)
15% Quizzes (4-6 in class quizzes)
25% Homework (5 HW Assignments, including coding and written parts)
30% Projects (3 XINU Programming Project)

Exams: will typically be approximately half theory of OS questions, and half coding problems based specifically on XINU code we have examined. There will be 2 exams, one approximately half-way through the semester and one during finals week. The second exam is non-cumulative and so is not a final but simply a second exam testing general knowledge on the second half of the course.

Quizzes: Small quizzes will periodically be given during regular class periods. The quizzes are a motivation to attend classes and to do the reading ahead of time. They will generally be on the current topic under discussion (e.g. they are not reviews of past material), and will consists of coding questions and theory questions.

Homework: There will be a few homework assignments. Most assignments have both written and coding parts. The coding parts are not necessarily XINU related and will not involve modifications or changes to the XINU code. Instead the purpose of the HW assignments is in evaluating the acquisition of general course concepts.

Projects: There will be 3 programming projects that will be directly related to modifying the XINU operating system, to for example add new functionality or change the implementation of some function of the OS in some way. Projects will generally be more involved that HW assignments. You will need access to the XINU code and the ability to compile it correctly. Downloads and instructions on using the code are given on the class web page (see Course Materials section above for information on accessing).

Letter grades will be assigned according to the following scale:

A at least 90% of the total points
B at least 80% but less than 90% of the total points
C at least 70% but less than 80% of the total points
D at least 60% but less than 70% of the total points
F less than 60% of the total points

You must earn an A on your own. Lower borderline grades may be affected by your class attendance, participation, and behavior; the pattern of your grades; and the class grade distribution.

For details of program requirements, see the separate handout **General Policy for Programming Assignments**.

Makeups: Makeups will not be given unless an excused absence is arranged prior to the exam or quis. You will receive a 0 for any exam or quiz you miss.

Attendance: You are responsible for everything covered in all class meetings. Class attendance will be taken at the beginning of each class.

Drops: If you find that you cannot complete the course, please don't forget to drop. If you're making an obvious effort in the course at the time you drop, you may drop passing no matter what your actual grade might be. If you just disappear, your grade will be whatever you have earned at the end of the semester.

Students requesting accommodations for disabilities must go through the Academic Support Committee. For more information, please contact the Director of Disability Resources & Services, Halladay Student Services Bldg., Room 303D, (903) 886-5835.
"All students enrolled at the University shall follow the tenets of common decency and acceptable behavior conducive to a positive learning environment." (See Student's Guide Handbook, Policies and Procedures, Conduct)
All students should be aware that plagiarism is a serious offense. This is true not only of written essays but also of work written in artificial computer languages such as C++. Copying code for assignments from other students or the internet is not allowed. You may discuss general aspects and strategies of the coding assignments with one another, but you must code the programming assignments on your own.

TENTATIVE SCHEDULE

Week	Date	#	Topic / Activity	Reading
1	19 Jan	1	Introduction, Review of old-style C syntax	Handout
		2	Introduction to UNIX programming environment concepts	Handout
2	26 Jan	1	Introduction to UNIX programming environment concepts	Handout
		2	Introduction to UNIX system administration	Handout
3	2 Feb	1	Introduction to Operating Systems Concepts	Ch 1
		2	Introduction to XINU OS and code **HW #1 Due**	Ch 1
4	9 Feb	1	Review of x86 Assembly and PC I/O	Ch 2
		2	Overview of the PC Machine and Run-Time Env.	Ch 2
5	16 Feb	1	List and Queue Manipulation routines in XINU	Ch 3
		2	Scheduling and Context Switching: Intro do Process Management **HW #2 Due**	Ch 4
6	23 Feb	1	Advanced Process Management	Ch 5
		2	Inter-Process Coordination **HW #3 Due**	Ch 6
7	2 Mar	1	Message Passing	Ch 7
		2	Memory Management **Project #1 Due**	Ch 8
8	9 Mar	1	**Exam 1**	
		2		
9	16 Mar	1	**Spring Break**	
		2		
10	23 Mar	1	Interrupt Processing	Ch 9
		2	Real-Time Clock Management	Ch 10
11	30 Mar	1	Device I/O	Ch 11
		2	Device Drivers **HW #4 Due**	Ch 12
12	6 Apr	1	System Initialization	Ch 13
		2	Window Management **Project #2 Due**	Ch 14
13	13 Apr	1	High-Level Memory Management and Message Passing	Ch 15
		2	More Message Passing	Ch 15
14	20 Apr	1	Disk Driver	Ch 16
		2	Disk Driver **HW #5 Due**	Ch 16
15	27 Apr	1	File Systems	Ch 17
		2	More File Systems	Ch 17
16	4 May	1	System Configuration	Ch 20
		2	**Project #3 Due**	
17	11 May		**Exam 2 (Wednesday May 11 4:30 – 7:10 pm)**	

430/530 Course Objectives

1. Be able to list the basic parts of an operating system and describe each part.
2. Be familiar with the basic PC hardware, environment, devices and machine language.
3. Be able to design and use List and Queue manipulation functions
4. Be able to use the queue data structure.
5. Understand various process management algorithms. Be able to code simple process management algorithms.
6. Understand the basics and algorithms of Memory Management in OS.
7. Understand interrupt processing and its importance in OS implementation
8. Understand device I/O and building device drivers
9. Understand windows and tty devices for I/O
10. Be familar wit Disk I/O and its use to implement file systems for OS
11. Understand basic system configuration and administration concepts for UNIX and XINU.