CS 410 – Operating Systems

Syllabus for Fall Semester 2010

(as of August 20th, 2010)

General Course Description: Overview of the concepts/theory of operating systems with emphasis on process management, memory management, file management, scheduling, device management, and synchronization.

Course Format: The course will be taught using a combination of lectures and discussions.

Prerequisites: CS 310 and CS 350

Instructor:

- Dr. Justin Ehrlich, Assistant Professor, Department of Computer Science
- 447P Stipes Hall
- JA-Ehrlich@wiu.edu

Class Meetings:

Section 001: MWF 10:00-10:50am, 211 Stipes Hall
Section 002: MWF 12:00-12:50am, 211 Stipes Hall

Office Hours:

MWF: 9:00-9:50am; 1:00-1:30pm
Other times are available by appointment if these times do not work.

Course Objectives:

- Given definitions and example code showing the use of POSIX functions, the student will develop C programs that demonstrate on a *nix-based system the application of various operating system programming techniques.
- Given concepts and algorithms for operating system process management, the student will describe approaches for processes, threads, CPU scheduling, process synchronization, and deadlocks.
- Given the concepts and algorithms for operating system memory management, the student will describe approaches for handling main memory and virtual memory.
- Given the concepts and algorithms for operating system storage management, the student will describe approaches for file system interfaces and file system implementations.

Required Course Textbook:

- Silberschatz, Galvin, and Gagne. *Operating System Concepts*. 8th Edition. Wiley, 2008

Course Web Site: http://faculty.wiu.edu/JA-Ehrlich/courses/cs410/

Course Topics:

- Part 1: *nix and POSIX
  - Overview and History
  - Shell Programming
  - System Programming
- Part 2: Process Management
  - Processes and Threads
  - Multithreaded implementations
  - Scheduling
  - Synchronization
  - Deadlocks
- Part 3: Memory Management
  - Segmentation
  - Paging
  - Virtual Memory
- Part 4: Storage Management
  - File system interface
  - File system implementation
  - HDD Structures
  - I/O Systems

Course Software:

- Development Environment: Unix-like environment utilizing POSIX system calls and running the BASH shell. Students can use either uxb3.wiu.edu (Solaris 10) or their own implementations such as CYGWIN, Mac OS X, or Fedora.
- Text editor: vi or EMACS – as these are by far the most prevalent editors in *nix systems, try to learn and stick with one of these.
- All programming assignment must be written in C. I will include examples of both the source and the makefile to get you started. There will also be a Bash scripting assignment.

Score Weighting and Final Grades: Your final grade will be based on weighted scores adding up to a total course score of 100. There will be no curves on individual assignments or exams, but there *may* be one on the final grade if a substantial number of students have less than a B.

Graded Item	Percentage
Programming Assignments	25%
Exercises and Quizes	10%
Midterm	20%
Final Exam	25%
Final Project	20%

Grading scale for undergraduate credit:

Percentage	Grade
90-100%	A
87-89%	A-
83-86%	B+
80-82%	B
77-79%	B-
73-76%	C+
70-72%	C
67-70	C-
60-66%	D
Below 60%	F

Grading scale for graduate credit:

Percentage	Grade
90-100%	A
80-89%	B
70-79%	C
60-69%	D
Below 60%	F

Class Attendance: If you need to miss class, make sure you read the notes on the course website, read the day's textbook section, and email me any assignments that are due BEFORE class starts – otherwise it's late. If you have a medical or family emergency on the day of an exam, email me ASAP and I'll setup a makeup exam.

Exercises:

These will be assigned from the text and will be graded from 0%-100% based upon number of correct answers. These are to be completed on your own.The answers should be turned in electronically in a .doc, .docx, or PDF format via Western Online. To be on time, the submission should be emailed to me before class begins. Late submissions will be docked 25% per day.

Quizzes:

These will be open book quizzes based upon the covered material. They are meant to keep you on track for the exams, for the midterm and final will have questions based, though modified, on these questions. These are to be completed on your own. The due date and time will be before class starts. Quizzes may not be completed late unless a valid excuse is given. The majority, if not all, of the quizzes will be on Western Online.

Program Assignments:

There will be periodic programming assignments to demonstrate applied knowledge of the most important topics in the class. On the day the program is due, I will invite a few volunteers to demonstrate their programs to the class for discussion and feedback. If there are no volunteers, I will randomly pick individuals, so be prepared!

Program assignments will be graded 100% for being on time and performing all of the requirements without error. Unless you come talk to me, late submissions will be docked 25% per day. For C programs, the student is to turn in both the source code and a makefile via email. The projects should be able to compile and execute on uxb3.wiu.edu. For the scripting assignment, the program must be able to execute using Bash on uxb3.wiu.edu. If there are compiling or linking errors, I will contact you and it will be considered late until you fix it and it compiles. If there are semantic errors, but the majority of the program works as specified and you still demonstrate knowledge of the targeted material, 10% will be deducted. If the errors show a lack of understanding, between 50% and 100% will be deducted, depending on the effort and if you sought my assistance prior to the due date. Please see me in advance if you are having difficulties understanding the assignment! Zip or tarball everything into one file and name it with your last name followed by the assignment number, e.g., ehrlich1.tar.gz. If you turn in the assignment 24 hours in advance, I will check it to make sure it compiles and runs correctly giving you a chance to correct it before it's due.

Final Project: Final projects will be assigned after the midterm. The students will choose the projects to conduct, with my approval. The students will be able to work in groups of two or by themselves, but group projects must be more extensive. During the last three days of class, the students will present their projects to the class. Scores from 0-150 will be assigned based upon the amount of work involved, the scope of the project, the learning experience, and the difficulty of the project. A score above 100 signifies extra credit and a full 50 points will allow the student to raise a full letter grade (10%). Extra credit will only be given to those projects that are above and beyond the expectation and full extra credit will only be applied to those extensive projects. Extra credit will be applied AFTER any class curve.

Graduate Students: Graduate students who take this course as a graduate course will earn credit towards their graduate degree. Such credit will require the completion of additional requirements on each assignment.

Exam: This course will have two written exams. Make-up exams will be given only for a valid excuse that has been approved by me. All exams are closed book and closed notes. Exams will consist of a combination of short-answer questions and short program completion. The midterm will cover material from Part 1 and Part 2. The final will cover material from Part 3 and Part 4. There will be a review session during the class before each exam.

E-mail: For quick questions on assignments or programming, send an email or ask in class. For major programming or assignment problems, come talk to me during the office hours.

Academic Honesty: Copied assignments, obtained either from another student or from other sources such as the Internet, will not be tolerated and will result in the completion of an academic integrity incident report. Each incident, if found to be valid, shall result in a zero score for the whole assignment for both the supplier and the recipient. Do not give/accept a paper or electronic solution of an individual assignment to/from another student. Copying on an exam will also result in a zero score for the exam.

Special Accommodations: In accordance with University policy and the Americans with Disabilities Act (ADA), academic accommodations may be made for any student who notifies the instructor of the need for an accommodation. It is imperative that you take the initiative to bring such needs to the instructor's attention, as he/she is not legally permitted to inquire about such particular needs of students. Students who may require special assistance in emergency evacuations (i.e., fire, tornado, etc.) should contact the instructor as to the most appropriate procedures to follow in such an emergency. Contact Disability Support Services at 298-2512 for additional services.