Syllabus

Instructor: Andrew Neel (aneel@memphis.edu)
Office hours: By Appointment only (Please arrange by email 2-3 days in advance)
Location: Dunn Hall Room 124
Time: Mon/Wed 5:30PM-7:00PM
Key Dates:
- Term: Jan 17 – May 5
- Last Day: April 26th
- Exam 1: February 20th
- Exam 2: March 20th
- Exam 3: April 17th
- Final Exam: Wed, May 3rd @ 5:30-7:30PM

Holidays:
- Spring Break: March 6-12
- Study Day: April 27th


Course Description
Operating system services, operating system structures; process and thread scheduling; management of asynchronous processes – concurrency, process synchronization, semaphores, monitors; deadlocks; memory management – paging, segmentation, and virtual memory; file system – access and allocation; hierarchy of storage devices.
PREREQUISITE: COMP 3160, and either COMP 3410 or EECE 4278.

Professional Conduct:
Students are expected to conduct themselves in a professional manner. Each student will further be held accountable to The University of Memphis’s code of conduct.

Classroom Expectations
I expect each student to appear in class prepared to discuss the topics of this course. Appropriate preparation includes but is not limited to reading the text, and reviewing recommended online materials, review of source code when needed. I further expect that each student will participate in classroom discussions.

Grading:
Mastery of this courses material will be evaluated as follows:

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Three (3) exams</td>
<td>60%</td>
</tr>
<tr>
<td>Class project</td>
<td>20%</td>
</tr>
<tr>
<td>Homework</td>
<td>20%</td>
</tr>
</tbody>
</table>

NOTE: I require all students to bring one blue exam booklet for themselves on exam day.
Limited Collaboration Policy:

Students are permitted and encouraged (but not required) to discuss the ideas and concepts of any classroom topic or assignment. Unless otherwise specified, the product of each assignment and test is expected to be sole, individual work each student. Specifically, students can discuss ideas and concepts but one student is not permitted to write code or prose for another student. All help is expected to be documented and credited appropriately.

*Warning 1:* Each student should accept help with care. It is very easy to mislead yourself into believing that you understand a concept when others are providing aid or assisting. In a crunch (such as an exam), this error can prove fatal.

*Warning 2:* Please give help with care. Collaboration is intended to improve the classes understanding of a concept. If too much help is given, students may be enabled to fail!

Plagiarism or cheating behavior in any form is unethical and detrimental to proper education and will not be tolerated. All work submitted by a student (projects, programming assignments, lab assignments, quizzes, tests, etc.) is expected to be a student's own original work. The plagiarism is incurred when any part of anybody else's work is passed as your own (no proper credit is listed to the sources in your own work) so the reader is led to believe it is therefore your own effort. Students are allowed and encouraged to discuss with each other and look up resources in the literature (including the internet) on their assignments, but appropriate references must be included for the materials consulted, and appropriate citations made when the material is taken verbatim.

If plagiarism or cheating occurs, the student will receive a failing grade on the assignment and (at the instructor's discretion) a failing grade in the course. The course instructor may also decide to forward the incident to the University Office of Student Conduct for further disciplinary action. For further information on U of M code of student conduct and academic discipline procedures, refer to: [http://www.memphis.edu/studentconduct/academic-misconduct/process.php](http://www.memphis.edu/studentconduct/academic-misconduct/process.php)

Course Outline:

1. Introduction
2. Processes and Threads
3. Memory Management
4. File Systems
5. Input / Output
6. Deadlocks
7. Virtualization and the Cloud
8. Multiple Processor Systems
9. Security
10. Case Study 1: Unix, Linux, and Android
11. Case Study 2: Windows 8
12. Operating System Design

*Chapters in italics may be assigned as self-study, homeworks, or considered optional as time permits.*