CS 441/541: Operating System Concepts  
Course Syllabus – Spring 2013

General Information

<table>
<thead>
<tr>
<th><strong>Location:</strong></th>
<th>Monday, Wednesday, Friday – Wing 218</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Times:</strong></td>
<td>Monday, Wednesday, Friday, 8:50–9:45 am</td>
</tr>
<tr>
<td><strong>Final Exam:</strong></td>
<td>Friday, May 17 from 4:45–6:45 pm</td>
</tr>
<tr>
<td><strong>Instructor:</strong></td>
<td>Josh Hursey  <em>(Pronounced: “her • sea”)</em></td>
</tr>
<tr>
<td><strong>Office:</strong></td>
<td>Wing 209 (ext. 6812)</td>
</tr>
<tr>
<td><strong>Email:</strong></td>
<td>jhursey (at) cs (dot) uwlax (dot) edu</td>
</tr>
<tr>
<td><strong>Office Hours:</strong></td>
<td>Monday, Tuesday, Wednesday, &amp; Thursday, 2:15–4:15 pm</td>
</tr>
<tr>
<td><strong>Class Web Page:</strong></td>
<td>[<a href="http://cs.uwlax.edu/~jjhursey/teaching/2013/spring-cs441/">http://cs.uwlax.edu/~jjhursey/teaching/2013/spring-cs441/</a>](<a href="http://cs.u">http://cs.u</a> wlax.edu/~jjhursey/teaching/2013/spring-cs441/)</td>
</tr>
</tbody>
</table>

Course Description

The study of the structures and algorithms of operating systems. Operating systems are viewed as managers and controllers of resources such as processors, memory, input and output devices and data. Topics include multiprogramming systems, CPU scheduling, memory management and device management.

Prerequisites

CS 340 and CS 370.

Learning Objectives

Students successfully completing this course will be able to:

- Explain what constitutes an operating system and why it is needed.
- Describe the services an operating system provides to users, processes, and other systems.
- Identify the major components of an operating system.
- Understand the notion of a process (a program in execution) and how it forms the basis of all computation.
- Describe the various aspects of processes, including scheduling, creation and termination, and communication.
- Explain different approaches to communication in client-server systems.
- Know the difference between a thread and a process, and explain the benefits of multithreading.
- Explain various CPU-scheduling algorithms and compare them with respect to turnaround time, throughput, and risk of starvation.
- Explain process synchronization and the critical-section problem.
- Identify both software and hardware solutions to the critical-section problem.
- Understand the nature of deadlocks in a computer system.
- Identify and understand different methods for preventing, avoiding, and detecting deadlocks.
• Understand the major memory-management techniques: contiguous allocation, paging, and segmentation.
• Describe the benefits of a virtual memory system.
• Explain the concepts of demand paging, page-replacement algorithms, and allocation of page frames.
• Understand the principles of the working-set model.
• Explain the properties of file systems and their interfaces.
• Understand file-system design tradeoffs, including access methods, file sharing, protection, and directory structures.
• Discuss file block allocation and free-block algorithms.

Assessment of Learning

CS 441 Rubric
Programming Projects ....................... 50%
Midterm Exam ............................... 25%
Final Exam ................................. 25%

CS 541 Rubric
Programming Projects ....................... 40%
Synthesis Papers ............................ 10%
Midterm Exam ............................... 25%
Final Exam ................................. 25%

Programming Projects
The programming assignments will involve writing C code to gain experience using low-level operating system services. You are not required to know the C programming language before starting this course. The assignments are often structured into multiple deliverables usually due weekly. There will be approximately six programming projects.

Synthesis Papers

Only those students enrolled in CS 541 must complete the synthesis papers.
Students will research an area of operating systems by analyzing three related papers in the domain. The student will discuss the contributions of the papers, and their relationship to one another. The synthesis paper can be a breakthrough experience in independent scholarship for a student. When coupled with one of the programming projects or outside work, the synthesis paper can provide a path for independent scholarship beyond the course. There will be two synthesis papers.
(Modified from Professors B. Plale and K. Siek)

Exams
The exams will be in written format, to be completed in class. The midterm exam will take place during a normal lecture session while the final exam will take place during the University-set time. The final exam will not be cumulative, but only cover the topics not covered by the midterm exam. As a consequence, the combined average score of the two exams must be a passing grade (i.e., D
or higher) for a student to receive a passing grade in this course. Otherwise a failing grade (i.e., F) will be given regardless of the other course assignments.

Letter Grades

The following breakdown of letter grades and percentages will be used throughout the course.

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>100 – 94 %</td>
</tr>
<tr>
<td>A/B</td>
<td>93 – 89 %</td>
</tr>
<tr>
<td>B</td>
<td>88 – 83 %</td>
</tr>
<tr>
<td>B/C</td>
<td>82 – 79 %</td>
</tr>
<tr>
<td>C</td>
<td>78 – 70 %</td>
</tr>
<tr>
<td>D</td>
<td>69 – 60 %</td>
</tr>
<tr>
<td>F</td>
<td>&lt; 60 %</td>
</tr>
</tbody>
</table>

Assignment Submission & Late Policies

Assignments will be submitted electronically via D2L’s Dropbox feature. The assignment will fully describe what is required to be submitted. For programming assignments this usually includes, at least, an archive of the fully-functional code and a brief writeup. Assignments are due on the due date before the start of class hours.

All homework assignments are due by the published time. Assignments submitted within 12 hours after the deadline will receive a 20% penalty. An additional 20% penalty will be applied for every additional 12 hours after that (including weekends) until the penalty reaches 100% at which time no credit will be given.

For example, if an assignment is due on Monday at 8:50 am, the following schedule describes the penalty for submitting that assignment late:

<table>
<thead>
<tr>
<th>Time</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monday after 8:50 am</td>
<td>−20 %</td>
</tr>
<tr>
<td>Monday after 8:50 pm</td>
<td>−40 %</td>
</tr>
<tr>
<td>Tuesday after 8:50 am</td>
<td>−60 %</td>
</tr>
<tr>
<td>Tuesday after 8:50 pm</td>
<td>−80 %</td>
</tr>
<tr>
<td>Wednesday after 8:50 am</td>
<td>−100 %</td>
</tr>
</tbody>
</table>

Exceptions may be granted under extraordinary circumstances if a student requests an extension prior to the deadline. If a request cannot be made prior to the deadline due to a medical (or possibly other type of) emergency, the student must request an extension as soon as possible. Students should not assume that exceptions to the grading policy have been or will be given simply because they have been requested. Exceptions to the grading policy will be made exclusively and explicitly via email or other written communication.

Academic Integrity & Collaboration

Most of the assignments will be completed in small groups of 2-3 students. The synthesis papers (CS 541 only) and the first programming project must be completed independently.

For all of the assignments in this course, it is reasonable to discuss general approaches to problem solutions with students outside of your group. However, the bulk of the work must be completed by the group members. Sharing or copying code from anyone inside or outside of the class, other than your group members, will be treated as a violation of the academic integrity rules. I will
aggressively pursue appropriate disciplinary action for any form of unethical behavior. The UWL procedures for handling academic misconduct can be obtained from the Office of Student Life.

You are expected to be able to individually, fully explain the results obtained for an assignment (including code produced), and may be asked to do so for any given assignment.

How to Do Well in the Course

You want to do well in this course. I want you to do well in this course. Below are some simple ways to help make that happen.

Time management
I cannot stress this enough. Programming courses are notoriously more time consuming than most other courses. You should expect the programming assignments to take a significant amount of time to complete. So start on programming assignments early to allow enough time to reflect on and wrestle with the problem(s).

Come to office hours
If you find yourself stuck on an assignment, struggling with a concept from class, concerned about your progress, or just want to discuss the class then stop by my office and speak with me. Students often find that spending a few minutes talking through a problem with me during office hours saves them hours of struggling alone. If you cannot make my office hours then send me an email and we can setup an appointment. This will help you.

Do the reading before you come to class
To get the most out of the class, you want to read ahead of time; that way, you will not be seeing things in class for the first time, and you can come with questions that you already want answered, to make class time more productive. If you wait until it is time to do a programming assignment to look at the material, it may be too late to help you.

Ask and answer questions in class
I have found that if I sit back and simply listen to others talk about a subject, I can often convince myself I understand something. Then, when I try to apply the concept, I realize that I really do not know what is going on at all. By participating in class, asking questions whenever I do not understand things, and trying to answer them, I find out what I do and do not know, and can do something about it.

Work with others to study and solve problems
When you write up your homework to hand in, you will be doing it on your own, making sure you really understand what you are doing (this is especially helpful come exam time). Before you write your own code, however, you will find things work much better if you work together with others. Form study groups, meet up to discuss how to design a program, bounce ideas off one another over lunch. It is amazing what the wisdom of crowds can produce.

Attendance Policy

Attendance will not be taken formally. However, I hope you will attend all class sessions. Class time will not be taken to review things missed due to lack of attendance. If you do have to miss a class, speak to your classmates about what you missed, and try to get notes and other materials from them.
Students with Disabilities

Any student with a documented disability (e.g., physical, sensory, psychological, learning disability, AD/HD, or are a current or prior military service member with wounded warrior status) who needs to arrange reasonable academic accommodations must contact Disability Resource Services (165 Murphy Library, (608) 785-6900) at the beginning of the semester. In addition to registering with Disability Resource Services, it is the student’s responsibility to discuss their needs with the instructor in a timely manner.

http://www.uwlax.edu/drs/