University of Wisconsin-La Crosse  
CS441/541 - Operating Systems Concepts

Spring 2018 – Section 01 – Credits: 3  
TTh 11:00 - 12:25pm, Wimberly 203

Instructor: Dr. Samantha Foley  
Office: 220 Wing  
Office hours: MW: 12:30 - 2:30pm  
TTh: 1:00 - 2:00pm  
and by appointment.  
Email: sfoley@uwlax.edu  
Web page: [http://cs.uwlax.edu/~ssfoley](http://cs.uwlax.edu/~ssfoley)

Course Details

Catalog 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: CS340; CS370; junior standing.


D2L: Lecture notes, assignment descriptions, assignment dropboxes, grades, and a calendar with due dates will be available on D2L. It is your responsibility to check it in a timely manner.

[http://www.uwlax.edu/d2l/](http://www.uwlax.edu/d2l/)

Assessments

Programming Projects: The programming projects will involve writing C programs in a UNIX environment to gain experience using low-level operating system services. The goal of these assignments is to practice the concepts presented in class. You are not required to know the C programming language or be familiar with the UNIX environment before starting this course, but you will be expected to develop these skills throughout the semester. Resources will be provided to help you with this task. The projects may be structured into multiple deliverables. Projects will be submitted electronically via Autolab or D2L. Pay close attention to how the projects are to be submitted and what files must be included. The project specification will fully describe what is required to be submitted and how.

Additionally, you will be working to develop your software development skills in the area of systems programming with an emphasis on defensive programming, testing, documentation, and team
programming. To these ends, you are encouraged to work in pairs (on some projects) and use professional tools to help you manage your teamwork (git). I strongly encourage you to make this team experience a positive one and that both members of the group contribute to the project and understand all aspects of the work.

**Research Project:** The last assignment will be a research project that has a paper and a presentation component. You will work on selected advanced topics in modern operating systems in groups of 3 or 4. The purpose of this assignment is to understand and present a specific topic in the current field of operating systems and be able to understand how it compares to the foundational concepts presented in class. More details about the project deliverables, topics, group selection, and presentations will be available after spring break. The presentations will take place the last 2 lectures of the semester.

**Exams:** The exams will be in written format, to be completed in class. There will be one exam during the semester which will take place during a normal lecture session, while the final exam will take place during the University-set time. The exams are listed in the course schedule and are not cumulative.

**Additional Requirements for Graduate Credit**

One or more additional components, optional for undergrads, but required for graduate students will be added to assignments.

**Course Outline**

This is the general course outline and is subject to change. The intention of this outline is to help you manage your time. Exact due dates are included in assignment descriptions and any changes will be communicated in a timely fashion.

<table>
<thead>
<tr>
<th>Week</th>
<th>Date (Tuesday)</th>
<th>Topic</th>
<th>Reading</th>
<th>Assignment Due (Points)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1/23</td>
<td>Intro./C Prog.</td>
<td>(1-1.14)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1/30</td>
<td>C Prog./Processes</td>
<td>(3-3.7)</td>
<td>Project 0 (10)</td>
</tr>
<tr>
<td>3</td>
<td>2/6</td>
<td>Scheduling</td>
<td>(5-5.5, 5.7, 5.8)</td>
<td>Project 1 (10)</td>
</tr>
<tr>
<td>4</td>
<td>2/13</td>
<td>OS Structure/Shell</td>
<td>(2.1-2.12)</td>
<td>Project 2 (80)</td>
</tr>
<tr>
<td>5</td>
<td>2/20</td>
<td>Threads</td>
<td>(4-4.6)</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>2/27</td>
<td>Synchronization</td>
<td>(6 - 6.10)</td>
<td>Project 3 (100) - demo</td>
</tr>
<tr>
<td>7</td>
<td>3/6</td>
<td>Deadlock/EXAM</td>
<td>(7-7.8)</td>
<td>Project 4 (20)</td>
</tr>
<tr>
<td>8</td>
<td>3/20</td>
<td>Spring Break</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>9</td>
<td>3/27</td>
<td>Memory</td>
<td>(8-8.6, 8.8)</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>4/3</td>
<td>Memory</td>
<td>(9-9.11)</td>
<td>Project 5 (80) - demo</td>
</tr>
<tr>
<td>11</td>
<td>4/10</td>
<td>I/O</td>
<td>(12-12.10)</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>4/17</td>
<td>File systems</td>
<td>(10-10.7)</td>
<td>Project 6 (100)</td>
</tr>
<tr>
<td>13</td>
<td>4/24</td>
<td>Protection</td>
<td>(11-11.10)</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>5/1</td>
<td>Security</td>
<td>(15.1-15.5, 15.10)</td>
<td></td>
</tr>
</tbody>
</table>

Midterm: Thursday, March 8 - in class

Final Exam: Friday, May 12 - 7:45 - 9:45am
Grading Structure

Projects 40%
Midterm 30%
Final 30%

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>94-100%</td>
</tr>
<tr>
<td>AB</td>
<td>89-93%</td>
</tr>
<tr>
<td>B</td>
<td>83-88%</td>
</tr>
<tr>
<td>BC</td>
<td>79-82%</td>
</tr>
<tr>
<td>C</td>
<td>70-78%</td>
</tr>
<tr>
<td>D</td>
<td>60-69%</td>
</tr>
</tbody>
</table>

Policies

Attendance: While attendance will not be taken formally, you are expected to come to all classes. If a class is missed, it is your responsibility to find out what material was covered.

Missed Exams: Exams missed due to significant planned events must be arranged before the exam, preferably at least one week in advance. Acceptable events include presenting academic work at a conference, or participation in an athletic competition. Family trips and oversleeping are not acceptable events. The exams are announced at the beginning of the semester and the final exam schedule is posted before the semester begins so there is plenty of advance notice to allow you to make sure you are available for the exams. Emergency situations will be handled as they arise and I ask that you contact me as soon as is reasonable.

Student Athletes: If you are to miss class, a quiz, an exam, or presentation due to an athletic event, written notice from your coach must be provided at least one week in advance. It is up to you to tell me in a timely fashion and then we will work out a time to makeup the quiz/exam/other at another time before the scheduled athletic event.

Late Assignments: While it is best to turn in your work on time, I understand there may be times where that is not possible. Assignments that are turned in up to 24 hours late will receive a penalty of 15%. Assignments turned in between 24 and 48 hours late will receive a penalty of 30%. There is no credit given for work turned in more than 48 hours late. If there are external circumstances causing the lateness, you must contact me in writing before the due date so I can consider making accommodations. Note that contacting me to ask for accommodation is not the same as receiving said accommodation.

Academic Integrity: Academic misconduct is a violation of the UW-L Student Honor Code and is unacceptable. I expect you to submit your own original work and participate in the course with integrity and high standards of academic honesty. When appropriate, cite original sources. Plagiarism or cheating in any form may result in failure of the assignment or the entire course, and may include harsher sanctions. Refer to the Student Handbook for a detailed definition of academic misconduct.

Failure to understand what constitutes plagiarism or cheating is not a valid excuse for engaging in academic misconduct. For a light-hearted tutorial on avoiding plagiarism I encourage you to review a ten-minute interactive tutorial from Acadia University: [http://library.acadiau.ca/tutorials/plagiarism/](http://library.acadiau.ca/tutorials/plagiarism/)

UW-L Syllabus Legal Statements

Additional legal statements from the university regarding accommodations and services available at UW-L can be found: [https://www.uwlax.edu/info/syllabus/](https://www.uwlax.edu/info/syllabus/)
Student Learning Outcomes

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.
- Identify potential threats to operating systems and security features designed to guard against them.
- Explain the mechanisms available in an operating system to control access to resources.