Course #:	**CSCI-C 435**
Course Title:	**Operating Systems I**
Course Type:	Required core
Prerequisites:	C243 Data Structures, C335 Computer Structures, and 2 additional courses above C243, such as C308, C311.
Credits:	4
Text Book:	Operating Systems, Design and Implementation by Andrew S. Tanenbaum and Albert S. Woodhull.
References:	Handouts Operating System Concepts, by Silberschatz and Galvin. Modern Operating Systems, by Andrew S. Tanenbaum.
Current Catalog Description:	Design and implementation of operating systems: the process model, process synchronization, semaphores, deadlock management, multi-tasking, multi-threading, interprocess communication, process scheduling, memory management, paging, segmentation, virtual memory management, file system design and implementation, I/O device drivers, interrupt handlers and spoolers. Students will complete the design and implementation of a simulated multi-tasking operating system.
Course Goals	The student who completes this course: • Will understand the basic components and layers in an operating system. • Will understand process scheduling algorithm and their use and impact in operating system. • Will be able to use synchronization services provided by an operating system in implementing multithreaded application programs. • Will understand the issues involved in memory and virtual memory management. • Will understand file systems, their allocation, de-allocation, and maintenance.
Major Topics Covered in the Course	1. Definitions, history, concepts and models 2. Processes and their states, models, process table, IPC, race conditions, mutual exclusion, semaphores, synchronization and scheduling. 3. Memory management, swapping, paging, virtual memory, page table, associate memory, page replacement algorithms (NFU, LIFO, LRU), working set, segmentation. 4. Deadlock (Detection, Avoidance, Resolution, Live lock or Starvation)

<table>
<tr><td></td><td>5. I/O hardware and software (Devices 'character and block', controllers, DMA, Interrupt handlers, device drivers, spooling, daemons, disks and disk scheduling, terminals, Raw and cooked I/O)
6. File System. (File structure, type, attributes, access mechanism, operations, directories and their operations.) (UNIX vs. other)
7. Interprocess Communication (Message Passing, shared memory)
8. Spoolers (Print Spooler, etc.)</td></tr>
<tr><td>Laboratory projects (specify number of weeks on each)</td><td>Installing Linux, recompiling the linux kernel. (2 Hours)</td></tr>
<tr><td>Estimate Curriculum Category Content (Semester hours)</td><td><table><tr><td>Area</td><td>Core</td><td>Advanced</td></tr><tr><td>Algorithms</td><td>30</td><td></td></tr><tr><td>Software Design</td><td>25</td><td>5</td></tr><tr><td>Comp. Arch.</td><td>5</td><td>1</td></tr><tr><td>Data Structures</td><td>4</td><td></td></tr><tr><td>Prog. Languages</td><td></td><td></td></tr></table>Additional hours may be dedicated to curriculum categories not listed above. For example explanation of concepts and theories. Discussion of social and ethical issues, discussion of inter personal relationships and working within groups.</td></tr>
<tr><td>Oral and Written Communications</td><td>Every student is required to submit at least __5___ written reports (not including exams, tests, quizzes, or commented programs) of typically __10___ pages and to make ___1__ oral presentations of typically __15___ minute’s duration.</td></tr>
<tr><td>Social and Ethical Issues</td><td>Discussion of computer security and privacy. No more that 1 lecture is spent on social and ethical issues.</td></tr>
<tr><td>Theoretical Content</td><td>The course introduces the basic theories of operating system design and implementation.</td></tr>
<tr><td>Problem Analysis</td><td>Approximately 5 programs are assigned during the semester. Each assignment corresponds to an operating system layer. Assignments are stated in terms of general requirements. Students must analyze the requirement and develop a design for the solution using concepts introduced in this and prerequisite courses.</td></tr>
<tr><td>Solution Design</td><td>Students are required to design solutions which correspond to various layers of operating systems. Including solution to problems such as scheduling, semaphores, message passing, memory management, file systems, spooling and device</td></tr>
</table>

	drivers.
Prepared By	Wolfer, Hakimzadeh