Course Title	Operating Systems				
Course Code	CSE320				
Course Type	Compulsory				
Level	Bachelor (1s Cycle)				
Year / Semester	3rd Year / 5th Semester				
Teacher's Name	TBA				
ECTS	6	Lectures / week	3 hours / 14 weeks	Laboratories / week	N/A
Course Purpose and Objectives	This course provides the students with a basic understanding of what an operating system is and how it works. The theory illustrates the problems handled by operating systems and concentrates on the applications of this specialized software to a real-world environment. The course follows an embedded laboratory approach, where students are required to utilize a variety of tools during the implementation of lectures for the solution of typical operating systems problems and the development of code for schedulers or page replacement algorithms.				
Learning Outcomes	Upon successful completion of this course, students should be able to: • Explain the role and main functionalities of the Operating System. • Describe the various features of processes as well as CPU-scheduling algorithms. • Analyse the critical-section problem and its software and hardware solutions. • Provide a description of deadlocks and methods for preventing or avoiding deadlocks in a computer system. • Describe various memory-management techniques and explain the concepts of a virtual memory system. • Explain the function of file systems, file system design and file system protection. • Describe new trends in operating system design.				
Prerequisites	CSE200, ECE210	Co-requisites	None		
Course Content	Introduction: Overview of an operating system; importance of operating systems; operating systems as resource managers; The need of Operating Systems, what they do and how they are designed. Operating system protection. Basic system resources: The hardware; an overview; main memory; the central processing unit, the registers; input and output devices; the secondary storage devices; interfaces; control unit; and channels.				

<table>
<tr><td></td><td>Process Management: The process concept and concurrency. Process scheduling, interprocess communication, process synchronization, and deadlock handling. Critical-Section; Problem and solutions (software, semaphores etc). Classical Problems of Synchronization (The Readers and Writers, Dining-Philosophers etc). Deadlock characterization. Methods for handling Deadlocks. Deadlock Prevention. Deadlock Avoidance. The Banker's Algorithm. Deadlock Detection.

Multiprogramming and Time-sharing: Software for multiprogramming and Time-sharing; allocating CPU time; main memory allocations; job scheduling; registers; Input/Output device allocation; control of data resources; secondary storage space management.

Memory Management: Memory allocation and memory management; processor management and priorities; interrupts and the flow of control; input/output device allocation; Segmentation; Paging and Virtual memory; segmentation systems; paging systems; virtual memory; implementing virtual memory.

File Systems: Physical storage of data. File operations (create, write, read, delete). Access methods (sequential, index etc.). Directory Systems (single-level, tree-structured). File Protection.

Trends in Operating system design; case study typical OS (Windows, UNIX, Solaris). Recent developments and contemporary issues pertaining to the subject-matter of the course.</td></tr>
<tr><td>Teaching Methodology</td><td>Face- to- face</td></tr>
<tr><td>Bibliography</td><td>“OPERATING SYSTEM CONCEPTS” by Silberschatz/Galvin, Addison-Wesley

“OPERATING SYSTEMS INTERNALS AND DESIGN PRINCIPLES” by William Stallings, Prentice Hall

“OPERATING SYSTEMS” by Gary Nutt, Addison-Wesley

“OPERATING SYSTEMS-DESIGN AND IMPLEMENTATION” by Andrew Tanenbaum, S., Prentice Hall</td></tr>
<tr><td>Assessment</td><td><table><tr><td>Examinations</td><td>70%</td></tr><tr><td>Assignments/Lab</td><td>20%</td></tr><tr><td>Class Participation and Attendance</td><td>10%</td></tr><tr><td></td><td>100%</td></tr></table></td></tr>
<tr><td>Language</td><td>English</td></tr>
</table>