CS 4323: Design and Implementation of Operating Systems I

Required Course: Required
Course Number: CS 4323
Course Name: Design and Implementation of Operating Systems I
Credit Hours: 3
Lecture Hours: 3
Lab Hours: 0
Instructors: Dr. Shital Joshi

Book Title(s): "Operating System Concepts", 9th Edition
Book Author(s): Avi Silberschatz, Peter Baer Galvin and Greg Gagne
Book Year(s):

Course Description: The course covers the key concepts in modern operating systems. The specific topics include process management, synchronization mechanisms, scheduling strategies, deadlock detection/avoidance and prevention, memory management, file systems, protection and security. Concepts will be reinforced through hands-on application of reading assignments and lecture materials through homework assignment, including programming projects and case studies involving Windows and Unix operating system.

Course Prerequisites: CS 2133 (Computer Science II); and CS 3443 (Computer Systems) or ECEN 3213(Computer Based Systems in Engineering); and CS 3653 (Discrete Mathematics for Computer Science) and CS 3353(Data Structures and Algorithm Analysis I) with a grade of "C" or better.

Course Goals: Upon the successful completion of the course, the students will be able to:

- Describe components of operating system and its interaction.
- Evaluate various policies for scheduling, deadlock, memory management, synchronization, system calls and file systems.
- Design and construct various OS software components like system calls scheduler, memory management and file management.

Student Outcomes:

Student Outcome	Course Outcome
1	• Identify the type of operating system suitable for a given application. • Justify the need and objective of job scheduling criteria with relevant example.

	• Analyze the risk of starvation and deadlock with the choice of CPU scheduler. • Evaluate the requirements for the process synchronization and co-ordination in contemporary operating system.
2	• Select appropriately among processes, user threads, or kernel threads to solve a concurrent Problem • Compare the strength and weakness of various CPU scheduling algorithm in terms of wait time, turnaround time, throughput and context-switching implementation challenge. • Compare different page replacement algorithms in terms of page fault • Identify both software and hardware solutions to the critical-section problem. • Identify and understand different methods for preventing, avoiding, and detecting deadlocks.
3	• Understand the feature of the given Raid level structure of hard disk. • Understand the management of memory management techniques. • Describe the services an operating system provides to users, processes, and other systems.
4	• 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
5	• Participate effectively in a team environment. • Ability to function effectively on teams to accomplish a common goal.
6	• Apply appropriate algorithms to avoid deadlock for given concurrent processes • Design a program to take the advantage of multiple processors. • Implement concurrent execution of program to maximum CPU utilization. • Apply suitable data passing mechanism among various processes running on either same machine or on different machine.

Course Topics:

Knowledge Areas that contain topics and learning outcomes covered in the course:

Knowledge Area (KA)	Total Hours of coverage
Operating Systems (OS)	26
System Fundamentals (SF)	5
Networking and Communication (NC)	3

Body of Knowledge coverage:

KA	Knowledge Unit	Topic Covered	Hours
OS	Overview of Operating Systems	Role and purpose of OS, Key design Issues	1
OS	Operating System Principles	Processes, Process control, threads, interrupts and context-switching	3
OS/SF	OS Scheduling and Dispatch, Resource allocation and scheduling	CPU scheduling, dispatcher, Scheduling policies, Deadlines, real-time concerns	4
OS	Concurrency	Basics of exclusive and synchronization, Interrupts, deadlocks, pthreads interface	4
OS	Memory Management	Storage systems, memory management, working sets and thrashing; latencies, caching, locality, cache consistency, fault handling	5
OS	File System	Files (metadata, operations, organization, etc), standard implementation techniques, file system partitioning, virtual file systems, memory mapped files, log structured file system	2
OS	Device Management	Serial and parallel devices, device drivers, interfaces, direct memory access, recovery from failure	3
OS	Security and Protection	Overview of system security, policy, access control, protection and authentication	2
OS	Virtual Machines	Paging and virtual memory, virtual file system, virtual devices and I/O, virtualization and its tradeoffs, hypervisor design	4
SF	Virtualization and Isolation	Rationale for protection and predictable performance, levels of indirections, methods of implementing virtual memory and virtual machines	3
NC	Reliable data delivery	OS role in reliable data delivery	1
	Networked application	Role of layering, role of OS in network naming scheme	1
	Routing and Forwarding	Role of OS in routing and forwarding	1