Department of computer science and Engineering

Academic Year	:	2025-2026	Course Name	:	Operating Systems
Semester	:	3RD SEM	Course Code	:	BCS303
Scheme	:	2022	L:T:P: S	:	3:0:2:0 Credits-4
Total Contact hours	:	40 Hours Theory + 20 Hours Lab	CIE Marks	:	50
Course Plan Author	:	Mrs. Manjushree G	SEE Marks	:	50
Date	:	21-07-2025	Total Marks	:	100

Course Prerequisites:

- Basic Programming in C
- Basic Computer Organization
- Basic Data Structures

Learning Objectives:

- To Demonstrate the need for OS and different types of OS
- To discuss suitable techniques for management of different resources
- To demonstrate different APIs/Commands related to processor, memory, storage and file system management.

Course Outcomes:

CO		*At the end of the course, student should be able to . . .*	**Blooms' Level**
CO1	:	Describe the structure and functionality of operating system.	L2
CO2	:	Apply appropriate CPU scheduling algorithms for the given problem.	L3
CO3	:	Analyze the various techniques for process synchronization and deadlock handling.	L3
CO4	:	Apply the various techniques for memory management.	L3
CO5	:	Apply the various techniques for secondary storage management strategies.	L3

Blooms' Taxonomy:

L1	L2	L3	L4	L5	L6
Remembering	**Understanding**	**Applying**	**Analyzing**	**Evaluating**	**Creating**

Program Outcomes:

PO1	**:**	Engineering knowledge	**PO7**	**:**	Environment and sustainability
PO2	**:**	Problem analysis	**PO8**	**:**	Ethics
PO3	**:**	Design/development of solutions	**PO9**	**:**	Individual and team work
PO4	**:**	Conduct investigations of complex problems	**PO10**	**:**	Communication
PO5	**:**	Modern tool usage	**PO11**	**:**	Project management and finance
PO6	**:**	The engineer and society	**PO12**	**:**	Life-long learning

Program Specific Outcomes:

PSO1:	Model computational problems by applying mathematical concepts and design solutions using suitable data structures & algorithmic techniques.
PSO2:	Demonstrate basic knowledge of computer science in efficient design of problem solutions of varying complexity.
PSO3:	Create career path to become a successful computer science professional, entrepreneur and relish for higher studies.

CO-PO-PSO Mapping:

CO	Program Outcomes													
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 12	PSO 1	PSO 2	PSO 3
CO1	1	-	-	-	-	-	-	-	-	-	-	2	1	1
CO2	2	1	-	-	-	-	-	-	-	-	-	2	1	1
CO3	2	2	1	-	-	-	-	-	-	-	-	2	1	1
CO4	2	1	-	-	-	-	-	-	-	-	-	2	1	1
CO5	1	1	-	-	-	-	-	-	-	-	-	2	1	1
Target	1.5	1.25	1	-	-	-	-	-	-	-	-	2	1	1

Course Content (Syllabus):

Module 1	**CH**
Introduction to operating systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and Security; Distributed system; Special-purpose systems; Computing environments**.** **Operating System Services:** User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure; Virtual machines; Operating System debugging, Operating System generation; System boot. **Textbook 1: Chapter – 1 (1.1-1.12), 2 (2.2-2.11)**	**08**
Module 2	
Process Management: Process concept; Process scheduling; Operations on processes; Inter process communication **Multi-threaded Programming:** Overview; Multithreading models; Thread Libraries; Threading issues. **Process Scheduling:** Basic concepts; Scheduling Criteria; Scheduling Algorithms; Thread scheduling; Multiple-processor scheduling, **Textbook 1: Chapter – 3 (3.1-3.4), 4 (4.1-4.4), 5 (5.1 -5.5)**	**08**
Module 3	
Process Synchronization: Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; **Deadlocks:** System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock. **Textbook 1: Chapter – 6 (6.1-6.6), 7 (7.1 -7.7)**	**08**
Module 4	
Memory Management: Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. **Virtual Memory Management:** Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. **Textbook 1: Chapter -8 (8.1-8.6), 9 (9.1-9.6)**	**08**
Module 5	
File System, Implementation of File System: File system: File concept; Access methods; Directory and Disk structure; File system mounting; File sharing; **Implementing File system:** File system structure; File system implementation; Directory implementation; Allocation methods; Free space management. **Secondary Storage Structure, Protection**: Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management;	**08**

Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix. **Textbook 1: Chapter – 10 (10.1-10.5) ,11 (11.1-11.5),12 (12.1-12.5), 14 (14.1-14.4)**	

Schedule of Instruction:

Class No	Topic	Date	RBT	CO	Mode
Module-1 **Introduction to Operating Systems**					
1.	**Introduction to Operating Systems** Definition, goals, what OS does, examples	15/09/2025	L2	CO1 ,5,6	PPTs , Chalk & Talk
2.	**Computer System Organization & Architecture** Hardware components, single vs multi-core, multiprocessor, clusters	17/09/2025	L2		
3.	**System Structures I** OS structure, operations, user & kernel modes, system components	18/09/2025	L2		
4.	**Process, Memory & Storage Management** Process management basics, memory management overview, storage basics	18/09/2025	L2		
5.	**Protection, Security & Computing Environments** Protection vs security, distributed systems, special-purpose systems, cloud, virtualization	22/09/2025	L2		
6.	**OS Services & User Interface** OS services to users & programs, User-OS interfaces, command-line & GUI	23/09/2025	L2		
7.	**System Calls, System Programs & OS Implementation** Types of system calls, system programs, OS design & implementation, structure models	24/09/2025	L2		
8.	**Virtual Machines, Debugging, Generation & Booting** Concept of VMs, OS debugging tools, system generation, system boot process	25/09/2025	L2		
9.	Module1-Test	29/10/2025			
Module-2 **Process Management**					
10.	**Process Concept** Program vs Process, PCB, process states, process control	30/09/2025	L2		
11.	**Process Scheduling Introduction** Scheduling levels, schedulers, context switching	6/10/2025	L2		

Class No	Topic	Date	RBT	CO	Mode
12.	**Operations on Processes** Creation, termination, fork(), exec(), process hierarchy	7/10/2025	L2	CO1 ,2	PPTs , Chalk & Chalk & Talk
13.	**Inter Process Communication (IPC)** IPC basics, shared memory, message passing	8/10/2025	L2		
14.	**Multithreading Overview & Models** Threads vs processes, Many-to-One, One-to-One, Many-to-Many models	9/10/202	L2		
15.	**Thread Libraries & Threading Issues** POSIX threads, thread creation, cancellation, thread pools	13/10/2025	L2		
16.	**Process Scheduling I** Scheduling criteria, FCFS, SJF — Gantt charts & problems	14/10/2025	L3		
17.	**Process Scheduling II** RR, Priority Scheduling,	15/10/2025	L3		
18.	**Process Scheduling II** Thread Scheduling, Multiprocessor Scheduling	16/10/2025	L3		
19.	Module-2:Case Study	23/10/2025	L3		
Module-3 **Process Synchronization**					
20.	**Critical Section Problem**	27/10/2025	L2	CO3 , 2	PPTs , Chalk & Talk
21.	Peterson's Solution + Synchronization Hardware	28/10/2025	L2		
22.	**Semaphores:** Definition, types	29/10/2025	L2		
23.	**Semaphores:** Classical Problems	30/10/2025	L2		
24.	**Deadlocks**: System Model & Characterization	3/11/202	L2		
25.	**Deadlock Prevention:** Mutual Exclusion,Hold and Wait, No Preemption, Circular Wait	4/11/202	L2		
26.	**Deadlock Avoidance:** Banker's Algorithm, Resource-Allocation Graph Algorithm	5/11/202	L2		
27.	**Deadlock Detection Problem solving**	6/11/2025	L3		
28.	**Deadlock Detection & Recovery solving**	10/11/2025	L3		
29.	Module-3:Case Study	11/11/2025	L3		
Module-4 **Memory Management**					

Class No	Topic	Date	RBT	CO	Mode
30.	**Memory Management Overview** Basics, role of memory management, requirements	17/11/202	L2	CO4 ,1	PPTs , Chalk & Talk
31.	**Swapping** Basic concept, how swapping works, swapping in modern OS	18/11/2025	L2		
32.	**Contiguous Memory Allocation** Single partition, multiple partition, fragmentation types	19/11/2025	L2		
33.	**Paging** Basic concept, address translation, advantages	20/11/2025	L2		
34.	**Page Table Structures** Single-level, multi-level, inverted page tables	24/11/2025	L2		
35.	**Segmentation** Concept, logical vs physical address, comparison with paging	25/11/2025	L2		
36.	**Virtual Memory Concepts** Background, demand paging, copy-on-write	26/11/2025	L2		
37.	**Page Replacement, Frame Allocation, Thrashing** Concepts only — LRU, FIFO, Optimal, allocation methotds, thrashing.	27/11/2025	L2		
38.	**Page Replacement** FIFO & LRU Problem Solving	1/12/2025	L3		
39.	**Page Replacement** Optimal Replacement Problem Solving	2/12/2025	L3		
	Module-4:Case Study	3/12/2025	L3		
Module-5 **File System, Implementation of File System**					
40.	**File Concept** Types of files, attributes, operations	4/12/2025	L2	CO1 ,5,6	PPTs , Chalk & Talk
41.	**Access Methods** Sequential, direct, indexed access	8/12/2025	L2		
42.	**Directory & Disk Structure** Directory structure, single-level, two-level, tree, Mounting concept, overview of sharing	9/12/2025	L2		
43.	**File System Structure & Implementation** Layers of file system, on-disk structures	10/12/2025	L2		
44.	**Allocation Methods** Contiguous, linked, indexed — concepts only Free Space Management & Directory Implementation Bitmaps, free lists, simple directory lookup,	11/12/2025	L2		

Class No	Topic	Date	RBT	CO	Mode
45.	**Secondary Storage & Disk Scheduling** Disk structure, scheduling (FCFS, SSTF)	15/12/2025	L3		
46.	**Secondary Storage & Disk Scheduling** Disk structure, scheduling (SCAN, C-SCAN),	16/12/2025	L3		
47.	disk management overview, Goals of protection, Principles of protection, Domain of protection, Access matrix.	17/12/2025	L2		
48.	Module 5-Test	18/12/2025			

Practical Component of IPCC

Lab	Topic	Date	RBT	CO	Mode
1.	Develop a c program to implement the Process system calls (fork (), exec(), wait(), create process, terminate process)	18,19/09/2025	L3	CO1,2	Demo
2.	Simulate the following CPU scheduling algorithms to find turnaround time and waiting time a) FCFS b) SJF c) Round Robin d) Priority.	25,26/9/2025		CO2	
3.	Develop a C program to simulate producer-consumer problem using semaphores.	3,9/10/2025	L3	CO3	
4.	Develop a C program which demonstrates interprocess communication between a reader process and a writer process. Use mk fifo, open, read, write and close APIs in your program.	10,16/10/2025	L3	CO1,3	
5.	Develop a C program to simulate Bankers Algorithm for DeadLock Avoidance.	17,23/10/2025	L3	CO3	
6.	Develop a C program to simulate the following contiguous memory allocation Techniques: a) Worst fit b) Best fit c) First fit.	24,30/10/2025	L3	CO4	
7.	Develop a C program to simulate page replacement algorithms: a) FIFO b) LRU	31/10/25,6/11/2025	L3	CO4	
8.	Simulate following File Organization Techniques a) Single level directory b) Two level directory	7,13/11/2025	L3	CO5	
9.	Develop a C program to simulate the Linked file allocation strategies.	14,20/11/2025	L3	CO5	
10.	Develop a C program to simulate SCAN disk scheduling algorithm.	21,27/11/2025	L3	CO5	

Textbooks:

T1	Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 8th edition, Wiley-India, 2015

Reference books:

R1	Ann McHoes Ida M Fylnn, Understanding Operating System, Cengage Learning, 6th Edition
R2	D.M Dhamdhere, Operating Systems: A Concept Based Approach 3rd Ed, McGraw- Hill, 2013.
R3	P.C.P. Bhatt, An Introduction to Operating Systems: Concepts and Practice 4th Edition, PHI(EEE), 2014.

R4	William Stallings Operating Systems: Internals and Design Principles, 6th Edition, Pearson.

Web links and Video Lectures (e-Resources):

1.	https://youtu.be/mXw9ruZaxzQ
2.	https://youtu.be/vBURTt97EkA
3.	https://www.youtube.com/watch?v=783KAB-tuE4&list=PLIemF3uozcAKTgsCIj82voMK3TMR0YE_f
4.	https://sites.google.com/view/manjujshree-coursewebsite/home

Assessment Schedule:

Assessment Type	Content	CO	Duration	Marks	Date
CIE 1 (Theory)	Module1, 2 & Module 3 (Half)	CO1,2,3,5, 6		15	12/10/25
CIE 2 (Theory)	Module 3(Half), 4 & Module 5	CO1,4,5,6			29/12/25
CIE (Lab) – Record	Program 1 to 10	CO1-CO5		15	weekly
CIE (Lab) – Test	Program 1 to 10	CO1-CO5		10	11,12/12/2025
Unit Test **Case study-1** **Case study-2** **Case study-3**	Module-1&5	CO1-CO5		10	23/10/2025 29/12/2025 11/11/2025 3/12/2025 18/12/2025
Semester End Examination	Modules 1 to 5	CO1-CO5		50	

Manjushree G — **Faculty in charge**

Dr. Sandeep V M — **Course - Coordinator**

Dr. Hemalatha K L — **HOD**

Case Study Guidelines

Each team should address the following:

Case Study Guidelines

Each team should address the following based on respective problem statement

1. **Overview of the OS, Processor, and Memory Configuration**
   - OS version and architecture
   - CPU and RAM (DDRAM) details
   - Key features relevant to CPU, memory, and synchronization management
2. **Implementation Techniques Used**
   - CPU Scheduling: Which scheduler, how tasks are prioritized
   - Synchronization: Mechanisms like mutexes, semaphores, spinlocks
   - Memory: Paging, segmentation, swapping, virtual memory
3. **Core Features and Performance Insights**
   - Multicore handling, fairness, deadlock prevention
   - Memory throughput and optimization methods
   - Real-time task scheduling (if any)
4. **Tools Used for Observation (if applicable)**
   - E.g., `top`, `htop`, Task Manager, Activity Monitor, `vmstat`, `free`, memory profilers
5. **Comparison with Classical Concepts**
   - CPU Scheduling: FCFS, SJF, Round Robin, Priority
   - Synchronization: Peterson's, Dekker's, Semaphores, Mutexes
   - Memory: Paging vs segmentation, static vs dynamic allocation
6. **Conclusion & Key Takeaways**
   - Unique findings or strengths of the platform
   - Real-world implications

Deliverables

- **Format**: 5-8 slide PPT & written report
- **Optional**: Live demo or screenshots of tools used
- **Deadline**: xx-xx-2025

Suggestion for teams

- Split tasks among teammates (research, design, writing, presentation)
- Refer to official OS documentation and credible tech sources
- Use visuals to simplify complex explanations

Sl No	Activity	Topic	Team No (10 Students/Team)
1	**Case Study**	CPU SCHEDULING IS WINDOWS 11, Core i3/i5/i7/i9, Xeon, Atom	**Team-1**
2	**Case Study**	CPU SCHEDULING IS Linux 6.15., Core i3/i5/i7/i9, Xeon, Atom	**Team-2**
3	**Case Study**	Ubuntu 24.04.2 LTS, Core i3/i5/i7/i9, Xeon, Atom	**Team-3**
4	**Case Study**	macOS 15 Sequoia, **Apple Silicon** (M1, M2&M3), Intel Xeon-W (Cascade Lake) and 8th-generation Coffee Lake CPUs	**Team-4**
5	**Case Study**	Synchronization in Windows 11 using Mutexes, Semaphores, and Critical Sections	**Team-5**
6	**Case Study**	Synchronization in Linux Kernel 6.15 with Spinlocks, Mutexes, RCU, and Semaphores	**Team-6**
7	**Case Study**	Ubuntu 24.04.2 LTS: Thread-level synchronization using pthread library	**Team-7**
8	**Case Study**	macOS 15 Sequoia: Grand Central Dispatch (GCD) and synchronization primitives (M1/M2/M3)	**Team-8**
9	**Case Study**	Memory Management in **Windows 11** with 8GB **DDR4/DDR5** on Intel/AMD processors	**Team-9**
10	**Case Study**	Memory Management in **Linux 6.15** with 8GB **DDRAM** on Intel/AMD processors	**Team-10**

11	**Case Study**	**Ubuntu 24.04.2 LTS**: Memory management with Swap, Paging, and DDRAM handling	**Team-11**
12	**Case Study**	**macOS 15 Sequoia**: Unified memory architecture on **M1/M2/M3**, 8GB DDR with Intel Xeon	**Team-12**

Sl No	Activity	Topic	Team No (10 Students/Team)
1	**Case Study**	CPU SCHEDULING IS WINDOWS 11, Core i3/i5/i7/i9, Xeon, Atom	**Team-1**
2	**Case Study**	CPU SCHEDULING IS Linux 6.15., Core i3/i5/i7/i9, Xeon, Atom	**Team-2**
3	**Case Study**	Ubuntu 24.04.2 LTS, Core i3/i5/i7/i9, Xeon, Atom	**Team-3**
4	**Case Study**	macOS 15 Sequoia, **Apple Silicon** (M1, M2&M3), Intel Xeon-W (Cascade Lake) and 8th-generation Coffee Lake CPUs	**Team-4**
5	**Case Study**	Synchronization in **Windows 11** using Mutexes, Semaphores, and Critical Sections	Team-5
6	**Case Study**	Synchronization in **Linux Kernel 6.15** with Spinlocks, Mutexes, RCU, and Semaphores	Team-6
7	**Case Study**	**Ubuntu 24.04.2 LTS**: Thread-level synchronization using `pthread` library	Team-7
8	**Case Study**	**macOS 15 Sequoia**: Grand Central Dispatch (GCD) and synchronization primitives (M1/M2/M3)	Team-8