COURSE CONTENT

OPERATING SYSTEMS LAB

III Semester: CSM
IV Semester: CSE/ CSD

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
24X0577	Core	0	0	2	1	40	60	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes:30				Total Classes: 30		

Prerequisites: Problem Solving using C and C++

Course Overview:

Operating Systems course provides theoretical knowledge about the structure of operating systems, process, memory management and virtual memory implementation principles, input-output management and deadlock avoidance, file system structure. The Operating System manages the computer's software and hardware as well as its memory and processes. Computer operating systems also allow users to see information, create and save files, and use applications such as email and web browsers. Various applications of operating systems include security, job accounting, error detection aids, coordination between other software's and users.

Course Objectives:

1. Describes functionalities of main components in operating systems.
2. Analyze the algorithms used in process management.
3. Gives synchronization and deadlock concepts.
4. Analyze the algorithms used in memory management.
5. Interpret the concepts of input and output storage for file management

Course Outcomes: After Completion of the Course, Students should be able to

- Explain the fundamental concepts of operating systems including system organization, architecture, OS structure, services, and various types of system calls.
- Analyze process management mechanisms including process scheduling algorithms, multithreading models, inter-process communication, and multiprocessor scheduling.
- Apply synchronization techniques and deadlock handling methods for solving critical section problems and ensure safe process execution in concurrent systems.
- Develop and implement C programs to demonstrate Inter-Process Communication (IPC) mechanisms including Pipes, FIFOs, Message Queues, and Shared Memory for efficient data exchange between processes in a Unix/Linux environment.
- Demonstrate understanding of memory management techniques including paging, segmentation, virtual memory, demand paging, and page replacement algorithms.

LISTOFEXPERIMENTS

1. Write C programs to simulate the following CPU Scheduling algorithms

a) FCFS b) SJF c) Round Robin d) priority

2. Write programs using the I/O system calls of UNIX/LINUX operating system (open, read, write, close, fcntl, seek, stat, opendir, readdir)

3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.
4. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls.
5. Write C programs to illustrate the following IPC mechanisms
   a) Pipes b) FIFOs c) Message Queues d) Shared Memory
6. Write C programs to simulate the following memory management techniques
   a) Paging b) Segmentation

TEXT BOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin,Greg Gagne 7th Edition, JohnWiley
2. Advanced programming in the Unix environment, W.R.Stevens, Pearson education.

REFERENCE BOOKS:

1. Operating Systems – Internals and Design Principles, Stallings, 5th Edition, Pearson Education/PHI,2005.
2. Operating System A Design Approach-Crowley,TMH
3. Modern Operating Systems, Andrew S Tanenbaum 2nd edition, Pearson/PHI
4. Unix programming environment, Kernighan and Pike, PHI. / Pearson Education
5. Unix Internals The New Frontiers, U.Vahalia, Pearson Education

ELECTRONIC RESOURCES:

1. **https://www.geeksforgeeks.org/operating-systems/operating-systems/**
2. **https://www.guvi.in/hub/operating-system-tutorial/**
3. **https://www.w3schools.in/operating-system/intro**
4. **https://en.wikipedia.org/wiki/Operating_system**
5. **https://pdos.csail.mit.edu/6.828/**

MATERIALS ONLINE:

1. Course template
2. Open-ended experiments