GANPAT UNIVERSITY

FACULTY OF ENGINEERING & TECHNOLOGY

Programme	Bachelor of Technology	Branch/Spec.	Computer Engineering (Artificial Intelligence)
Semester	IV	Version	2.0.0.0
Effective from Academic Year	2020-21	Effective for the batch Admitted in	July 2019
Subject code	2CEIT401	Subject Name	Operating Systems

Teaching scheme						Examination scheme (Marks)			
(Per week)	Lecture (DT)		Practical (Lab.)		Total		CE	SEE	Total
	L	TU	P	TW					
Credit	3	0	1	-	4	Theory	40	60	100
Hours	3	0	2	-	5	Practical	30	20	50

Pre-requisites:

Course on Data Structure and Programming for Problem Solving

Objectives of the course:

1. To learn the mechanisms of OS to handle processes and threads and their communication.
2. To understand the mechanisms IPC, concurrent process, critical section problem & solution, mutual exclusion.
3. To gain concept of memory management, paging, segmentation & page replacement algorithms.
4. To know the I/O components, device drivers & RAID, Disc scheduling mechanism.
5. To learn the concept of Files system & FILE access methods & protection.

Theory syllabus

Unit	Content	Hrs
1	**Introduction:** What is an OS? Evolution of OS, Services of OS, Types of OS, Different View of OS, Basics of Memory and Architecture	02
2	**Process Management:** Process Abstraction, Process Management, Various Types of Schedulers and it's role, System calls, Threads, Types of Threads, Multi-threading Models, Process Scheduling, Process Scheduling Algorithms	06
3	**Inter Process Communication:** Race Conditions, Critical Section, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's Solution ,The Producer Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing and Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem	08
4	**Deadlock:** Deadlock Problem, Deadlock Characterization, Deadlock Detection, Deadlock recovery, Deadlock avoidance: Banker's algorithm for single & multiple resources, Deadlock Prevention	05
5	**Memory Management:** Paging: Principle Of Operation, Page Allocation, H/W Support For Paging, Multiprogramming With Fixed partitions, Segmentation, Swapping, Virtual Memory: Concept, Performance Of Demand Paging, Page Replacement Algorithms, Thrashing, Locality	08
6	**Input Output Management:** I/O Devices, Device Controllers, Direct Memory Access, Principles Of Input/outputs,	05

	Interrupt Handler, Device Driver, Device Independent I/O Software, Disks: RAID levels, Disks Arm Scheduling Algorithm, Error Handling	
7	**File Systems:** File Naming, File Structure, File Types, File Access, File Attributes, File Operations, Memory Mapped Files , Directories- Hierarchical Directory System, Pathnames, Directory Operations, File System Implementation, Contiguous Allocation, Linked List Allocation, Linked List Using Index, Inodes, Implementing Directories In C, MS-DOS, and UNIX. Shared Files, Disk Space Management, File System Reliability, File System Performance	08
8	Recent trends in Operating system design and their applicability to HPC.	03

Practical content	
Experiments/Practical's would be carried out based on syllabus	
Text Book	
1	Operating System Concepts Essentials by AviSilberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.
2	Operating Systems: Internals and Design Principles, William Stallings, Prentice Hall of India.
Reference Books	
1	Operating System: A Design-oriented Approach by Charles Crowley, Irwin Publishing
2	Operating Systems: A Modern Perspective by Gary J. Nutt, AddisonWesley
3	Design of the Unix Operating Systems by Maurice Bach, Prentice-Hall of India
4	Understanding the Linux Kernel, Daniel P. Bovet, Marco Cesati, O'Reilly and Associates
ICT/MOOCs Reference	
1	https://nptel.ac.in/courses/106106144/
2	https://nptel.ac.in/courses/106/102/106102132/

Course Outcomes:

Upon completion of this course, students will be able to:

1. Create processes and threads.
2. Implement algorithms for process scheduling for a given specification of CPU utilization, Throughput, Turnaround Time, Waiting Time, and Response Time.
3. Apply techniques for optimally allocating memory to processes by increasing memory utilization and for improving the access time.
4. Design and implement file management system.
5. Implement I/O management functions in OS as part of a uniform device abstraction by performing operations for synchronization between CPU and I/O controllers.