Name of the Course: Principles of Operating Systems

Sr. No.	Heading	Particulars
1	**Description the course:**	**Introduction:** The Principles of Operating Systems course introduces students to the fundamental role an operating system plays in managing hardware and software resources. It covers essential concepts like process management, memory handling, file systems, and CPU scheduling, providing a strong foundation in how computers function at a low level. **Relevance:** This course is highly relevant as it bridges the gap between computer hardware and user applications. Understanding the operating system's role helps students grasp how various programs run efficiently and how resources like memory and CPU are allocated and managed. **Usefulness:** The course equips students with practical knowledge that is directly applicable in configuring, using, and troubleshooting different operating systems such as Linux, Windows, and macOS. It also deepens their understanding of how applications interact with system resources, which is critical for developers, testers, and IT professionals. **Application:** Operating system principles are used extensively in fields such as embedded systems, mobile application development, cloud computing, and cybersecurity. From smartphones to servers and IoT devices, the knowledge of OS design and implementation is central to building and maintaining modern computing environments. **Interest:** This course excites students by uncovering the inner workings of the systems they use every day. The hands-on aspects, like working with commands, simulating scheduling algorithms, and exploring file structures, make it both intellectually engaging and practically rewarding. **Connection with Other Courses:** The concepts learned here support and are reinforced in

		several other subjects such as Computer Architecture, Data Structures and Algorithms, System Programming, and Cybersecurity. The interdisciplinary nature of the course strengthens students' overall understanding of the computer science domain. **Demand in the Industry:** There is a consistent demand in the IT industry for professionals who understand the workings of operating systems. Tech companies developing system software, embedded platforms, or managing cloud infrastructure seek candidates who have a strong grasp of OS-level functioning. **Job Prospects:** Career opportunities after learning this subject include roles like system administrator, kernel developer, embedded systems engineer, and DevOps engineer. A deep knowledge of operating systems also opens doors to specialized fields such as cybersecurity, performance tuning, and system software development.
2	**Vertical:**	Major
3	**Type:**	Theory
4	**Credits:**	2 credits
5	**Hours Allotted:**	30 Hours
6	**Marks Allotted:**	50 Marks
7	**Course Objectives (CO):** **CO 1.** To learn basic concepts and structure of operating systems **CO 2.** To understand process communication techniques **CO 3.** To study various CPU scheduling algorithms **CO 4.** To learn about Memory management **CO 5.** To learn about File system management and implementation	
8	**Course Outcomes (OC):** After successful completion of this course, students would be able to - **OC 1.** Work with any type of operating system **OC 2.** Handle threads, processes, process synchronization **OC 3.** Implement CPU scheduling algorithms **OC 4.** Understand the background role of memory management **OC 5.** Design file system	
9	**Modules:-** **Module 1 (15 hours):**	
	Fundamentals of Operating systems – Definition of Operating System, Operating System's role, Operating-System Operations, Functions of Operating System, Computing Environments Operating-System Services, User and Operating-System Interface, System Calls, Types of System Calls, Operating-System Structure	

	Processes – Threads - Overview, Multicore Programming, Multithreading Models, Process Concept, Process Scheduling, Operations on Processes, Inter-process Communication **Process Synchronization** – General structure of a typical process, race condition, The Critical-Section Problem, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors **CPU Scheduling** – Basic Concepts, Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, SRTF, Priority, RR, Multilevel Queue Scheduling, Multilevel Feedback Queue Scheduling), Thread Scheduling	
	Module 2 (15 hours):	
	Deadlocks – System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock **Memory Management** – Main memory background, Logical address space, Physical address space, MMU, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table **Virtual Memory** – Background, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing, Mass-Storage Structure: Overview, Disk Structure, Disk Scheduling, Disk Management **File System Interface and Implementation** – File Concept, Access Methods, Directory and Disk Structure, File-System Mounting, File Sharing File-System Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management	
10	**Text Books** 1. Abraham Silberschatz, Peter Galvin, Greg Gagne, Operating System Concepts, Wiley, 9th Edition	
11	**Reference Books** 1. Achyut S. Godbole, Atul Kahate, Operating Systems, Tata McGraw Hill 2. Naresh Chauhan, Principles of Operating Systems, Oxford Press 3. Andrew S Tanenbaum, Herbert Bos, Modern Operating Systems, 4e Fourth Edition, Pearson Education, 2016	
12	**Internal Continuous Assessment: 40%**	**Semester End Examination: 60%**