**FF No.: 654**

**AI3002: Operating System**

**Course Prerequisites:**

1. Basics of Computer System
2. Computer Organization
3. Data Structures
4. Any Programming Language.

**Course Objectives:**

1. To understand the basic concepts and functions of Operating System.

2. To gain knowledge of process synchronization and its mechanism.

3. To get familiar with CPU scheduling algorithms.

4. To discuss different deadlock handling mechanisms.

5. To learn memory management techniques and virtual memory.

6. To evaluate various disk scheduling algorithms.

**Credits: 4 Teaching Scheme Theory: 2** Hours/Week

**Tut: 1** Hours/Week

**Lab**: **2** Hours/Week

**Course Relevance:**

This course focuses on functions of operating system. Operating system is a System software that manage the resources of the computer system and simplify applications programming. The Operating System acts as a platform of information exchange between your computer's hardware and the applications running on it.

|  |
| --- |
| **SECTION-1** |
| **Introduction:** What is OS?, Interaction of OS and hardware, Goals of OS, Basic functions of OS, OS Services, System Calls, Types of System calls, Types of OS: Batch, Multiprogramming, Time Sharing, Parallel, Distributed & Real-time OS.(4 Hrs)  **Process management:** Process Concept, Process States: 2, 5, 7 state models, Process Description, Process Control, Multithreading models, Thread implementations – user level and kernel level threads, Concurrency: Issues with concurrency, Principles of Concurrency, Mutual Exclusion: OS/Programming Language Support: Semaphores, Mutex, Classical Process Synchronization problems.(7 Hrs)  **Uniprocessor Scheduling:** FCFS, SJF, RR, Priority(3 Hrs) |
| **SECTION-2** |
| **Deadlock:** Principles of deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Deadlock Recovery (4Hrs)  **Memory Management:** Memory Management requirements, Memory Partitioning, Paging, Segmentation, Address translation, Placement Strategies: First Fit, Best Fit, Next Fit and Worst Fit. Virtual Memory, VM with Paging, VM with Segmentation, Page Replacement Policies: FIFO, LRU, Optimal (7 Hrs)  **I/O management:** I/O Devices - Types, Characteristics of devices, I/O Buffering. Disk Scheduling: FCFS, SSTF, SCAN, C-SCAN (3Hrs) |
| **List of Tutorials:**   1. Linux commands 2. Comparison of different OS 3. OS structures 4. Inter Process Communication 5. Symmetric Multiprocessor 6. Thread Scheduling 7. Translation Lookaside buffer 8. Secondary storage management 9. Linux Memory management 10. File System in Windows and Linux |
|  |
| **List of Practicals: (Any Six)**   1. Execution of Basic Linux commands. 2. Execution of Advanced Linux commands. 3. Any shell scripting program. 4. Write a program demonstrating use of different system calls. 5. Implement multithreading for Matrix Operations using Pthreads. 6. Implementation of Classical problems using Threads and Mutex. 7. Implementation of Classical problems using Threads and Semaphore. 8. Write a program to compute the finish time, turnaround time and waiting time for the following algorithms:   First come First serve b) Shortest Job First (Preemptive and Non Preemptive)  Priority (Preemptive and Non Preemptive) d) Round robin   1. Write a program to check whether given system is in safe state or not using Banker’s Deadlock Avoidance algorithm. 2. Write a program to calculate the number of page faults for a reference string for the following page replacement algorithms:    1. FIFO b) LRU c) Optimal |
| **List of Course Projects:**   1. Design and implementation of a Multiprogramming Operating System: Stage I    * 1. CPU/ Machine Simulation      2. Supervisor Call through interrupt 2. Design and implementation of a Multiprogramming Operating System: Stage II    * 1. Paging      2. Error Handling      3. Interrupt Generation and Servicing      4. Process Data Structure 3. Design and implementation of a Multiprogramming Operating System: Stage III    * 1. Multiprogramming      2. Virtual Memory      3. Process Scheduling and Synchronization      4. Inter-Process Communication      5. I/O Handling, Spooling and Buffering |
| **Assessment Scheme:**   1. ESE 2. CVV 3. LAB-Course Assignment and Project Evaluation 4. Programming Practical |
| **Text Books:**   1. *Stalling William; “Operating Systems”; 6th Edition, Pearson Education;* 2. *Silberschatz A., Galvin P., Gagne G.;“Operating System Concepts” ; 9th Edition; John Wiley and Sons;* 3. *Yashavant Kanetkar; “Unix Shell Programming”; 2nd Edition, BPB Publications* 4. *Sumitabha Das; “Unix Concepts and Applications”; 4th Edition, TMH.* 5. *D M Dhamdhere; “Systems Programming & Operating Systems”; Tata McGraw Hill Publications, ISBN – 0074635794* 6. *John J Donovan; “Systems Programming”; Tata Mc-Graw Hill Edition, ISBN-13978-0-07-460482-3* |
| **Reference Books:**   1. *Silberschatz A., Galvin P., Gagne G; “Operating System Principles”; 7th Edition, John*   *Wiley and Sons.*   1. *Forouzan B. A., Gilberg R. F.; “Unix And Shell Programming”; 1st Edition, Australia*   *Thomson Brooks Cole.*   1. *Achyut S. Godbole , Atul Kahate; “Operating Systems”; 3rd Edition, McGraw Hill****.*** |
|  |
| **Moocs Links and additional reading material:**   1. [www.nptelvideos.in](http://www.nptelvideos.in) 2. <https://www.udemy.com/> 3. <https://learn.saylor.org/> 4. [https://www.coursera.org/](https://www.coursera.org/learn/os-power-user) 5. <https://swayam.gov.in/> |
| **Course Outcomes:**  Upon completion of the course, student will be able to –  1) Examine the functions of a contemporary Operating System with respect to convenience, efficiency and the ability to evolve.  2) Demonstrate knowledge in applying system software and tools available in modern operating system for process synchronization mechanisms.  3) Apply various CPU scheduling algorithms to construct solutions to real world problems.  4) Identify the mechanisms to deal with Deadlock.  5) Illustrate the organization of memory and memory management techniques  6) Acquire a detailed understanding of various I/O buffering techniques and disk scheduling algorithms. |
| **CO- PO Map**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **CO** | **Programme Outcomes** | | | | | | | | | | | | **Program Specific Outcomes** | | | | | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** | **PSO3** | **PSO4** | | **CO1** | **3** | **2** |  | **3** | **3** |  |  |  |  |  |  |  | **3** | **2** |  |  | | **CO2** | **3** |  | **3** |  | **3** | **2** | **2** | **2** | **2** | **2** | **2** |  |  |  |  |  | | **CO3** | **3** | **2** |  | **2** | **3** | **3** |  |  |  |  |  |  | **3** |  | **3** |  | | **CO4** | **2** | **3** |  | **3** |  |  |  |  |  |  |  |  |  | **3** |  | **3** | | **CO5** | **2** |  |  |  |  |  |  |  |  |  |  | **2** |  |  |  |  | | **CO6** | **3** | **2** |  | **2** | **3** |  |  |  |  |  |  | **2** |  | **3** |  | **3** | |
| **CO attainment levels:**   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | CO | CO1 | CO2 | CO3 | CO4 | CO5 | CO6 | | Level | 2 | 2 | 3 | 4 | 3 | 4 | |
| **Future Courses Mapping:**   1. Advance Operating System 2. Unix Operating System 3. Linux programming 4. Distributed System/Computing 5. System Programming |
| **Job Mapping:**   1. Linux Administration 2. Kernel Developers 3. Application Developers 4. System programmer 5. System architect |