

Kadi Sarva Vishwavidyalaya
Master of Computer Application (MCA)
Year – I (Semester – II) (W.E.F. August 2014)
Subject Name: Operating System – MCA 202

Sub Total Credit	Teaching scheme		Examination scheme				
	(per week)		MID	CEC	External		Total Marks
	Th	Pr	Th	Th	Th.	Pr.	
4	3	2	25	25	50	50	150

Rationale (Course Objective):

The primary objective is to impart knowledge about fundamental principles and design issues of Operating Systems. A practical implementation of Operating system concepts using UNIX based C programming.

Learning Outcome: This course will enable to:

- Have a good orientation towards concept-based approach and practical-based approach
- Students will be able to describe the components of a modern operating system
- Explain how they interact with the computer hardware
- Apply operating system concepts practically
- Apply the concepts of operating systems design to practical problems.
- Know the basic theories of all operating systems structure and how an operating system manages the computer systems.

Instructional Strategies:

- Generally lecture method will be adapted.
- Teaching aids such as OHP and LCD projectors will be used.
- Emphasis on self study will be handled through seminars.
- Case study discussion on selected topics

Course Contents:

Unit I: Introduction to Operating System: (20%)

Introduction to Operating System: History of Operating Systems, Operating System Concepts, System Calls, Operating System Structure. Processes and Threads: Processes, Threads, InterProcess Communication, Scheduling.

Unit II: Memory Management: (20%)

Memory Management: A Memory Abstraction, Virtual Memory, Page Replacement Algorithms, Segmentation.

Unit III: File System and Input / Output: (20%)

File Systems: Files, Directories, File System Implementation, File System Management and Optimization. Input/Output: Principles of I/O Hardware, Principles of I/O Software, I/O Software Layers, Disks, User Interfaces.

Unit IV: Deadlock and Multiple Processor Systems (20 %)

Deadlocks: Introduction to Deadlocks, Resources, Deadlock Avoidance, Deadlock Prevention and Other Issues. Multiple Processor Systems: Multiprocessors, MultiComputers, Virtualization

Unit V: Security:**(20 %)**

Security: The Environment, Basics of Cryptography, Protection Mechanisms, Authentication, Insider Attacks, Malware. Case Studies: Linux, Windows Vista and Symbian OS

Text Books:

1. “ Modern Operating Systems”, by Andrew S.Tannenbaum, PHI, 3rd Edition

Chapter & Topics –

Chapter 1: 1.1, 1.2, 1.5, 1.6, 1.7,
Chapter 2: 2.1, 2.2, 2.3, 2.4,
Chapter 3: 3.2, 3.3, 3.4, 3.7,
Chapter 4: 4.1, 4.2, 4.3, 4.4,
Chapter 5: 5.1, 5.2, 5.3, 5.4, 5.5, 5.6,
Chapter 6: 6.2, 6.4, 6.5, 6.6, 6.7,
Chapter 8: 8.1, 8.2, 8.3,
Chapter 9: 9.1, 9.2, 9.3, 9.4, 9.5, 9.7
Case Studies from Chapter 10, 11 & 12

Reference Books:

1. “Operating System Concepts”, William Stallings, Pearson, 5th Ed
2. “Operating Systems”, Madnick E., Donovan J., Tata McGraw Hill, 2001

Practical List:**Practical 1:**

1. Configuring Operating System
2. Basic UNIX Commands

Practical 2 – Process:

3. Header files: Process creation and Process joining
4. Create processes using fork() and check different states i.e. zombie, orphan
5. Sum of numbers from 1 to 10, by dividing the job into two processes(parent and one child)
6. Copy the contents of one array to another.
7. Create two child processes and display the output.
8. Program to add four integer values using 2 process
9. Program to find out the factoids of a number
10. Program to fork a child and print the process id of parent and child process

Practical 3 – Thread:

11. Program to create a thread and join the thread
12. Create four threads and print its output.
13. Program to find whether the number is Prime or not
14. Program to find factorial of the given numbers using threads.
15. Sum of numbers using thread
16. Program to find maximum number from the integer numbers using thread
17. To find the total no of prime numbers between 1 to n by using thread.

Practical 4 – Scheduling:

18. Implement ROUND ROBIN algorithm for CPU scheduling.
19. Implement Shortest Job First algorithm for CPU scheduling.
20. Implement First Come First Serve algorithm for CPU scheduling.
21. Implement Priority for CPU scheduling algorithm

Practical 5 – IPC:

22. Implement IPC using pipe to read and write a string from the user.
23. Using pipe print odd and even numbers.
24. Read n number of characters and print the characters using IPC