RKUNIVERSITY

Course Title	Operating Systems
Course Code	CE410
Course Credit	Theory : 03
	Practical : 01
	Tutorial : 00
	Credits : 04

Course Learning Outcomes

On the completion of the course, students will be able to:
- **Develop** synchronized programs which share resources.
- **Analyze** processscheduling algorithms with reference to their applications.
- **Identify** reasons of deadlocks and **design** deadlock-free systems.
- **Analyze** memory management methods and **Identify** their merits and demerits**.**
- **Compare** file system implementation of different operating systems.
- **Understand** interaction of operating system with I/O hardware.

Sr. No.	Name of chapter & details	Hours Allotted
	Section – I	
1	**Introduction to Operating System** Introduction, Classification of Operating systems, Different structuresof Operating Systems, OS Services, System calls.	04
2	**Concurrent Processes** Process Concept, Process Scheduling, Operations on processes, IPC : Shared Memory and Message Passing,Producer-Consumer problem	06
3	**Process Synchronization** Critical Section Problem, Resolving race conditions with and without busy waiting, Semaphores, Mute, Monitors, Introduction to threads, Thread management, Classical IPCProblems : Sleeping barber, Dining Philosophers, Readers-Writers.	07
4	**CPU Scheduling** Scheduling Concepts, Preemptive and non-preemptive Scheduling, Scheduling Algorithms for batch and interactive OS.	07
	Section – II	

5	**Deadlock** System model, Deadlock characterization, Prevention, Avoidance and detection, Recovery from deadlock.	04
6	**Memory Management** Memory partitioning, Swapping,Virtual Memory, Paging, Page replacement algorithms,Modeling page replacement algorithms, Design Issues of Paging Systems, Implementation Issues in Paging, Segmentation.	08
7	**I/O Management and Disk Scheduling** Principles of I/O hardware, Principles of I/O software, I/O Software Layers, Disks: hardware, formatting, disk arm scheduling algorithms, error handling, stable storage.	07
8	**File System and Directories** File concept (File Naming, File Structure, and File Types), File organization and access mechanism, File Attributes, File Operations, File System Implementation, Directories: Directory Levels, pathnames, Directory Operations.	05

Instructional Method and Pedagogy

- Lectures will be conducted using black-board, power point presentation and videos.
- Lectures will be focused more on problem solving, not just theoretical study.
- Students will be given assignments to practice numerical examples.
- Labs are designed such that students get working knowledge of Linux Operating Systems.
- Some Lab tutorials are aimed to implement IPC concepts.

Reference Books

- Andrew S. Tanenbaum, Modern Operating Systems,PHI Publication.
- Silberschatz, Galvin, and Gagne, Operating System Principles, Seventh Edition, Wiley India Pvt Ltd.
- Achyut S. Godbole, Operating Systems, Second Edition, Tata Mc Graw Hill.
- D. M. Dhamdhare, Operating Systems, Second Edition Tata Mc Graw Hill.
- Sumitaba Das, Unix Concept and Applications,Tata Mc Graw Hill
- YashwantKanitkar,Unix Shell Programming, BPB publications.

Additional Resources

- NPTEL Video Lectures of Operating Systems course of Computer Science & Engineering by Prof. Kumkum Garg, Indian Institute of Technology,Roorkee - 247 667[Available at: http://nptel.ac.in/courses/Webcourse-contents/IISc-BANG/Operating%20Systems/New_Index1.html]
- www.spokentutorial.org (Tutorials on Linux)

RKUNIVERSITY

List of Experiments

Tutorial-1

Understand and practice basic Utility commands in Linux.

[commands like cal,date,echo, printf, bc, script,mailx,passwd, who,uname, tty and sttyp are to be covered]

Tutorial-2

Understand and practice Linux Commands for file and directory handling. Following tasks are included:

- Creating files and directories
- Copying, Moving, deleting and renaming files and directories
- Displaying content of files and directories
- Searching
- Changing rights to access files and directories
- Creating archives and compression

[Commands like cat, mkdir, rmdir, cd, cp, rm, mv, more, wc, file, comm, diff, tar, gzip and chmod are to be covered]

Tutorial-3

Perform following tasks.

- Sort all files of given directory in descending order of file names.
- Sort the files of given directory in descending order of updation time.
- List all files beginning with character 'p'.
- List all files ending with ".dat" and starting character is any.
- List all the files beginning with the character 'h' and also store them in a file called 'hfile'.
- Copy file1 to file2 without using copy command.
- Append contents of file1 and file2 to file3 and display file3.
- Count the total no. of User working in the system.
- Store the sorted list of all users in one file.
- Find the total no. of blank lines in a given file.

- Find the files that have only read & execute permissions for others.

Tutorial-4

Develop shell scripts for following problems:

- Accept the month and year from the user and display the corresponding calendar.
- Accept an alphabet from user and list all the files starting from that alphabet.
- Accept three numbers. Find maximum and minimum from them.
- Accept three numbers. Find sum and average of them.

Tutorial-5

Develop shell scripts for following problems:
- Accept two numbers and operator from user. Carry the specified operation and display the result. Possible operators are +,-,*,/
- Accept two numbers from command line. Display all prime numbers in that range.
- Accept a string from user. Check whether string is palindrome or not.
- Accept a number from user. Find the factorial of it.

Tutorial-6

1) **Write** a C program as follows : When program runs, child process is created using fork(). Child displays message "Hello, I am child, my ID is <child's ID>". Parent waits for child to finish its execution. Parent displays message "I am parent, my process ID is <process ID>".
2) **Write** a C program as follows: It is same as previous one. But parent should wait for 50 seconds and then display the message.

Tutorial-7

Write twoC programs, 'client' and 'server'. When 'server' runs, it creates a shared memory segment. Then string "Hello I am server" is written in shared memory. When 'client' runs, it reads from shared memory and displays the string on screen.

Tutorial-8 [Mini Project]

Write two C programs, 'producer' and 'consumer'. They have shared a buffer. 'Producer' writes into buffer. 'Consumer' reads from buffer. Use Semaphores to achieve synchronization.

Tutorial-9

To **Understand and practice** Linux Commands for User management. Following tasks are included:

- Creating and deleting new users
- Changing passwords
- Assigning users to groups
- Concept of sudo users