GUJARAT TECHNOLOGICAL UNIVERSITY

Syllabus for Integrated MCA 2nd Semester
Subject Name: Operating Systems (OS)
Subject Code: 2628604

1. Learning Objectives:

- To provide an understanding of the basic concepts and role of modern Operating systems.
- To understand the concept of the interaction between user application, OS and hardware architecture.
- To provide an understanding of concept of process, concurrency, deadlock prevention, avoidance, detection and recovery for identification of solution strategies.
- To understand the issues involved with preemptive vs non-preemptive scheduling.
- To study the concept of management of memory, disk and file. Students can understand the efficient use of primary and secondary storage for their application.

2. Prerequisites:

- Basic knowledge of computer hardware and software.
- Knowledge of programming languages like C, C++, etc.

3. Contents:

Unit No.	Course Content	Weightage Percentage
1	**Operating System Overview and Processes** Operating system Overview: Operating system (OS) objectives and Functions, Evolution of OS, Major Achievements of OS, Developments Leading to Modern OS, OS design considerations for multi-processor and multi-core Process Description and Control: Process, Process State, Process Description, Process Control and Execution of the OS. Threads: Process and Threads, Types of threads	10%
2	**Concurrency Control and Deadlocks** **Concurrency - Mutual Exclusion and Synchronization:** Mutual Exclusion: Software Approaches, Principles of Concurrency, Mutual Exclusion: Hardware Support, Semaphores, Monitors, Message Passing, Reader / Writer Problem. **Deadlocks:** Deadlock and Starvation: Principles of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, An Integrated Deadlock Strategy, Dining Philosophers Problem	25%

3	**Memory** **Memory Management:** Memory Management Requirements, Memory Partitioning, Paging, Segmentation. **Virtual Memory:** Hardware and Control Structures, Operating System Software	20%
4	**Scheduling** **Uni processor Scheduling:** Types of Processor Scheduling, Scheduling Algorithms, Traditional UNIX **Multiprocessor, Multi core and Real time Scheduling:** Multiprocessor and Multi-core Scheduling, Real time Scheduling	15%
5	**Input / Output Management** **I/O Management and Disk Scheduling:** I/O Devices, Organization of the I/O Function, OS Design Issues, I/O Buffering, Disk Scheduling, RAID, Disk cache	20%
6	**Virtual Machines And Modern Operating Systems** **Virtual Machines:** Virtual Machine Concepts, Hyper-visors, Container virtualization, Processor issues, Memory Management, I/o Management, VMware ESXi **Modern Operating Systems Overview:** Cloud Operating Systems and IoT Operating Systems	10%

Desirable Additional Topics for Seminar / Reading:

1) Case Study: Mobile Operating Systems (Android& iOS)
2) Distributed Processing, Client/Server Architecture and Clusters
3) Operating System Security
4) File Management

4. Main Reference Book(s):
1. Stalling W, "Operating Systems", Pearson, 9th edition

5. Additional Reference Book(s):

1) Andrew S. Tanenbaum, Herbot BOS, "Modern Operating Systems", Pearson, ISBN 978-93-978-9325-7577-6
2) Silberschatz, A., Peter B. Galvin and Greg Gagne, "Operating System Principles",
3) Wiley-Indian Edition, 9th Edition
4) Ann McHoes, I M Flynn, "Understanding Operating Systems", Cengage India Publication, 8th Edition,
5) Bach M J, "The Design of UNIX Operating System", Prentice Hall India, 1993.

GUJARAT TECHNOLOGICAL UNIVERSITY

Syllabus for Integrated MCA 2nd Semester
Subject Name: Operating Systems (OS)
Subject Code: 2628604

With effective
from academic
year 2018-19

6. Chapter-wise Coverage from Main Reference Book:

Unit#	Topics
1	Chapter 2 (2.1 to 2.4, 2.6), Chapter 3 (3.1 to 3.5), Chapter 4 (4.1, 4.2)
2	Chapter 5, Chapter 6 (6.1 to 6.6)
3	Chapter 7 (7.1 to 7.4), Chapter 8 (8.1 to 8.2)
4	Chapter 9 (9.1 to 9.3), Chapter 10 (10.2 and 10.2)
5	Chapter 11 (11.1 to 11.7)
6	Chapter 14 (14.1 to 14.7), Chapter 16 (16.2,16.4)

8. Accomplishments of the Student after Completing the Course:
- Understanding the principles and internals of Operating System.
- Working knowledge of LINUX/UNIX System.
- Harnessing the facilities provided by Operating System in application development.

Practical List

1) Shell Commands
- date, ls, who, cal, ps, wc, cat, uname, pwd, mkdir, rmdir, cd, cp, rm, mv, diff, chmod, grep, sed, head, tail, cut, paste, sort, find, awk
- Regular expression using sed, grep and awk
- Regular expression using meta characters (^, *, [], (dot), escape, <word>, $, { }, -,+)

2) Regular Expression

1	Write a command to display all lines which begins with "NOKIA" string from anprd.lst file. [prd.lst file should exist with data]
2	Delete first, last and all the blank lines from the specified file.
3	Searches for a line which does not start with the vowel letter in any specific file
4	Create a file named customer.dat which contains customer name, address and city. Sort the file in ascending order of customer name and then by city and redirect the output to name.out file.
5	Create a file named customer.dat which contains customer name, address and city. a) Display all the lines who are started with A to M. b) Display 5 lines after the match of the word "Bombay".

6	Create columnar file named "emp" contains Empno, Empnm, Dept, Salary and Hiredate. Write required command for following: a) Cut the Empnm and Salary column and transfer it to empsal file. Display the difference between "emp" and "empsal" file and also display total number of words and characters of "emp" and "empsal" file. b) Sort the data in descending order of joining year and also transfer the sorted output in "experience" file and display only unique lines of "experience" file.
7	Create columnar file named "emp" contains Empno, Empnm, deptno, Desig, and Salary. Write required command for following a. Cut the deptno, Empnm and Salary column and transfer it to empsal file. Also display first 4 and 7th onwards employees from empsal file.
8	Create columnar file named "emp" contains Empno, Empnm, Desig, and Salary. Write required command for following b) Cut the empno,empnm and transfer them into "empinfo.out" file and count frequency of "SHAH" word of that file using single command
9	Display all such files from your login which has size >= 50.
10	Create columnar file named "restaurants" which contains restaurant name, pizza variety, delivery area and prize columns - Cut the restaurant name and price column and transfer it to file named "economy" and display first 3 three lines of the file "economy".
11	Display all the files with read and write permissions throughout your login and save output in file and also display on terminal
12	Write a command to substitute „/" with „:" throughout the file with all occurrences of /etc/group
13	Using awk command Count number of lines found in a file
14	Using awk command find Sum of fields (Marks). File Format roll#, name, mark1, mark2, mark3
15	Using awk command Remove duplicate lines (similar to uniq) from file.
16	Using awk command Delete all Blank lines
17	Using awk command Print Last Line of a file
18	Using awk command find and replace "Hindustan" or "Bharat" to "INDIA"
19	Using awk command Print first line of a file
20	Using awk command search for IND, BHA and HIN
21	Using awk command search for IND and BHA and HIN (in that order)

22	Using awk command print only lines of less than 65 characters
23	Using awk command print section of file between two regular expressions (inclusive)
24	Using awk command print lines between 5 and 8 the line

3) Shell Scripts

1	Write a script to compare identically named files in two different directories and if they are same, copy one of the min a third directory.
2	Write a script to copy the file system from two directories to anew directory in such a way that only the latest file is copied in case there are common files in both the
3	Write a script for generating a mark sheet after reading data from a file. File contains student rollno, name, and marks of three subjects.
4	Write a script to make following file and directory management operations menu based: 1. Display current directory 2. List directory 3. Make directory 4. Change directory 5. Copya file 6. Rename a file
5	Write a script which reads a text file and output the following 1. Count of character, words and lines. 2. File inreverse. 3. Frequency of particular word in the file 4. Lower case letter in place of upper case letter.
6	Write A Script To Perform Following String Operations Using Menu: 1. COMPARE TWO STRINGS. 2. JOINTWO STRINGS. 3. FINDTHE LENGTH OF AGIVENSTRING. 4. OCCURRENCE OFCHARACTER ANDWORDS 5. REVERSETHE STRING.
7	Write a shell script to add the statement #include <stdio.h> at the beginning of every C source file in current directory containing printf and fprintf.
8	Write a script that behaves both in interactive and non-interactive mode. When no arguments are supplied, it picks up each C program from current directory and lists the first 10 lines. It then prompts for deletion of the file. If the user supplies arguments with the script, then it works on those files only.
9	Write a script that deletes all leading and trailing spaces in all lines in a file. Also remove blank lines from a file. Locate lines containing only printf but not fprintf.

10	Write a script to display the files 1) Having size more than user entered from given directory. 2) Files not accessed in last Month 3) List zero size files 4) Delete all output files (files with extension lst and out).
11	Write a Script for Simple Database Management System Operation. Database File Contains Following Fields. (EMP_NO,EMP_NAME,EMP_ADDRESS,EMP_AGE,EMP_GENDER,EMP_DESIGNATION, EMP_BASIC_SALARY) Provide Menu Driven Facility For VIEW RECORD BASEDON QUERY 1. DELETE RECORD 2. MODIFY RECORD. 3. COUNT TOTAL NUMBER OF RECORDS 4. List employees in descending order of salary for gender 5. EXIT Create file with required data.

4) Practical List for OS Conceptual clarity (desirable)

1	Write a C program for implementing concurrency using producer, consumer problem using
2	Write a C program for implementing dining philosopher problem under LINUX.
3	Write a C program for implementing UNIX Processor Scheduling algorithms 1) FCFS 2) RR 3) SPN 4) SRT