Program	Bachelor of Technology (BTech)	**Semester - 4**
Type of Course	Professional Core	
Prerequisite	Basic knowledge of data structures and C programming language.	
Course Objective	This course enables students to understand the importance of an Operating system, its functionalities to manage resources of Computer and Peripherals, program development, and its execution. The student will acquire knowledge of Process Management, Memory Management, File Management, and I/O Management.	

Teaching Scheme (Contact Hours)				Examination Scheme				
				Theory Marks		Practical Marks		
Lecture	Tutorial	Practical	Credit	SEE	CIA	SEE	CIA	Total Marks
3	0	2	4	70	30	25	25	150

***SEE** - Semester End Examination, **CIA** - Continuous Internal Assessment (It consists of Assignments/Seminars/Presentations/MCQ Tests, etc.)*

Course Content

T - Teaching Hours | **W** - Weightage

Sr.	Topics	T	W
1	**Introduction**	8	20
	Definition of Operating Systems, Generations of Operating Systems, Types of Operating Systems, OS Services, System Calls, OS structure: Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine, Basic concept of Multiprogramming, Multiprocessing, Multitasking, Time Sharing Operating Systems, Parallel Processing Operating System and Distributed Operating Systems		
2	**Process and Thread Management**	8	20
	Processes: Definition, Process states, Process state transitions, Process Control Block, Context switching, Basic concepts of threads, Benefit of threads, Type of threads, Multi Threading Models. **Process Scheduling**: Definition, Scheduling objectives, Types of Schedulers, Scheduling criteria (Definition only): CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time, Scheduling algorithms: FCFS, SJF, SRTN, RR, Pre-emptive and Non pre-emptive priority		
3	**Interprocess Communication**	10	20
	Interprocess Communication, Race Condition, Critical Section, Mutual Exclusion, Hardware solution, Disabling interrupts, Shared lock variable, Strict alternation, TSL (Test and Set Lock) instruction, Exchange instruction, Dekker's solution, Peterson's solution, Producer Consumer Problem, Semaphores, Monitors, Mutex, Pipes and Message passing, Barrier and Signal, Classical IPC Problems: Reader's & Writer's Problem, Dinning Philosopher Problem **Deadlock:** Deadlock Definition, Deadlock characteristics, Deadlock detection and recovery, Deadlock avoidance: Banker's algorithm, Deadlock prevention		
4	**Memory Management**	10	20
	Basic concepts of memory, Logical and Physical address map, Memory partitioning: Fixed and Variable Partitioning, Memory Allocation: Allocation Strategies (First Fit, Best Fit, and Worst Fit), Swapping, Paging and Fragmentation, Demand Paging Virtual Memory: Basics concepts of Virtual Memory, Page Replacement Policies (FIFO, LRU, Optimal, Other Strategies), Thrashing		
5	**I/O Management and File Management**	9	20
	I/O Management: I/O Devices, Organization of I/O functions, Operating system design issues, I/O buffering, Disk Scheduling (FCFS, SCAN, C-SCAN, SSTF, LOOK, C-LOOK), RAID, Disk Cache **File Management:** File concept, Access methods, File types, File operation, Directory structure, File system structure, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency & performance		
	Total	45	100

Suggested Distribution Of Theory Marks Using Bloom's Taxonomy

Level	Remembrance	Understanding	Application	Analyze	Evaluate	Create
Weightage	30	50	20	0	0	0

NOTE : This specification table shall be treated as a general guideline for the students and the teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Course Outcomes

At the end of this course, students will be able to:	
CO1	**discuss** the generations, structures and functions of Operating System.
CO2	**implement** the CPU scheduling algorithms for Process management.
CO3	**describe** Inter-Process Communication problems, its solutions, and deadlock.
CO4	**explain** the role of paging, segmentation and virtual memory in operating systems.
CO5	**practice** file management and IO management methods.

Reference Books

1.	**Modern Operating Systems (TextBook)** By Andrew S. Tanenbaum | PHI
2.	**Operating Systems: Internals & Design Principles (TextBook)** By William Stallings | Pearson Education India
3.	**Operating System Concepts (TextBook)** By Peter B. Galvin, Greg Gagne, Abraham Silberschatz | John Wiley & Sons, Inc.

List of Practical

1.	**Unix/Linux Introduction** Development of Unix/Linux, Role & Function of Kernel, System Calls, Elementary Linux command & Shell Programming.
2.	**Demonstration of Unix Commands: cd, ls, man, echo, cal, date, clear, cat** **Part : A** Perform the following Unix Commands: cd, ls, man, echo, cal, date, clear, cat **Part : B** 1. Write a C Program to simulate the LS Command.
3.	**Demonstration of Unix Commands: who, whoami, uname, passwd, mkdir, rmdir, cp, mv, rm, cut, paste, more** **Part : A** Perform the following Unix Commands: who, whoami, uname, passwd, mkdir, rmdir, cp, mv, rm, cut, paste, more **Part : B** 1. Write a C Program to simulate the CP Command.
4.	**Demonstration of Unix Commands: cmp, comm, diff, chmod, chown, chgrp, file, finger, sleep, kill, ps, wc** **Part : A** Perform the following Unix Commands: cmp, comm, diff, chmod, chown, chgrp, file, finger, sleep, kill, ps, wc **Part : B** 1. Write a C Program to simulate the WC Command.
5.	**Demonstration of Unix commands: ln, nl, head, tail, sort, find, uniq, tr, history, pipe, write, wall** **Part : A** Perform the following Unix Commands:

	ln, nl, head, tail, sort, find, uniq, tr, history, pipe, write, wall **Part : B** 1. Write a C Program to simulate the CMP Command.
6.	**Implementation of IF statement using shell script** **Part : A** Write a Shell Script: 1. Which works like a calculator and performs below operations Addition, Subtract, Division and Multiplication. 2. To find a largest number from 2 numbers. 3. To check whether given no is ODD or EVEN. **Part : B** Write a Shell Script: 1. To check whether given no is NEGATIVE or POSITIVE. 2. To check whether given no is divisible by 5 or not. **Part : C** Write a Shell Script: 1. To check whether a given year is leap year or not. 2. To accept two integers and check whether they are equal or not.
7.	**Implementation of IF ELSE statement using shell script** **Part : A** Write a Shell Script: 1. To check whether a number is greater than 10 or not. 2. To find a largest number from 2 numbers. 3. To check whether given no is ODD or EVEN. **Part : B** Write a Shell Script: 1. To check whether given no is NEGATIVE or POSITIVE. 2. To check whether given no is divisible by 5 or not. **Part : C** Write a Shell Script: 1. To check whether a given year is leap year or not. 2. To accept two integers and check whether they are equal or not.
8.	**Implementation of NESTED IF using shell script** **Part : A** Write a Shell Script: 1. To check given year is Leap year or not. [If a year can be divisible by 4 but not divisible by 100 then it is leap year but if it is divisible by 400 then it is leap year] 2. Write a shell script to generate mark sheet of a student. Take 3 subjects, calculate, and display total marks, percentage and Class obtained by the student. 3. To find a largest number from 3 numbers. **Part : B** Write a Shell Script: 1. To enter basic salary of an employee and calculate Gross salary according to given conditions Basic Salary >= 10000 : DA = 80% of basic salary, HRA = 20% of basic salary + DA Basic Salary >= 20000 : DA = 90% of basic salary, HRA = 25% of basic salary + DA Basic Salary >= 30000 : DA = 95% of basic salary, HRA = 30% of basic salary + DA. **Part : C** Write a Shell Script: 1. To accept two integers and check whether they are equal or not if both numbers are not equal then find the largest number from the two numbers.

9.	**Implementation of IF ELSE IF LADDER using shell script** **Part : A** Write a Shell Script: 1. To print day name based on day number.[Sunday to Saturday] 2. To accept two integers and check whether they are equal or not if both numbers are not equal then find the largest number from the two numbers and also check whether the largest number is divisible by 5 or 7 or both. 3. To find a largest number from 3 numbers. **Part : B** Write a Shell Script: 1. To find a largest number from 4 numbers. **Part : C** Write a Shell Script: 1. To input electricity unit charges and calculate total electricity bill according to the given condition: For first 50 units Rs. 0.50/unit For next 100 units Rs. 0.75/unit For next 100 units Rs. 1.20/unit For unit above 250 Rs. 1.50/unit An additional surcharge of 20% is added to the bill.
10.	**Implementation of WHILE Loop using shell script** **Part : A** Write a Shell Script: 1. To print 1 to 10. 2. To print 1 to 'n' numbers. 3. To find Sum & Average of 'n' numbers. 4. To print odd numbers between 1 to n **Part : B** Write a Shell Script: 1. To print numbers between two given numbers which is divisible by 2 but not divisible by 3 2. To find factorial of given number n. 3. To check whether a given number is palindrome or not. **Part : C** Write a Shell Script: 1. To display the multiplication table of the given number.
11.	**Implementation of FOR Loop using shell script** **Part : A** Write a Shell Script: 1. To find the value of one number raised to the power of another. 2. To check whether a given number is prime or not. 2. Which will accept a number n and display first n prime numbers as output? 3. To find first n Fibonacci numbers like: 0 1, 1, 2, 3, 5, 13, … **Part : B** Write a Shell Script: 1. To print sum of series 1 + 4 + 9 + 16 + 25 + 36 + …n 2. To find factorial of the given number. **Part : C** Write a Shell Script: 1. To find factors of the given number
12.	**Implementation of SWITCH CASE using shell script** **Part : A** Write a Shell Script: 1. To read weekday number and print weekday name using switch. 2. To read gender (M/F) and print corresponding gender using switch.

	3. Check whether a character is VOWEL or CONSONANT using switch. **Part : B** Write a Shell Script: 1. To find number of days in a month using switch case. 2. To check whether number is EVEN or ODD using switch. **Part : C** Write a Shell Script: 1. To print total number of days in a month using switch case.
13.	**Implementation of commands using shell script** **Part : A** Write a Shell Script: 1. To scans the name of the command and executes it. 2. To display current month calendar. 3. To validate the entered date. (E.g. Date format is: dd-mm-yyyy) **Part : B** Write a Shell Script: 1. Which will print the following menu and execute the given task? • Display calendar of current month • Display today's date and time • Display usernames that are currently logged in the system • Display your name at given x, y position • Display your terminal number • Exit 1. To checks whether a given user is valid or not. 2. To finds total no. of users and finds out how many of them are currently logged in. **Part : C** Write a Shell Script: 1. To display all executable files, directories and zero sized files from current directory. 2. To display the date, time and a welcome message (like Good Morning etc.). The time should be displayed with “a.m.” or “p.m.” and not in 24 hours notation.

Miscellaneous

Useful Links
https://www.geeksforgeeks.org/ https://www.tutorialspoint.com/ https://nptel.ac.in/ https://www.coursera.org/