Lab-1	**Demonstration of Unix Commands: cd, ls, man, echo, cal, date, clear, cat** **Part: A** 1. Perform the following Unix Commands: cd, ls, man, echo, cal, date, clear, cat **Part: B** 1. Combine the content of file1.txt and file2.txt into a new file named combined.txt. 2. Display the current date and time in the format YYYY-MM-DD HH:MM:SS. 3. Create a file with the current timestamp as its name with content as File created at today's date. 4. Display the current date, append the output to a file log.txt, and then display the file's contents. **Part: C** 1. Display the current month's calendar, write it to a file calendar.txt, and then view the contents of the file. 2. Display the manual page for the ls command, save it to a file ls_manual.txt, and then append a custom message to the file. 3. Clear the terminal screen, move to a directory project, list all the files, and print a message confirming the action. 4. Navigate to a folder called logs, print the current date, and save both the directory name and date to a file log_report.txt.
Lab-2	**Demonstration of Unix Commands: who, whoami, uname, passwd, mkdir, rmdir, cp, mv, rm, cut, paste, more** **Part: A** 1. Perform the following Unix Commands: who, whoami, uname, passwd, mkdir, rmdir, cp, mv, rm, cut, paste, more **Part: B** 1. Copy a file file.txt to a new location backup/file.txt, then move it to diet/file.txt 2. Extract the first and second fields from a file data.txt and merge them with another file extra.txt column-wise. 3. Create a directory called backup and copy all .txt files from the current directory into this backup folder. 4. Extract the first three fields from data1.txt, extract the first three fields from data2.txt, and combine them side by side. **Part: C** 1. Display the system's kernel version and save the output to a file called sysinfo.txt, then copy the file to a folder named system_backup. 2. Create a directory project_backup, move all .sh files from the current directory to this new directory, then delete one of the .sh files from the backup. 3. Extract specific fields from file1.txt and file2.txt, combine them using paste, and display the result page by page using more.

Lab-3	**Demonstration of Unix Commands: cmp, comm, diff, chmod, chown, chgrp, file, finger, sleep, kill, ps, wc** **Part: A** 1. Perform the following Unix Commands: cmp, comm, diff, chmod, chown, chgrp, file, finger, sleep, kill, ps, wc **Part: B** 1. Compare two files file1.txt and file2.txt, and if they are different, display a message and count the number of lines in file1.txt. 2. Compare the contents of config_old.txt and config_new.txt, and if there is a difference, change the permissions of config_new.txt to read-only for all users. 3. Check the type of file1.txt, compare it with file2.txt, and if they differ, count the number of words in file2.txt. **Part: C** 1. Compare two sorted files list1.txt and list2.txt using comm, check the type of list1.txt, and count the number of lines in list2.txt. 2. Compare two files data_old.csv and data_new.csv, and if they differ, count the number of characters in data_new.csv and display the result. 3. Change the permissions of a file script.sh to be executable by the owner, change the ownership to user diet, and after 5 seconds, print a confirmation message.
Lab-4	**Demonstration of Unix Commands: ln, nl, head, tail, sort, find, uniq, tr, history, pipe, write, wall** **Part: A** 1. Perform the following Unix Commands: ln, nl, head, tail, sort, find, uniq, tr, history, pipe, write, wall **Part: B** 1. Create a hard link named file_link to a file file.txt, then display the first 10 lines of file.txt with line numbers. 2. Display the last 20 lines of log.txt, sort the lines, and remove duplicate entries. 3. Find all .txt files in the current directory, display the first 5 lines of the search result, and send the result to user john using write. 4. Display the last 10 commands from the command history and broadcast them to all logged-in users. 5. Add line numbers to the file data.txt, then sort the lines and remove duplicates. 6. Create a symbolic link dir_link to a directory mydir, then find all .sh files in the directory and broadcast the result to all users using wall. 7. Display the first 5 lines and the last 5 lines of file.txt. 8. Find all .log files, sort them alphabetically, and remove duplicates if any. **Part: C** 1. Display the first 10 lines of input.txt, translate all lowercase letters to uppercase, and add line numbers. 2. Create a symbolic link log link to the file logfile.txt, display the last 50 lines of logfile.txt, and remove duplicates. 3. Add line numbers to file.txt, then display the first 15 lines and sort them in reverse order. 4. Remove duplicate lines from data.txt, convert all spaces to underscores. 5. Sort the lines in records.txt, remove duplicates, and count the number of unique lines. 6. Search your command history for all occurrences of chmod, then remove duplicates. 7. Display the first 10 lines of data.csv, then sort them in ascending order.

	8. Replace all lowercase letters with uppercase in message.txt. 9. Find all files larger than 1MB in the current directory, add line numbers
Lab-5	**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. 3. To check whether person is eligible to vote. (age>18) **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. **3.** To check if a triangle is valid based on side lengths. (Sum of two sides are greater than third side)
Lab-6	**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. 3. To check whether person is eligible to vote. (age>18). **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. 3. To check if a triangle is valid based on side lengths. (Sum of two sides are greater than third side)
Lab-7	**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. 2. To check if a number is positive, negative, or zero, and check for even/odd if positive. **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. 2. To check the type of a triangle based on sides. • Check for valid triangle • All sides are equal THEN equilateral • Any two sides are equal THEN isosceles • ELSE scalene
Lab-8	**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 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. 2. Determine the grade based on marks. (Accept marks of 5 subjects) • Percentage > 90 THEN Grade A • Percentage 80 to 90 THEN Grade B • Percentage 70 to 80 THEN Grade C • Percentage 60 to 70 THEN Grade D • Percentage 50 to 60 THEN Grade E • Less than 50 THEN Grade F **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.

Lab-9	**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.
Lab-10	**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. 3. Which will accept a number n and display first n prime numbers as output? 4. 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
Lab-11	**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. 4. For basic calculator. **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. 3. To convert Number to Words (1-5) Using Switch Case

	Part: C Write a Shell Script: 1. To perform basic mathematical based on choice Using Switch Case.
Lab-12	**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 2. To checks whether a given user is valid or not. 3. To finds total no. of users and find 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.
Lab-13	**Implementation of commands using shell script** **Part: A** Write a Shell Script: 1. To Check if a File Exists 2. To Print the Current Date and Time **Part: B** Write a Shell Script: 1. To Create a Directory 2. To Display a List of Files in a Directory **Part: C** Write a Shell Script: 1. To Count the Number of Lines in a File 2. To Check if a Directory Exists

Lab-14	**Implement CPU Scheduling Algorithms in C** **Part: A** Write a C Program to implement: 1. First-Come, First-Served (FCFS) Scheduling algorithm **Part: B** Write a C Program to implement: 2. Shortest Job First (SJF) Scheduling algorithm **Part: C** Write a C Program to implement: 3. Round Robin (RR) Scheduling algorithm
Lab-15	**Implement Page Replacement Algorithms in C** **Part: A** Write a C Program to implement: 1. FIFO Page Replacement algorithm **Part: B** Write a C Program to implement: 1. Optimal Page Replacement algorithm **Part: C** Write a C Program to implement: 1. LRU Page Replacement algorithm