



NAME OF THE PROGRAM: CSE	DEGREE: B.Tech
COURSE NAME: Operating System Lab	SEMESTER: 5th
COURSE CODE: PCC-CS592	COURSE CREDIT: 2
COURSE TYPE: LAB	CONTACT HOURS: 4P

Exp. No.	List of Experiments	Week
1.	a) Write Shell script to find out Factorial of a given number. b) Write Shell script to determine a given year is leap year or not. c) Write Shell script to find out sum of digits of a given number. d) Write Shell script to generate Fibonacci series up to N-th term. e) Write a shell script to compute the Common Logarithm of an integer N. The value of N should be taken as command-line argument. Note: the resulting value should not contain more than two digits after the decimal point.	Week 1
2.	a) Write a shell script for Summation of N natural numbers, where the values of N natural numbers are given as command line arguments. b) Write a shell script that sorts an array of integer using any well- known sorting algorithm c) Write a shell script to check an input string is a valid user or not	Week 2
3.	a) Write a menu driven script to do the following: i) Show the permissions of a file or directory; ii) Show the number of files and directories under current directory separately; iii) Show the last modification time of a file in current directory; iv) Exit from menu. b) Write a menu driven shell script with three options: i) Accept your personal details along with a password and store them in a data file (e.g. personal.dat); ii) Display the personal details stored in the data file after successfully matching the password stored in the file; iii) Exit from menu. Hint: Use ls, wc, grep, cut, pipe () and redirection (<, >, >>).	Week 3
4.	a) Create a student file (Students.dat) containing four fields, Name, Roll, Phone and Address. The field separator is ' '. Now, write a menu driven script to do the following: (i) Sort the records in reverse order of Roll number and store the output in a new file (Sorted.dat). (ii) Replace lower case letters with the upper case letters in the file 'Sorted.dat'. Take file input using redirection. (iii) Write commands to find out the records with unique name in the file 'Sorted.dat'. Append the output records to 'Students.dat'. (iv) Display only the 2nd and 3rd lines from the above file (Students.dat). b) Create a marks file (Marks.dat) containing five fields, Name, Roll, Subject_1, Subject_2 and Subject_3. The field separator is '~'. Also, write a shell script to find the name of the student who has got the highest marks in total. The file name should be supplied as command line argument. Also, check for sufficient number of command line arguments passed and show appropriate error message.	Week 4



Exp. No.	List of Experiments	Week
5.	a) Write a C program to create a child process and print the PID & PPID from both the parent and child processes. Also verify the output of your program with the 'ps' command. b) Write a C program to create an Orphan process and verify the output of your program with the 'ps' command. c) Write a C program to create a Zombie process and verify the output of your program with the 'ps' command.	Week 5
6.	a) Implement IPC between parent and child process where parent will print a message received from the child, who will take the message as user input. Use unnamed pipe for IPC. b) Implement IPC between two processes where process-1 will take two strings as user input and send them to process-2. Process-2 will compare them and print the result (SAME OR NOT SAME). Use FIFO for IPC.	Week 6
7.	a) Write a Program to demonstrate the use of signal. The process will print a message infinitely until an interrupt signal occurs. It will handle the signal and will print a message along with the signal number that it has got. b) Write a program to demonstrate the use of signal. Parent process will stop until an alarm received from child process. c) Write a Program to create a child process. The parent will send a signal to the child every 5 seconds and the child will handle the signal and check if an input number is a leap year or not.	Week 7
8.	Write a 'C' program in LINUX to create a thread that determines the summation of N natural numbers using POSIX thread.	Week 8
9.	a) WAP using semaphore which two process will synchronize each other to print baabbaabbaab pattern. b) WAP where a newly created thread will check and number is prime or not, which number is entered in main thread.	Week 9
10.	Simulate Producer-Consumer Problem using multi-threading and Semaphore.	Week 10
11.	Take one string as input from user in process P1 and store it in a shared memory. Process P2 will read the string from the shared memory and will check if it is a PALINDROME or not and return 1 (for Palindrome) or 0 (for NOT Palindrome) back to P1, which will print the result.	Week 11
12.	a) Simulate the Reader-Writer problem using multiple processes. b) Simulate the Dining Philosopher problem using multiple processes.	Week 12