**LINUX Programming Lab Programs**

**PART – I : *Introduction to Linux/Unix commands (utilities)***

1. **General Purpose commands:**

date, who, who am I, uname, cal, tty, stty, echo, printf, bc, script, passwd ,finger

1. **File Handling utilities:**

 **directory related commands** : pwd,mkdir,cd,rmdir,ls

 **File related commands**: cat, cp, mv, rm, chmod, chown, chgrp, file, find, ln, ulink, ulimit,umask,touch

1. **Process Related Commands:**

ps, kill, nohup, at, batch, crontab, fg, bg, jobs

 **4.Filters:**

cat, head, tail, cut, paste, cmp, comm, diff, sort, more, less, pg, tr, uniq etc….

 **5.Network Related commands:**

telnet, ftp, rlogin, arp

 **6. Disk and backup utilites**

 **disk utilities** : df, du

 **backup utilities:** cpio ,tar

 **7. Advanced filters (grep,sed,awk)**

 **grep: ( grep,egrep,fgrep)**

1. Write a grep command that selects the lines from the file1 that have exactly three characters
2. Write a grep command that selects the lines from the file1 that have at least three characters.
3. Write a grep command that selects the lines from the file1 that have three or fewer characters
4. Write a grep command that count the number blank lines in the file1
5. Write a grep command that count the number nonblank lines in the file1
6. Write a grep command that selects the lines from the file1 that have the string UNIX.
7. Write a grep command that selects the lines from the file1 that have only the string UNIX.
8. Write a grep command that copy the file to the monitor, but delete the blank lines.
9. Write a grep command that selects the lines from the file1 that have at least two digits without any other characters in between
10. Write a grep command that selects the lines from the file1 that do not start with A to G

 **sed:**

1. Write a sed command that print lines numbersof lines beginning with “O”
2. Write a sed command that delete digits in the given input file.
3. Write a sed command that delete lines that contain both **BEGIN** and **END**
4. Write a sed command that delete lines that contain **BEGIN** but not **END**
5. Write a sed command that deletes the first character in each line in a file
6. Write a sed command that deletes the last character in each line in a file

g) Write a sed command to delete character before last character in each line in a file

h)Write a sed command that swaps the first and second character in each line in the file

i)Write a sed command that swaps the first and second word in each line in the file

**awk**

**1.Write** an awk command to print the lines and line number in the given input file

2. Write an awk command to print first field and second field only if third field value is >=50 in the given input file. (input field separator is “:” and output field separator is “,”)

**3 .** Consider the marks.txt is a file that contains one record per line( comma separate fields) of the student data in the form of studentid, student name, Telugu marks, English marks, Maths Marks, Science marks, Social Marks. Write an awk script to generate result for every students in the form of studentid, studentname, Total Marks and result. Result is PASS if marks is >=30 in TELUGU and English, and if marks>=40 in other subjects. Result is fail otherwise.

4 Write an awk program to print the fields 1 and 4 of a file that is passed as command line argument. The file contains lines of information that is separated by “,” as delimeter. The awk program must print at the end the average of all 4th field data.

5. Write an awk program to demonstrate user defined functions and system command.

6. Write an awk script to count the number of lines in a file that do not contain vowels.

7. Write an awk script to find the number of characters, words and lines in a file.

**PART – II : SHELL PROGRAMMING**

1. Write shell script to perform integer arithmetic operations
2. Write a shell script to perform floating point arithmetic operations using command line arguments
3. Write a shell script to check the given file is writable or not
4. Write a shell program to find out reverse string of the given string and check the given string is palindrome or not
5. Write a shell program to find out factorial of the given number
6. Write a shell script to find out whether the given number is prime number or not
7. Write a shell script to accept two file names and check if both exists. If the second filename exists, then the contents of the first filename should be appended to it. If the second file name does not exist then create a new file with the contents of the first file.
8. Write a shell script that computes the gross salary of a employee according to the following

1) if basic salary is <1500 then HRA 10% of the basic and DA =90% of the basic

2) if basic salary is >=1500 then HRA 500 and DA =98% of the basic

The basic salary is entered interactively through the key board

1. . Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
2. . Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.
3. Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
4. Write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.
5. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.
6. . Write a shell script to list all of the directory files in a directory
7. Write a shell script to implement menu driven program to display list of users who are currently working in the system, copying files (cp command), rename a file, list of files in the directory and quit option.( Hint: use case structure)
8. Write a shell program to simulate ‘**cat’** command
9. Write a shell program to simulate ‘head**’** command
10. Write a shell script count the number of lines and number of words in the given input file without using wc command
11. Write a shell program to demonstrate user defined shell functions
12. Write a shell script to search an element in the list

**PART –III C Programs Using UNIX System calls**

1 Write a C program that makes a copy of a file using standard I/O and system calls.(using command line arguments)

2 Write a C program to implement **‘cat’** command using system calls

3 Write a C program to implement **‘ls’** command using system calls

4 Write a program that takes one or more file/directory names as command line input and reports the following information on the file. A. File type. B. Number of links. C. Time of last access.

 D. Read, Write and Execute permissions.

 **(Or)**

 Write a C program to demonstrate **stat** system call

5 Write a C program to emulate the Unix ls –l command.

6 Write a C program that demonstrates redirection of standard output to a file.Ex: ls > f1.

7 Write a C program to create a child process and allow the parent to display “parent” and the child to display “child” on the screen.

8 Write a C program to create a Zombie process.9 Write a C program that illustrates how an orphan is created.10 Write a C program that illustrates how to execute two commands concurrently with a command pipe. Ex:- ls –l | sort11 Write a C program that illustrates suspending and resuming processes using signals.12 Write a C program to implement IPC using unnamed pipes ( anonymous pipe)

13 Write a C program to implement half duplex communication between two unrelated processes using named pipe(FIFO)

14 Write a C program to implement full duplex communication between two unrelated processes using named pipe(FIFO)

15 Write the C programs to demonstrate message queue IPC

16 Write a C program that illustrates two processes communicating using shared memory.17 Write a C Program that demonstrate semaphores

18 Write client and server programs(using c) for interaction between server and client processes using TCP/IP sockets.19 Write client and server programs(using c) for interaction between server and client processes using UDP sockets.

20 Write a C program to demonstrate multi threading