**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