*LAB # 10*

interprocess communication

# ***OBJECTIVE:***

*To illustrate IPC through pipe and fork system calls*

*LAB task*

EXAMPLE PROGRAMS:

**Program 1**

#include<stdio.h>

#include<unistd.h>

#include<sys/ipc.h>

#include<sys/uio.h>

#include<sys/types.h>

#include<fcntl.h>

main()

{

int pid,pfd[2],n,a,b,c;

if(pipe(pfd)==-1)

{

printf("\nError in pipe connection\n");

exit(1);

}

pid=fork();

if(pid>0)

{

printf("\nParent Process");

printf("\n\n\tFibonacci Series");

printf("\nEnter the limit for the series:");

scanf("%d",&n);

close(pfd[0]);

write(pfd[1],&n,sizeof(n));

close(pfd[1]);

exit(0);

}

else

{

close(pfd[1]);

read(pfd[0],&n,sizeof(n));

printf("\nChild Process");

a=0;

b=1;

close(pfd[0]);

printf("\nFibonacci Series is:");

printf("\n\n%d\n%d",a,b);

while(n>2)

{

c=a+b;

printf("\n%d",c);

a=b;

b=c;

n--;

}

}

}

***Exercises:***

1. *What did you learn after running the Program 1?*

***Output:***

*After running Program 1, I learned how a parent and child process can communicate using a pipe in C. The parent takes input for the Fibonacci series limit and sends it through the pipe. The child reads this value, then calculates and displays the Fibonacci series. This program shows basic inter-process communication using fork() and pipe()What did you learn after running the Program 3?*