Assignment Report – Experiment 2

Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam
Department of Computer Science & Engineering
M.Tech CSE - III Semester (2025–26)
Course: ICS1313 – Operating System Practices Laboratory
Experiment No: 2
Name: Simiyon Vinscent Samuel L
Reg no: 3122247001062

Title

Implementing Inter-Process Communication Using Pipes with System Calls

Objective

To design and implement C programs demonstrating inter-process communication (IPC) using pipes. The tasks include:

- Generating a Fibonacci series through parent-child communication.
- Transferring file content between processes using pipes.
- Executing a command, capturing its output, and writing it to a file.
- Implementing two-way communication using two pipes.

Task 1: Fibonacci Series using Pipe

Parent sends number of terms, child receives it and prints Fibonacci series.

```c
#include<stdlib.h>
#include<stdio.h>
#include<unistd.h>
#include<fcntl.h>
void fibbo(int n)
{
    long a=-1,b=1;
    long sum=0;
    for(int i=0;i<n;i++)
    {
        printf("%ld\t",sum);
        a=b;
        b=sum;
        sum=a+b;
    }
    printf("\n");
}
int main()
{
    int fd[2],n,number;
    pid_t pid;
    if(pipe(fd)==-1)
    {
        perror("pipe creation failed!\n");
        exit(1);
    }
```

```
    pid=fork();
    if(pid>0)
    {
        printf("enter the number of terms in fibbonacci series:");
        scanf("%d",&n);
        close(fd[0]);
        write(fd[1],&n,sizeof(int));
        close(fd[1]);
    }
    else if (pid==0)
    {
        close(fd[1]);
        read(fd[0],&number,sizeof(int));
        // printf("number=%d\n",number);
        fibbo(number);
        close(fd[0]);
    }
    else
    {
        perror("fork failed!");
        exit(2);
    }
    return 0;


}
web@samsamuel:/mnt/c/Users/sam/OneDrive/one drive back up/OneDrive - SSN-
Institute/Documents/projects/os/a2$ cc fibbo.c
web@samsamuel:/mnt/c/Users/sam/OneDrive/one drive back up/OneDrive - SSN-
Institute/Documents/projects/os/a2$ ./a.out
enter the number of terms in fibbonacci series:5
0       1       1       2       3
web@samsamuel:/mnt/c/Users/sam/OneDrive/one drive back up/OneDrive - SSN-
Institute/Documents/projects/os/a2$
```

Task 2: File Transfer Using Pipe

```
#include<stdlib.h>
#include<stdio.h>
#include<unistd.h>
#include<fcntl.h>
#include<string.h>
#define SIZE 1024
int main()
{
    int fd[2],nread;
```

```c
    char buffer[SIZE];
    pid_t pid;
    ssize_t bytes_read,bytes_write;
    // FILE* in="input.txt";
    // FILE* out="output.txt";
    if(pipe(fd)==-1)
    {
        perror("pipe creation failed!\n");
        exit(1);
    }
    pid=fork();
    if(pid>0)
    {
        close(fd[0]);
        int input=open("input.txt",O_RDONLY,0644);
        while ((bytes_read = read(input, buffer, SIZE)) != 0)
        {
            write(fd[1], buffer, bytes_read);
        }
        close(fd[1]);
    }
    else if (pid==0)
    {
        close(fd[1]);
        int output=open("output.txt",O_WRONLY | O_CREAT ,0644);
        while((bytes_read=read(fd[0],buffer,SIZE))!=0)
        {
            bytes_write=write(output,buffer,bytes_read);
            if(bytes_write==-1)
            perror("write failed\n");
        }
        close(fd[0]);
    }
    else
    {
        perror("fork failed!");
        exit(2);
    }
    return 0;

}
```

```
web@samsamuel:/mnt/c/Users/sam/OneDrive/one drive back up/OneDrive - SSN-
Institute/Documents/projects/os/a2$ cc file.c
web@samsamuel:/mnt/c/Users/sam/OneDrive/one drive back up/OneDrive - SSN-
Institute/Documents/projects/os/a2$ ./a.out
web@samsamuel:/mnt/c/Users/sam/OneDrive/one drive back up/OneDrive - SSN-
Institute/Documents/projects/os/a2$ cat {input,output}.txt
hi i am simiyon
hi i am simiyon
web@samsamuel:/mnt/c/Users/sam/OneDrive/one drive back up/OneDrive - SSN-
Institute/Documents/projects/os/a2$
```

Task 3: Command Execution and Output to File

```c
#include<stdlib.h>
#include<stdio.h>
#include<unistd.h>
#include<fcntl.h>
#include<string.h>
#include<sys/wait.h>
#define SIZE 1024
int main()
{
    int fd[2];
    char buffer[SIZE];
    pid_t pid;
    ssize_t bytes_read,bytes_write;
    // FILE* in="input.txt";
    // FILE* out="output.txt";
    if(pipe(fd)==-1)
    {
        perror("pipe creation failed!\n");
        exit(1);
    }
    pid=fork();
    if(pid>0)
    {
        wait(NULL);
        close(fd[1]);
        int output_fd = open("ls.txt", O_CREAT | O_WRONLY | O_TRUNC,
0644);
        if (output_fd == -1) {
            perror("File open failed");
            exit(1);
        }
        while ((bytes_read = read(fd[0], buffer, SIZE)) > 0)
```

```
        {
            bytes_write=write(output_fd, buffer, bytes_read);
            if(bytes_write==-1)
            perror("write failed\n");
        }


        int output=open("ls.txt",O_RDONLY,0644);
        if (output == -1) {
            perror("Failed to reopen file for reading");
            exit(1);
        }
        while((bytes_read=read(output,buffer,SIZE))>0)
        {
            bytes_write=write(1,buffer,bytes_read);
            if(bytes_write==-1)
            perror("write failed\n");
        }

        close(fd[0]);

    }
    else if (pid==0)
    {
        close(fd[0]);
        dup2(fd[1],STDOUT_FILENO);
        if(execl("/bin/ls","ls","-l",NULL)==-1)
        {
            perror("invalid pathway using home path");

            exit(1);
        }
        close(fd[1]);
    }
    else
    {
        perror("fork failed!");
        exit(2);
    }
    return 0;

}
web@samsamuel:/mnt/c/Users/sam/OneDrive/one drive back up/OneDrive - SSN-Institute/Documents/projects/os/a2$ cc ls.c
```

```
web@samsamuel:/mnt/c/Users/sam/OneDrive/one drive back up/OneDrive - SSN-Institute/Documents/projects/os/a2$ ./a.out

total 32

-rwxrwxrwx 1 web web 16424 Jul 21 18:40 a.out

-rwxrwxrwx 1 web web   947 Jul 15 22:36 fibbo.c

-rwxrwxrwx 1 web web  1120 Jul 15 22:36 file.c

-rwxrwxrwx 1 web web    17 Jul 21 18:39 input.txt

-rwxrwxrwx 1 web web  1698 Jul 15 23:07 ls.c

-rwxrwxrwx 1 web web   343 Jul 15 23:07 ls.txt

-rwxrwxrwx 1 web web    17 Jul 21 18:40 output.txt

web@samsamuel:/mnt/c/Users/sam/OneDrive/one drive back up/OneDrive - SSN-Institute/Documents/projects/os/a2$
```

Conclusion

This assignment provided hands-on experience with inter-process communication using pipes. It included tasks involving parent-child communication, file transfer via pipe, command execution and output capturing, and two-way communication using multiple pipes. The practical exposure reinforced understanding of process synchronization, system calls like pipe(), fork(), read(), write(), dup2(), and inter-process communication mechanisms.