IT253 - OPERATING SYSTEMS

ASSIGNMENT 3

Name: **Sachin Prasanna**
Roll No.: **211IT058**

1)

Results:

Server side:

```
student@itadmin-HP-ProDesk-600-G5-MT:~/Desktop/OS Labs/oslab_3/exercise1$ gcc fifoserver.c -o ser
student@itadmin-HP-ProDesk-600-G5-MT:~/Desktop/OS Labs/oslab_3/exercise1$ ./ser
Received string: "hi os lab" and length is 9
Received string: "assignment 3" and length is 12
Received string: "i am sachin" and length is 11
Received string: "byebye" and length is 6
Received string: "end" and length is 3
student@itadmin-HP-ProDesk-600-G5-MT:~/Desktop/OS Labs/oslab_3/exercise1$
```

Client side:

```
student@itadmin-HP-ProDesk-600-G5-MT:~/Desktop/OS Labs/oslab_3/exercise1$ gcc fifoclient.c -o cli
student@itadmin-HP-ProDesk-600-G5-MT:~/Desktop/OS Labs/oslab_3/exercise1$ ./cli
FIFO_CLIENT: Send messages, infinitely, to end enter "end"
Enter string: hi os lab
Sent string: "hi os lab" and string length is 9Enter string: assignment 3
Sent string: "assignment 3" and string length is 12Enter string: i am sachin
Sent string: "i am sachin" and string length is 11Enter string: byebye
Sent string: "byebye" and string length is 6Enter string: end
Sent string: "end" and string length is 3
student@itadmin-HP-ProDesk-600-G5-MT:~/Desktop/OS Labs/oslab_3/exercise1$
```

Observations: These two C programs implement a simple client-server communication and is an example of one - way communication. The client asks the user to input a string, which is then sent to the server to which the server calculates the length of the inputted string and prints it on its console.

The program terminates when the client sends an "end" message, to which both client and server are shut down and the pipe is closed.

2)

Code:
Server Side:

```c
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <fcntl.h>
#include <unistd.h>
#include <string.h>
#include <ctype.h>
#define FIFO_FILE "/tmp/fifo_twoway"


int main()
{
    int fd;
    char readbuf[80];
    char end[10];
    int to_end;
    int read_bytes;
    char original[80];
    /* Create the FIFO if it does not exist */
    mkfifo(FIFO_FILE, S_IFIFO | 0640);
    strcpy(end, "end");
    fd = open(FIFO_FILE, O_RDWR);
    while (1)
    {

        read_bytes = read(fd, readbuf, sizeof(readbuf));
        readbuf[read_bytes] = '\0';
```

```c
        to_end = strcmp(readbuf, end);
        if (to_end == 0)
        {
            printf("Ending\n");
            close(fd);
            break;
        }

        strcpy(original, readbuf);

        int i = 0;
        while(readbuf[i] != '\0'){
            readbuf[i] = toupper(readbuf[i]);
            i++;
        }

        printf("Received String: \"%s\" \nCapitalised String: \"%s\"\n", original,
readbuf);
        write(fd, readbuf, strlen(readbuf));
        sleep(2);
    }
    return 0;
}
```

Client Side:

```c
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <fcntl.h>
#include <unistd.h>
#include <string.h>
#define FIFO_FILE "/tmp/fifo_twoway"
int main()
{
    int fd;
    int end_process;
```

```c
    int stringlen;
    int read_bytes;
    char original[80];
    char readbuf[80];
    char end_str[5];
    fd = open(FIFO_FILE, O_CREAT | O_RDWR);
    strcpy(end_str, "end");
    while (1)
    {
        printf("Enter string to be sent: ");
        fgets(readbuf, sizeof(readbuf), stdin);
        stringlen = strlen(readbuf);
        readbuf[stringlen - 1] = '\0';
        end_process = strcmp(readbuf, end_str);
        // printf("end_process is %d\n", end_process);
        if (end_process != 0)
        {
            write(fd, readbuf, strlen(readbuf));
            strcpy(original, readbuf);
            read_bytes = read(fd, readbuf, sizeof(readbuf));
            readbuf[read_bytes] = '\0';
            printf("Sent String: \"%s\" \nCapitalised String: \"%s\" \n", original,
                    readbuf);
        }
        else
        {
            write(fd, readbuf, strlen(readbuf));
            printf("Sent String: %s\nEnding", readbuf);
            close(fd);
            break;
        }
    }
    return 0;
}
```

Outputs:

Server:

```
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$ gcc fifoserver2.c -o ser
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$ ./ser
Recieved String: "this is nitk"
Capitalised String: "THIS IS NITK"
Recieved String: "os ASSignment"
Capitalised String: "OS ASSIGNMENT"
Ending
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$
```

Client:

```
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$ gcc fifoclient3.c -o cli
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$ ./cli
Enter string to be sent: this is nitk
Sent String: "this is nitk"
Capitalised String: "THIS IS NITK"
Enter string to be sent: os ASSignment
Sent String: "os ASSignment"
Capitalised String: "OS ASSIGNMENT"
Enter string to be sent: end
Sent String: end
Ending
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$
```

3)

Code:
Server Side:

```c
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <fcntl.h>
#include <unistd.h>
#include <string.h>
#include <stdlib.h>
#define FIFO_FILE "MYFIFO"
int main() {
    int fd;
    char read_buf[80];
    char end[10];
```

```c
    int to_end;
    int read_bytes;
    char send_buf[80];
    int send_buf_len;
    int end_process;
    char message[50];
    int i = 0;
    char ch;

    /Create the FIFO if it does not exist/
    mknod(FIFO_FILE, __S_IFIFO | 0640, 0);
    fd = open(FIFO_FILE, O_RDWR);
    read_bytes = read(fd, read_buf, sizeof(read_buf));
    read_buf[read_bytes] = '\0';
    printf("Received filename: %s\n", read_buf);
    FILE *file_ptr;
    file_ptr = fopen(read_buf, "r");
    if (file_ptr == NULL)
    {
       printf("file can't be opened \n");
       strcpy(message, "file can't be opened.\n");
       write(fd, message, strlen(message) + 1);
       close(fd);
       exit(0);
    }
    else
    {
       while ((ch = fgetc(file_ptr)) != EOF)
       {
          send_buf[i] = ch;
          i++;
       }
       send_buf_len = strlen(send_buf) + 1;
       write(fd, send_buf, send_buf_len);
    }
    fclose(file_ptr);
    return 0;
```

```
}
```

Client Side:

```c
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <fcntl.h>
#include <unistd.h>
#include <string.h>
#define FIFO_FILE "MYFIFO"
int main()
{
    int fd;
    int end_process;
    int input_len;
    char read_buf[80];
    char end_str[5];
    int fd1;
    int read_bytes;
    char received_buf[80];

    fd = open(FIFO_FILE, O_CREAT | O_RDWR);
    printf("Enter name of file:");
    fgets(read_buf, sizeof(read_buf), stdin);
    input_len = strlen(read_buf);
    read_buf[input_len - 1] = '\0';
    write(fd, read_buf, strlen(read_buf));
    printf("Received Content after file read: \"%s\"\n", read_buf);
    read_bytes = read(fd, received_buf, sizeof(received_buf));
    received_buf[read_bytes] = '\0';
    printf("Received string: %s", received_buf);
    return 0; }
```

Outputs:

Client (When File Doesn't exist):

```
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$ ./cli
Enter name of file: hey
File name Sent: hey
Recived Content after file read: file can't be opened.
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$
```

Server (When File Doesn't exist):

```
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$ gcc fifoserver2.c -o ser
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$ ./ser
Recieved filename: test.txt
file can't be opened.
```

Client (File does exist):

```
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$ gcc fifoclient2.c -o cli
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$ ./cli
Enter name of file: test.txt
File Name Sent: test.txt
Recieved Content after file read: this assignment is on fifo pipes.
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$
```

Server (File does exist):

```
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$ gcc fifoserver2.c -o ser
sachinprasanna@LAPTOP-740CVK81:/mnt/c/Users/91900/Desktop/OSLAB$ ./ser
Recieved filename: test.txt
```

Text file

```
≡ test.txt
  1    this assignment is on fifo pipes.
```

****