Assignment 7

7a - Write Read

Code

7a_writer.c

```
// C program to implement one side of FIFO
// This side writes first, then reads
#include <stdio.h>
#include <string.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>

int main()
{
    int fd;

    // FIFO file path
    char *myfifo = "/tmp/myfifo";

    // Creating the named file(FIFO)
    // mkfifo(<pathname>, <permission>)
    mkfifo(myfifo, 0666);

    char arr1[180], arr2[180];
    while (1)
    {
        // Open FIFO for write only
        fd = open(myfifo, O_WRONLY);

        // Take an input arr2ing from user.
        // 80 is maximum length
        fgets(arr2, 180, stdin);

        // Write the input arr2ing on FIFO
        // and close it
        write(fd, arr2, strlen(arr2) + 1);
        close(fd);

        // Open FIFO for Read only
        fd = open(myfifo, O_RDONLY);

        // Read from FIFO
        read(fd, arr1, sizeof(arr1));

        // Print the read message
        printf("User2: %s\n", arr1);
        close(fd);
    }
    return 0;
}
```

7a_reader.c

```
// C program to implement one side of FIFO
// This side reads first, then reads
#include <stdio.h>
#include <string.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>

int main()
{
    int fd1, sen, wor;

    // FIFO file path
    char *myfifo = "/tmp/myfifo";

    // Creating the named file(FIFO)
    // mkfifo(<pathname>,<permission>)
    mkfifo(myfifo, 0666);
    char str1[180], str2[180];
    while (1)
    {
        // First open in read only and read
        fd1 = open(myfifo, O_RDONLY);
        read(fd1, str1, 180);

        // Print the read string and close
        printf("User1: %s\n", str1);
        close(fd1);

        // Counting sentences and words
        sen = 0;
        wor = 0;

        for (int i = 0; str1[i] != '\0'; i++)
        {
            if (str1[i] == ' ')
                wor++;
            if (str1[i] == '.')
                sen++;
        }

        sprintf(str2, "Words : %d Sentences : %d", wor, sen);

        // Now open in write mode and write
        // string taken from user.
        fd1 = open(myfifo, O_WRONLY);
        scanf("\n");
        write(fd1, str2, strlen(str2) + 1);
        close(fd1);
    }

    return 0;
}
```

Output

```
abhishek-jadhav@abhishek-jadhav-ubuntu:~/Codes/OS Assignments/33232$ ./a.out
write first then read

abhishek-jadhav@abhishek-jadhav-ubuntu:~/Codes/OS Assignments/33232$ ./b.out
User1: write first then read
```

7b - Shared Memory

Code

`7b_client.c`

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/shm.h>
#include <string.h>
int main()
{
        int i;
        void *shared_memory;
        char buff[100];
        int shmid;
        shmid = shmget((key_t)2345, 1024, 0666);
        printf("Key: %d\n", shmid);
        shared_memory = shmat(shmid, NULL, 0);
        printf("Process attached at %p\n", shared_memory);
        printf("Data read from shared memory is : %s\n", (char *)shared_memory);
}
```

`7b_server.c`

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/shm.h>
#include <string.h>
int main()
{
    int i;
    void *shared_memory;
    char buff[100];
    int shmid;
    shmid = shmget((key_t)2345, 1024, 0666 | IPC_CREAT);

    printf("Key of shared memory is %d\n", shmid);
    shared_memory = shmat(shmid, NULL, 0);
    printf("Process attached at %p\n", shared_memory);
    printf("Enter some data to write to shared memory\n");
    read(0, buff, 100);
    strcpy(shared_memory, buff);
    printf("You wrote : %s\n", (char *)shared_memory);
}
```

Output

```
abhishek-jadhav@abhishek-jadhav-ubuntu:~/Codes/OS Assignments/33232$ ./b.out
Key of shared memory is 10
Process attached at 0x7fd268b74000
Enter some data to write to shared memory
Data shared with a
You wrote : Data shared with a


abhishek-jadhav@abhishek-jadhav-ubuntu:~/Codes/OS Assignments/33232$ ./a.out
Key: 10
Process attached at 0x7f3795a0c000
Data read from shared memory is : Data shared with a
```