

Exp – 7)

AIM: to write a c program to do inter process communication (IPC) using shared memory between sender process and receiver process.

PROGRAM:

sender.c

```
#include <stdio.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <unistd.h>
#include <string.h>

#define SHM_SIZE 1024 |

int main() {

    key_t key = ftok("shmfile", 65);
    int shmid = shmget(key, SHM_SIZE, 0666 | IPC_CREAT);
    if (shmid == -1) {
        perror("shmget failed");
        return 1;
    }
    char *str = (char *) shmat(shmid, NULL, 0);
    if (str == (char *) -1) {
        perror("shmat failed");
        return 1;
    }
    sprintf(str, "Hello from the sender process!");
    printf("Sender: Data written to shared memory.\n");
    sleep(5);
    shmdt(str);

    return 0;
}
```

## reciever.c

```
#include <stdio.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <unistd.h>

#define SHM_SIZE 1024

int main() {

    key_t key = ftok("shmfile", 65);
    int shmid = shmget(key, SHM_SIZE, 0666);
    if (shmid == -1) {
        perror("shmget failed");
        return 1;
    }

    char *str = (char *) shmat(shmid, NULL, 0);
    if (str == (char *) -1) {
        perror("shmat failed");
        return 1;
    }
    printf("Receiver: Data from shared memory: \"%s\"\n", str);
    shmdt(str);
    shmctl(shmid, IPC_RMID, NULL);

    return 0;
}
```

## OUTPUT:

```
jagadesh@LAPTOP-33VRBQ67:/mnt/c/Users/Parthiban/OS Exps/shell/C programs$ ./sender
Sender: Data written to shared memory.
```

```
jagadesh@LAPTOP-33VRBQ67:/mnt/c/Users/Parthiban/OS Exps/shell/C programs$ ./reciever
Receiver: Data from shared memory: "Hello from the sender process!"
```