

# OS LAB MANUAL

230701263

(CS23431)

Lab:3

Roll No:230701263

EX.NO:7

## IPC USING SHARED MEMORY

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 <string.h>
#include <unistd.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) {

} char *str = (char *)shmat(shmid, NULL, 0);
    if (str == (char *)-1) { perror("shmat
        failed"); return 1; }
    printf("Writing to shared
        memory...\n"); sprintf(str,
```

```

        "hello,what is your name");
        sleep(5); shmdt(str);
    printf("Message sent successfully!\n");
    return 0;
}

```

### Receiver.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("Message Received: %s\n", str);
    shmdt(str);

    shmctl(shmid, IPC_RMID, NULL);
    return 0;
}

```

Output for receiver.c

Input:

```

Writing to shared memory...
Message sent successfully!

```

Output for receiver.c

```
Message Received: Hello from Sender!
```