Ex. No. : 7 **Date : 28.03.2025**

Register No. : 230701385 **Name : VISHWAK S**

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.

Algorithm:

sender

1. Set the size of the shared memory segment
2. Allocate the shared memory segment using shmget
3. Attach the shared memory segment using shmat
4. Write a string to the shared memory segment using sprintf
5. Set delay using sleep
6. Detach shared memory segment using shmdt

receiver

1. Set the size of the shared memory segment
2. Allocate the shared memory segment using shmget
3. Attach the shared memory segment using shmat
4. Print the shared memory contents sent by the sender process.

Program:

D: > OS > C IPC.c > ...

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

#define SHM_SIZE 1024

void sender();
void receiver();

int main()
{
    int choice;
    while (1)
    {
        printf("\n1. Sender\n2. Receiver\n3. Exit\nEnter choice: ");
        scanf("%d", &choice);

        switch (choice)
        {
        case 1:
            sender();
            break;
        case 2:
            receiver();
            break;
        case 3:
            printf("Exiting...\n");
            exit(0);
        default:
            printf("Invalid choice! Please enter 1, 2, or 3.\n");
        }
    }

    return 0;
}
```

```c

void sender()
{
    key_t key = ftok("shmfile", 65);
    int shmid = shmget(key, SHM_SIZE, 0666 | IPC_CREAT);
    char *shm_ptr = (char *)shmat(shmid, NULL, 0);

    printf("Enter message: ");
    getchar();
    fgets(shm_ptr, SHM_SIZE, stdin);

    printf("Message sent!\n");
    sleep(10);

    shmdt(shm_ptr);
}

void receiver()
{
    key_t key = ftok("shmfile", 65);
    int shmid = shmget(key, SHM_SIZE, 0666);
    char *shm_ptr = (char *)shmat(shmid, NULL, 0);

    printf("Received: %s", shm_ptr);

    shmdt(shm_ptr);
    shmctl(shmid, IPC_RMID, NULL);
}

```

Output:

```
1. Sender
2. Receiver
3. Exit
Enter choice: 1
Enter message: Hello, Receiver!
Message sent!

1. Sender
2. Receiver
3. Exit
Enter choice: 2
Received: Hello, Receiver!

1. Sender
2. Receiver
3. Exit
Enter choice: 3
Exiting...


=== Code Execution Successful ===
```

Result:

Hence a C program to do Inter Process Communication (IPC) using shared memory between sender process and receiver process has been executed successfully.