Ex. No.: 7

Date: 27/03/25

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.
5. Detach shared memory segment using shmdt

Program Code:

sender.c

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

```
int main()
{
    int i;
    void * shrmem;
    char buf[100];
    int shmid;
    shmid = shmget((key_t 2345, 1024, 0666 | IPC_CREAT);
    printf("Key of shared memory is %d\n", shmid);
    shrmem = shmat(shmid, NULL, 0);
    printf("Process attached at %p\n", shrmem);
    sleep(5);
    printf("Write to Shared Memory\n");
    read(0, buf, 100);
    strcpy(shrmem, buf);
    printf("User Input: %s\n", (char *) shrmem);
    shmdt(shrmem);
}
```

receiver.c

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

int main()
{
  int i;
  void * shrmem;
  char buff[100];
  int shmid;
  shmid = shmget((key_t)2345, 1024, 0666);
  printf("Key of shared memory is %d\n", shmid);
  shrmem = shmat(shmid, NULL, 0);
  printf("Process attached at %p\n", shrmem);
  printf("Data read from SharedMemory: %s\n",(char *)shrmem
  shmdt(shrmem);
}
```

Output:-

Terminal 1:-

Key of shared memory is 98316
Process attached at 0x7f2b471e8000
Enter some to Write to Shared Memory
Welcome
User Input: Welcome

Terminal 2:-

Key of shared memory is 98316
Process attached at 0x7f2b471e8000
Data Read from Shared Memory
Welcome

Sample Output

Terminal 1

```
[root@localhost student]# gcc sender.c -o sender
[root@localhost student]# ./sender
```

Terminal 2

```
[root@localhost student]# gcc receiver.c -o receiver
[root@localhost student]# ./receiver
Message Received: Welcome to Shared Memory
[root@localhost student]#
```

Result:

Hence the Interprocess Communication for Shared Memory is Implemented and Executed