Ex. No.: 7

Date: 28-03-25

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

```c
#include <stdio.h>
#include <stdlib.h>
#include <sys/ipc.h>
#include <string.h>
#include <unistd.h>
#include <sys/shm.h>
#define SHM_SIZE 1024

void sender();
void receiver();

void sender()
{
    key_t key = ftok("shmfile", 6.5);
    int shmid = shmget(key, SHM_SIZE |0666|
                        IPL_(REAT));

    char * shm_ptr = (char *) shmt (shmid, NULL=0);
    printf("Enter message:");
    get char();
    fgets(shm_ptr, SHM_SIZE, stdin);
    printf("Message sent! \n");
    sleep(10);
    shmdt(shm.ptr)
}
```

```c
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: %.5", shm_ptr);
    shmdt (shm-ptr);
    shmct (shmid, IPC_RMID, NULL);

}

int main () {
    int choice;
    while (1) {
        printf ("\n1. sender \n 2. Receiver \n
                    3. Exit \n Enter choice: ");

        scanf ("%.d", &choice);

    { case 1:
            sender ();
            break;
        case 2:
            receiver ();
            break;
        case 3:
            printf ("Exiting... \n");
            exit (0);
        default:
            printf ("Invalid choice!");
        }
    }
    return 0;
}
```

51

Output:

1. sender
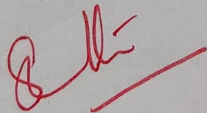2. receiver
3. exit

Enter choice: 1

   Enter message : Hi, receiver!
   Message sent!

1. Sender
2. Receiver
3. Exit

Enter choice : 2
Received: Hi, receiver!

Result:

Hence the c program to do IPC using shared memory between sender and receiver has been successfully executed.

52