Ex. No.: 7

Date: 02-04-25

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 shinat
- 4. Print the shared memory contents sent by the sender
- 5. Detach shared memory segment using shmdt

Program Code:

sender.c

)

5

)

)

3

5

5

Henclude < stdis. n)

Hinclude < stdis. h)

Hinclude < stys/ipc. h)

Hinclude < sys/shm. h)

Hinclude < string. h)

Hindude < string. h)

Hadjine SHH_SIZE 1024

```
int main ()
    huy-t key=1234;
    int shirid;
     char +shin;
    should = shonget (lay, SHM_SIZE, D6661PC-CREAT);
    ill should 20) ¿
        person ("shinget failed");
        return 1;
     shin = (char *) shinat (shinid, NVLL, 0);
     9(shim == (char*)-1)d
           person ("ahmat failed");
           suturn 1;
     sprintfe shm, "Hello world");
     firmtfe" Keisered mensage: "63\n", shm);
      shindt(shm);
      situen 0;
 receiver.c
tim clude (stdiw. h)
#include < stdib. h)
Hinclude < sys/ ju. h)
Dinclude < sys 1 shur.h)
Hinclude < string. h)
```

Holefine SHM-SIZE 1024

```
int main () 1
    lay-t bey=1224;
    mt should ;
    char * shm;
    should = shonget (hut, SHU_SIZE, Ob66)
    41shmid ==-1)d
         person C'ahmout failed's
         situmi;
     Ahm = (chart) ahmat (shind, NULL, 0);
     2 (shim = 2 (char*) -1)1
         person (" showat failed")
         return 1;
     printpl''Received minage: 1.5 m", shm);
     shmat(shm)?
     rutum 0;
 9
Dulput:
    gec - o undaripe . c
    gec - o receiver épots c
   1/sender
   Received murrage: Hello World
  -/ receiver
  Received musage: Hello world
```

3

3

3

3

3

3

U

0

0

V

0

O

0

3

1

3

13

2

13

3

3

3

3

0

3

5

0

5

)

9

9

Sample Output

Terminal 1

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

Terminal 2

1

)

)

)

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

t: A C program a executed for interprocess Communication using should numbry.