**IPC USING SHARED MEMORY**

**Ex.No:7**

# AIM:

To write a c program to implement IPC using shared memory.

# ALGORITHM:

Step 1: Start the process

Step 2: Declare the segment size Step 3: Create the shared memory

Step 4: Read the data from the shared memory Step 5: Write the data to the shared memory Step 6: Edit the data

Step 7: Stop the process

# PROGRAM:

#include<stdio.h> #include<stdlib.h> #include<unistd.h> #include<string.h> #include<sys/ipc.h> #include<sys/shm.h> #include<sys/types.h> #define SEGSIZE 100

int main(int argc, char \*argv[ ])

{

int shmid,cntr; key\_t key; char \*segptr;

char buff[]="poooda ";

key=ftok(".",'s');

if((shmid=shmget(key, SEGSIZE, IPC\_CREAT | IPC\_EXCL | 0666))== -1)

{

if((shmid=shmget(key,SEGSIZE,0))==-1)

{

perror("shmget"); exit(1);

}

}

else

{

printf("Creating a new shared memory seg \n"); printf("SHMID:%d",shmid);

}

system("ipcs –m"); if((segptr=(char\*)shmat(shmid,0,0))==(char\*)-1)

{

perror("shmat"); exit(1);

}

printf("Writing data to shared memory…\n"); strcpy(segptr,buff);

printf("DONE\n");

printf("Reading data from shared memory…\n"); printf("DATA:-%s\n",segptr);

printf("DONE\n");

printf("Removing shared memory Segment…\n"); if(shmctl(shmid,IPC\_RMID,0)== -1)

printf("Can‟t Remove Shared memory Segment…\n"); else

printf("Removed Successfully");

}