**KHUSHBU PATEL - 150034**

**IPC ASSIGNMENT**

**QUE : 1**

**CODE FOR SHARED MEMORY WRITER**

#include <stdio.h>

#include <sys/ipc.h>

#include <sys/types.h>

#include <sys/shm.h>

int main()

{key\_t key = ftok("shmfile",65);

char \*p;

char str[40];

int id;

id=shmget(key,40,IPC\_CREAT|06444);

if(id<0)

{

perror("shmget");

}

printf("Shared memory ID=%d \n",id);

p=shmat(id,0,0);

printf("Enter the data: \n");

gets(str);

printf("Process starts from... %s\n",str);

shmdt(str);

}

CODE FOR SHARED MEMORY READER

#include<stdio.h>

#include<sys/ipc.h>

#include<sys/types.h>

#include<sys/shm.h>

int main()

{

char \*r;

int id;

id=shmget(40,250,IPC\_CREAT|0644);

if(id<0)

{

perror("shmget");

return 0;

}

printf("id = %d\n",id);

r = shmat(id,0,0);

printf("data reads = = = %s",r);

//scanf("%s",p);

return 0;

}



QUE:2

#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

#include<sys/types.h>

#include<sys/stat.h>

#include<fcntl.h>

#include<string.h>

#include<sys/ipc.h>

#include<sys/msg.h>

typedef struct

{

long msgtype;

char buffer[100];

}msg;

main()

{

int qid;

int i;

msg m1,m2;

qid=msgget(32,IPC\_CREAT|0644);

m1.msgtype=10;

for(i=0;i<10;i++)

{

m1.buffer[i]='E';

m2.msgtype=20;

}

for(i=0;i<10;i++)

{

m2.buffer[i]='I';

//m2.msgtype=20;

}

i=msgsnd(qid,&m1,sizeof(msg),0);

printf("Return value of message send = %d ",i);

i=msgsnd(qid,&m2,sizeof(msg),0);

printf("Return value of message send = %d ",i);

i=msgrcv(qid,&m1,sizeof(msg),10,0);

printf("\n...RECEIVED MSG LOOKS LIKE...\n");

for(i=0;i<10;i++)

{

printf("%c ",m1.buffer[i]);

}

printf("\nmsg rec == %d\n",i);

printf("\n");

i=msgrcv(qid,&m2,sizeof(msg),20,0);

for(i=0;i<10;i++)

{

printf("%c ",m2.buffer[i]);

}

//printf("\nmsg rec == %d\n",i);

struct msqid\_ds buf;

qid=msgget(32,IPC\_CREAT | 0644);

printf("\nQUEUE ID = %d\n",qid);

msgctl(qid,IPC\_STAT,&buf);

printf("\n...STRUCTURE OF MSG QUEUE...\n");

printf("Number of msg in queue...%hi\n",buf.msg\_qnum);

printf("Maximum number of bytes in queue..%hi\n",buf.msg\_qbytes);

qid=msgget(49,IPC\_CREAT|0644);

//printf("qid = %d\n",qid);

msgctl(qid,IPC\_RMID,NULL);

printf("\n...MESSAGE REMOVED...\n");

}

Text

Description automatically generated

QUE : 3

#include<unistd.h>

#include<sys/types.h>

int fds[2];

void parent\_code()

{

char \*writebuf1="hello !!";

char \*writebuf2="chipmates";

close(fds[0]);

write(fds[1],writebuf1,10);

write(fds[1],writebuf2,10);

printf("\n...write end of pipe...\n");

}

void child\_code()

{

char readbuf[20];

int n,i;

close(fds[1]);

n=read(fds[0],readbuf,20);

printf("\n...Read end of file...\n");

printf("No of char reading from child :=%d\n",n);

for(i=n+1;i>=0;i--)

{

printf("%c",readbuf[i]);

}

}

main()

{

int res,i;

pid\_t p;

res=pipe(fds);

if(res==-1)

{

perror("pipe");

exit(1);

}

p=fork();

if(p==0)

{

child\_code();

}

else

{

parent\_code();

}

}

OUTPUT :

Text

Description automatically generated