**SSN College of Engineering, Kalavakkam**

**Department of Computer Science and Engineering**

**Semester - IV**

**UCS1411 – Operating Systems Lab**

**Academic Year: 2019-2020 Batch: 2018-2022**

**Name : Rahul Ram M**

**Class : CSE – B**

**Register Number : 185001121**

**Lab Exercise 5 InterProcess Communications using Shared Memory**

**1.Develop an application for getting a name in parent and convert it into uppercase in child**

**using shared memory.**

**CODE:**

#include <sys/ipc.h>

# define NULL 0

#include <sys/shm.h>

#include <sys/types.h>

# include<unistd.h>

# include<stdio.h>

# include<stdlib.h>

# include<string.h>

#include <sys/wait.h>

#include <stdio\_ext.h>

#include<ctype.h>

// parent writing a char in shared memory and child reads it and prints it.

int main()

{

int pid;

char \*a,\*b;

int id;

id=shmget(111,50,IPC\_CREAT | 00666);

pid=fork();

if(pid>0) //parent

{

//id=shmget(111,50,IPC\_CREAT | 00666);

char s[25];

a=shmat(id,NULL,0);

printf("Parent : \n");

printf("Enter Name : ");

scanf("%s",s);

strcpy(a,s);

wait(NULL);

shmdt(a);

}

else //child

{

sleep(5);

printf("\nChild : \n");

//id=shmget(111,50,0);

b=shmat(id,NULL,0);

for(int i=0;i<strlen(b);i++)

{

b[i]=toupper(b[i]);

}

printf("%s\n",b);

shmdt(b);

}

shmctl(id, IPC\_RMID,NULL);

}

**OUTPUT:**

rahul@rahul-HP-Notebook:~$ ./upper

Parent :

Enter Name : hi

Child :

HI

------------------------------------------------------------------------------------------------------------------

**2.Develop an client / server application for file transfer using shared memory.**

**CODE:**

**SERVER:**

#include <sys/ipc.h>

# define NULL 0

#include <sys/shm.h>

#include <sys/types.h>

# include<unistd.h>

# include<stdio.h>

# include<stdlib.h>

# include<string.h>

#include <sys/wait.h>

#include <stdio\_ext.h>

#include<ctype.h>

// parent writing a char in shared memory and child reads it and prints it.

int main()

{

int id,av;

char \*a,c,\*turn,\*avail;

id=shmget(111,50,IPC\_CREAT | 00666);

av=shmget(112,50,IPC\_CREAT | 00666);

avail=shmat(av,NULL,0);

a=shmat(id,NULL,0);

char s[100];

turn=shmat(av,NULL,0);

turn[0]=1;

printf("Server Running : ");

printf("Waiting for File Name \n");

avail[0]='N';

while(avail[0]=='N');

//turn[0]=2;

//while(turn[0]==2);

printf("File Name : %s",a);

FILE \*f1=fopen(a,"r");

if(f1!=NULL)

{

fread(a,150,1,f1);

printf("\n\nFile Contents :\n\n%s\n",a);

}

else

{

printf("\nFile Doesnt Exist !!!\n");

strcpy(a,"-1");

}

avail[0]='N';

shmdt(a);

shmdt(avail);

return 0;

}

**CLIENT:**

#include <sys/ipc.h>

# define NULL 0

#include <sys/shm.h>

#include <sys/types.h>

# include<unistd.h>

# include<stdio.h>

# include<stdlib.h>

# include<string.h>

#include <sys/wait.h>

#include <stdio\_ext.h>

#include<ctype.h>

// parent writing a char in shared memory and child reads it and prints it.

int main()

{

int id,av;

char \*file,\*b,c,\*avail;

av=shmget(112,50,0);

avail=shmat(av,NULL,0);

id=shmget(111,50,0);

b=shmat(id,NULL,0);

printf("Enter File Name : \n");

scanf("%s",file);

strcpy(b,file);

avail[0]='Y';

while(avail[0]=='Y');

if(strcmp(b,"-1")==0)

{

printf("\nFile Doesnt Exist !!!\n");

}

else

{

printf("\nEnter New File Name : \n");

scanf("%s",file);

FILE\* fptr=fopen(file,"w");

printf("\nFile Contents :\n\n%s\n",b);

fprintf(fptr,"%s",b);

}

shmdt(b);

shmdt(avail);

shmctl(id, IPC\_RMID,NULL);

shmctl(av, IPC\_RMID,NULL);

return 0;

}

**OUTPUT:**

**SERVER:**

(base) rahul@rahul-HP-Notebook:~$ ./fs

Server Running : Waiting for File Name

File Name : files.txt

File Contents :

An Operating System (OS) is an interface between a computer user and computer hardware.

**CLIENT:**

(base) rahul@rahul-HP-Notebook:~$ ./fc

Enter File Name :

files.txt

Enter New File Name :

files2.txt

File Contents :

An Operating System (OS) is an interface between a computer user and computer hardware.

------------------------------------------------------------------------------------------------------------------------

**3. Develop an client/server chat application using shared memory.**

**SERVER:**

# include <sys/ipc.h>

# define NULL 0

# include <sys/shm.h>

# include <sys/types.h>

# include <unistd.h>

# include <stdio.h>

# include <stdlib.h>

# include <string.h>

# include <sys/wait.h>

# include <stdio\_ext.h>

# include <errno.h>

# include <fcntl.h>

# include <ctype.h>

# define BUFF\_SIZE 1024

int main()

{

char \*avail,\*usr1,\*usr2;

int id,id1,id2;

id=shmget(111,50,IPC\_CREAT | 00666);

id1=shmget(112,50,IPC\_CREAT | 00666);

id2=shmget(113,50,IPC\_CREAT | 00666);

avail=shmat(id,NULL,0);

usr1=shmat(id1,NULL,0);

usr2=shmat(id2,NULL,0);

strcpy(usr1,"-1");

strcpy(usr2,"-1");

while(1){

printf("You : ");

scanf (" %[^\n]",usr1);

}

if(strcmp(usr1,"exit")==0){

avail[0]='Y';

shmdt(usr1);

shmctl(id1, IPC\_RMID,NULL);

break;

}

avail[0]='Y';

while(avail[0]=='Y');

printf("Client : %s\n",usr2);

}

}

**CLIENT:**

# include <sys/ipc.h>

# define NULL 0

# include <sys/shm.h>

# include <sys/types.h>

# include <unistd.h>

# include <stdio.h>

# include <stdlib.h>

# include <string.h>

# include <sys/wait.h>

# include <stdio\_ext.h>

# include <errno.h>

# include <fcntl.h>

# include <ctype.h>

# define BUFF\_SIZE 1024

int main()

{

char \*avail,\*usr1,\*usr2;

int id,id1,id2;

id=shmget(111,50,IPC\_CREAT | 00666);

id1=shmget(112,50,IPC\_CREAT | 00666);

id2=shmget(113,50,IPC\_CREAT | 00666);

avail=shmat(id,NULL,0);

usr1=shmat(id1,0,0);

usr2=shmat(id2,0,0);

while(1){

printf("Server : %s\n",usr1);

printf("You : ");

scanf (" %[^\n]",usr2);

//if(strcmp(usr1,"-1")!=0){

// printf("Server : %s\n",usr1);

// strcpy(usr1,"-1");

//}

if(strcmp(usr2,"exit")==0){

avail[0]='N';

shmdt(usr2);

shmctl(id2, IPC\_RMID,NULL);

break;

}

avail[0]='N';

while(avail[0]=='N');

}

shmctl(id, IPC\_RMID,NULL);

}

**OUTPUT:**

**SERVER:**

(base) rahul@rahul-HP-Notebook:~$ ./cs

You : hello

Client : hi

You : how are you?

Client : fine.how are you?

You : fine

Client : bye

You : bye

Client : exit

You : exit

**CLIENT:**

(base) rahul@rahul-HP-Notebook:~$ ./cc

Server : hello

You : hi

Server : how are you?

You : fine.how are you?

Server : fine

You : bye

Server : bye

You : exit