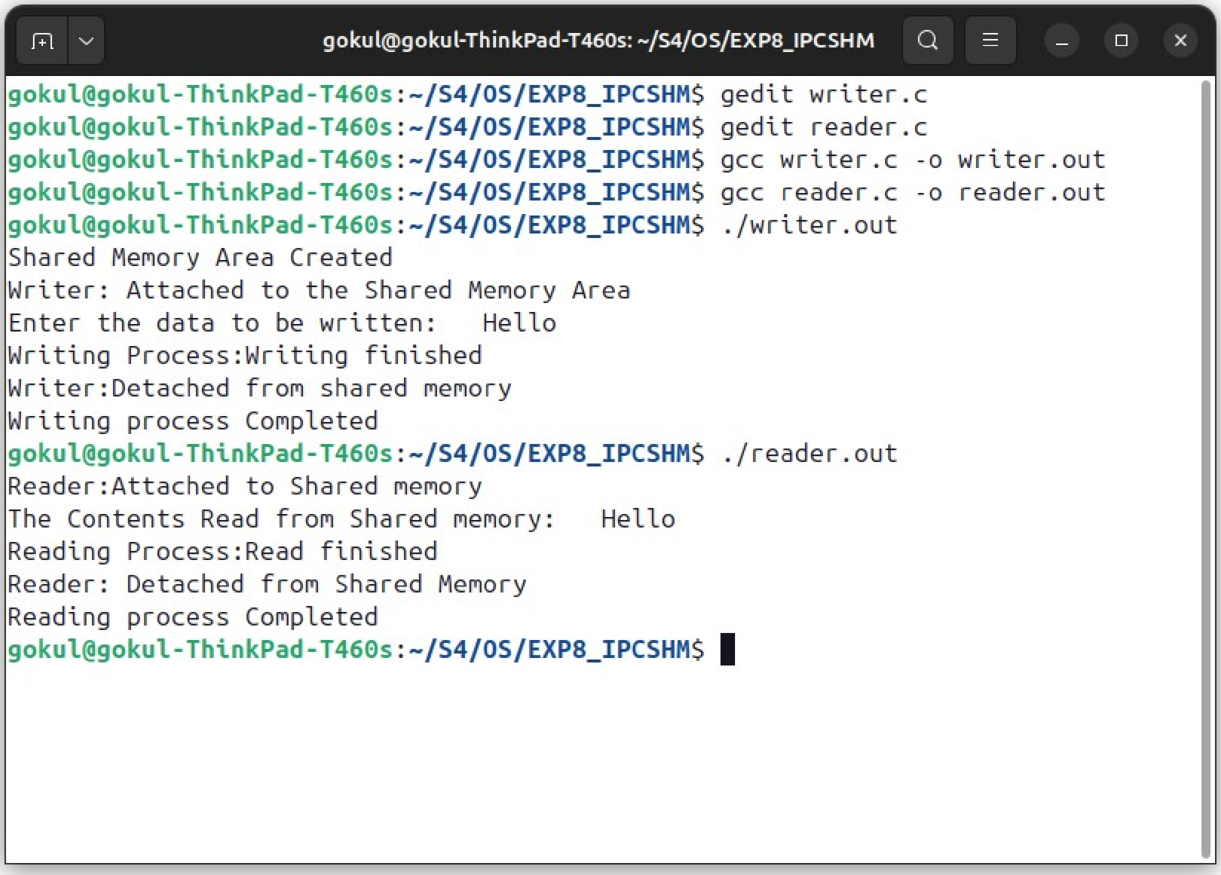
Writer process:

```
#include<stdio.h>
#include<sys/ipc.h>
#include<sys/shm.h>
#include<sys/types.h>
#include<string.h>
#include<stdlib.h>
#include<unistd.h>


int main(){
	int shid,nt,i;
	void *shmp;
	char buffer[100];

	int avail;
	shid=shmget(01234,sizeof(shmp),0644|IPC_CREAT);// create a shared memory area
	if(shid==-1){
		printf("Failed to create shared memory pool aborting process\n");
		return 1;
	}
	printf("Shared Memory Area Created\n");

	shmp=shmat(shid,NULL,0);//atatch to the shared memory
	if(shmp==(void *)-1){
		printf("Failed to attach to shared memory,process terminating\n");
		return 1;
	}
	printf("Writer: Attached to the Shared Memory Area\n");

	//writing to shared memory
	printf("Enter the data to be written:\t");
	scanf("%s",buffer);
	strcpy(shmp,buffer);

	printf("Writing Process:Writing finished\n");

	if(shmdt(shmp)==-1){//detaching from shared memory
		printf("Failed to detatch from Shared memory, process terminating\n");
		return 1;
	}
	printf("Writer:Detached from shared memory\n");

	printf("Writing process Completed\n");
	return 0;
}
```

Reader process:

```
#include<stdio.h>
#include<sys/ipc.h>
#include<sys/shm.h>
#include<sys/types.h>
#include<string.h>
#include<stdlib.h>
#include<unistd.h>

int main(){
	int shid,nt,i;
	void *shmp;
	char buffer[100];

	int avail;
	shid=shmget(01234,sizeof(shmp),0644|IPC_CREAT);// create a shared memory area
	if(shid==-1){
		printf("Failed to create shared memory pool aborting process\n");
		return 1;
	}

	shmp=shmat(shid,NULL,0);//atatch to the shared memory
	if(shmp==(void *)-1){
		printf("Failed to attach to shared memory,process terminating\n");
		return 1;
	}
	printf("Reader:Attached to Shared memory\n");

	//reading from shm

	printf("The Contents Read from Shared memory:\t");
	strcpy(buffer,shmp);
	printf("%s\n",buffer);

	printf("Reading Process:Read finished\n");


	if(shmdt(shmp)==-1){//detaching from shared memory
		printf("Failed to detatch from Shared memory, process terminating\n");
		return 1;
	}
	printf("Reader: Detached from Shared Memory\n");
```

```
        if(shmctl(shid, IPC_RMID,0)==-1){//Destroying the shared memory
                printf("Failed to Destroy shared memory, Process terminating\n");
                return 1;
        }
        printf("Reading process Completed\n");
        return 0;
}
```



```
gokul@gokul-ThinkPad-T460s:~/S4/OS/EXP8_IPCSHM$ gedit writer.c
gokul@gokul-ThinkPad-T460s:~/S4/OS/EXP8_IPCSHM$ gedit reader.c
gokul@gokul-ThinkPad-T460s:~/S4/OS/EXP8_IPCSHM$ gcc writer.c -o writer.out
gokul@gokul-ThinkPad-T460s:~/S4/OS/EXP8_IPCSHM$ gcc reader.c -o reader.out
gokul@gokul-ThinkPad-T460s:~/S4/OS/EXP8_IPCSHM$ ./writer.out
Shared Memory Area Created
Writer: Attached to the Shared Memory Area
Enter the data to be written:   Hello
Writing Process:Writing finished
Writer:Detached from shared memory
Writing process Completed
gokul@gokul-ThinkPad-T460s:~/S4/OS/EXP8_IPCSHM$ ./reader.out
Reader:Attached to Shared memory
The Contents Read from Shared memory:   Hello
Reading Process:Read finished
Reader: Detached from Shared Memory
Reading process Completed
gokul@gokul-ThinkPad-T460s:~/S4/OS/EXP8_IPCSHM$ 
```