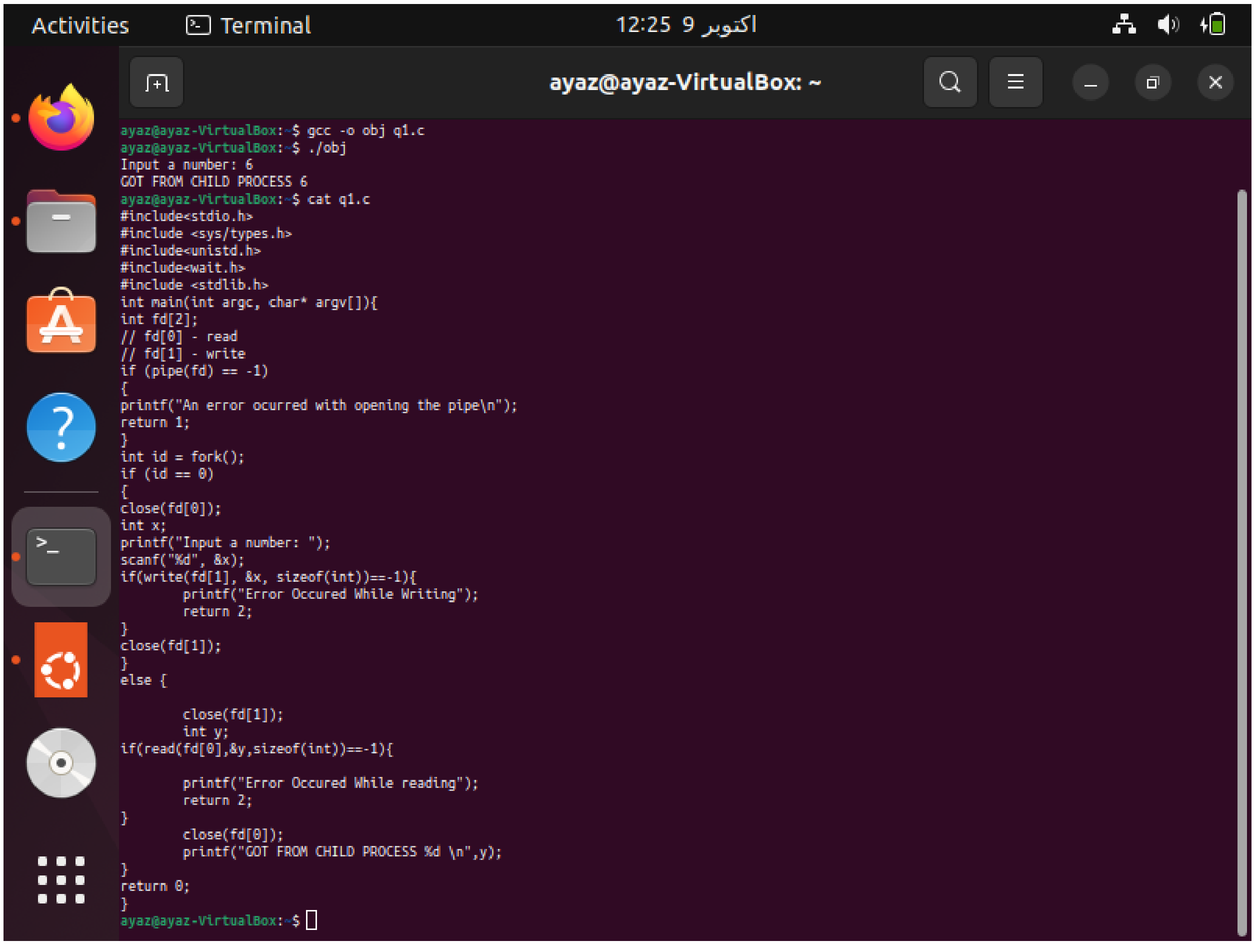
Ayaz hasan
20k-1044
SE-5A

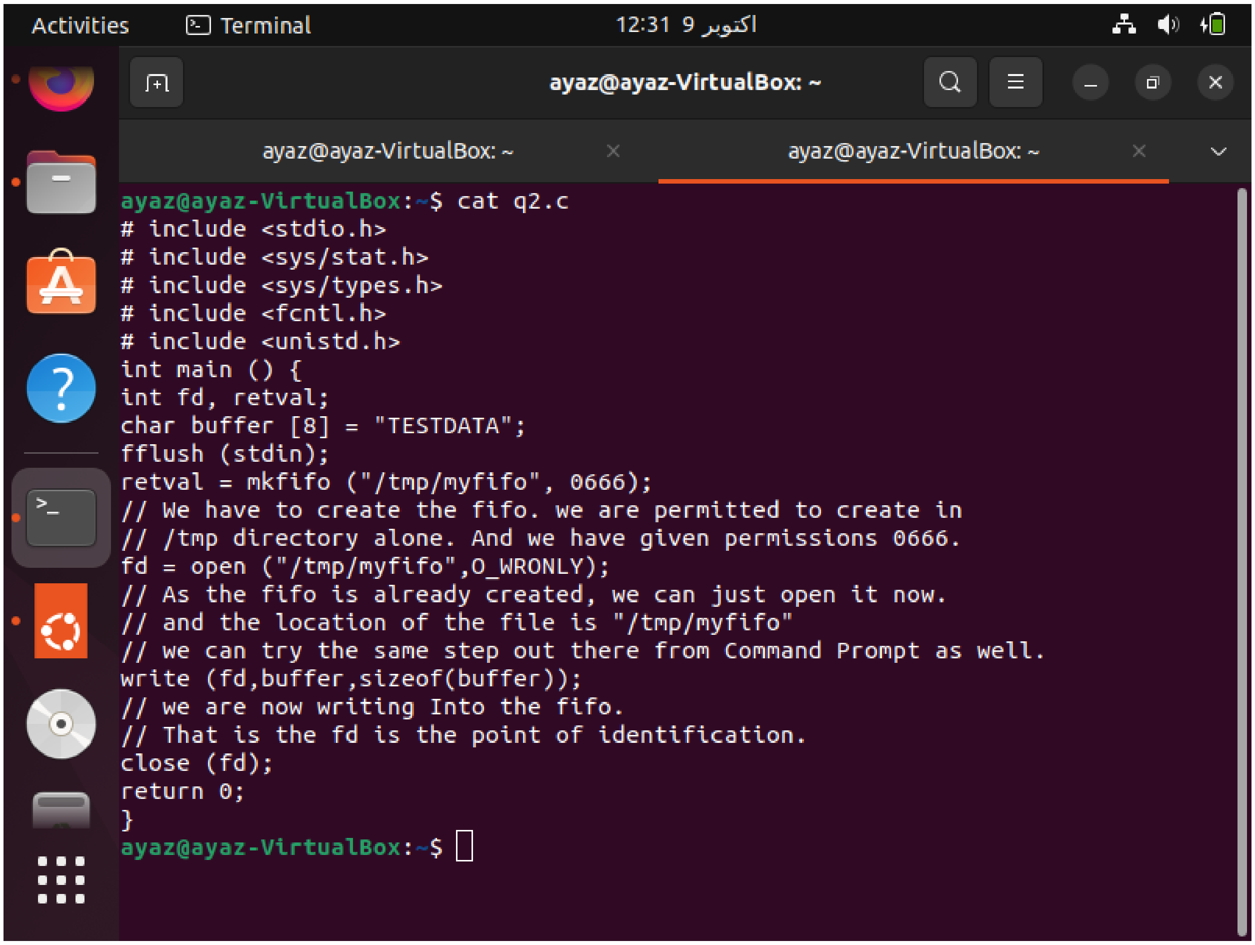
LAB 6

Task 1

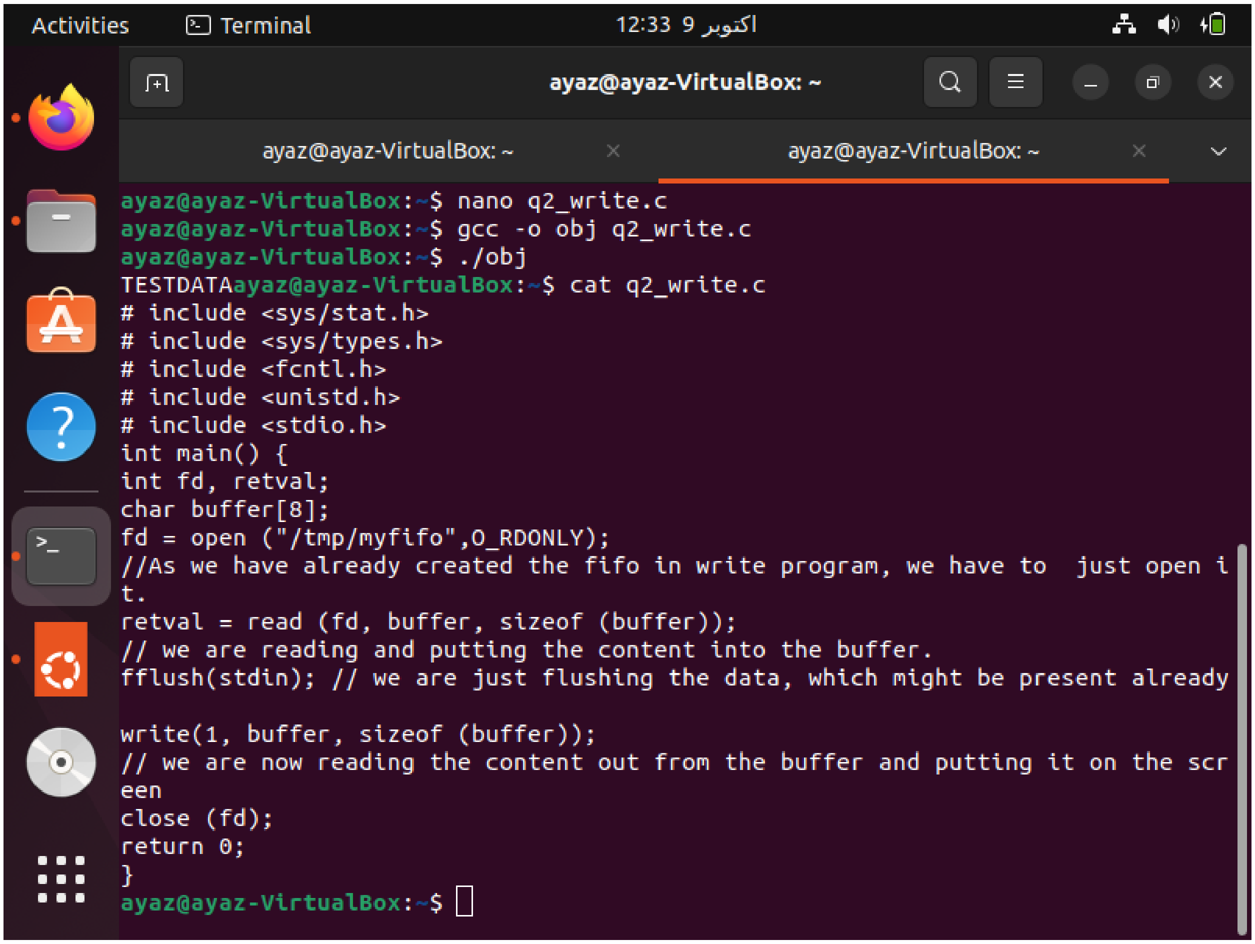


```
ayaz@ayaz-VirtualBox:~$ gcc -o obj q1.c
ayaz@ayaz-VirtualBox:~$ ./obj
Input a number: 6
GOT FROM CHILD PROCESS 6
ayaz@ayaz-VirtualBox:~$ cat q1.c
#include<stdio.h>
#include <sys/types.h>
#include<unistd.h>
#include<wait.h>
#include <stdlib.h>
int main(int argc, char* argv[]){
int fd[2];
// fd[0] - read
// fd[1] - write
if (pipe(fd) == -1)
{
printf("An error ocurred with opening the pipe\n");
return 1;
}
int id = fork();
if (id == 0)
{
close(fd[0]);
int x;
printf("Input a number: ");
scanf("%d", &x);
if(write(fd[1], &x, sizeof(int))==-1){
        printf("Error Occured While Writing");
        return 2;
}
close(fd[1]);
}
else {

        close(fd[1]);
        int y;
if(read(fd[0],&y,sizeof(int))==-1){

        printf("Error Occured While reading");
        return 2;
}
        close(fd[0]);
        printf("GOT FROM CHILD PROCESS %d \n",y);
}
return 0;
}
ayaz@ayaz-VirtualBox:~$ 
```

Task 2

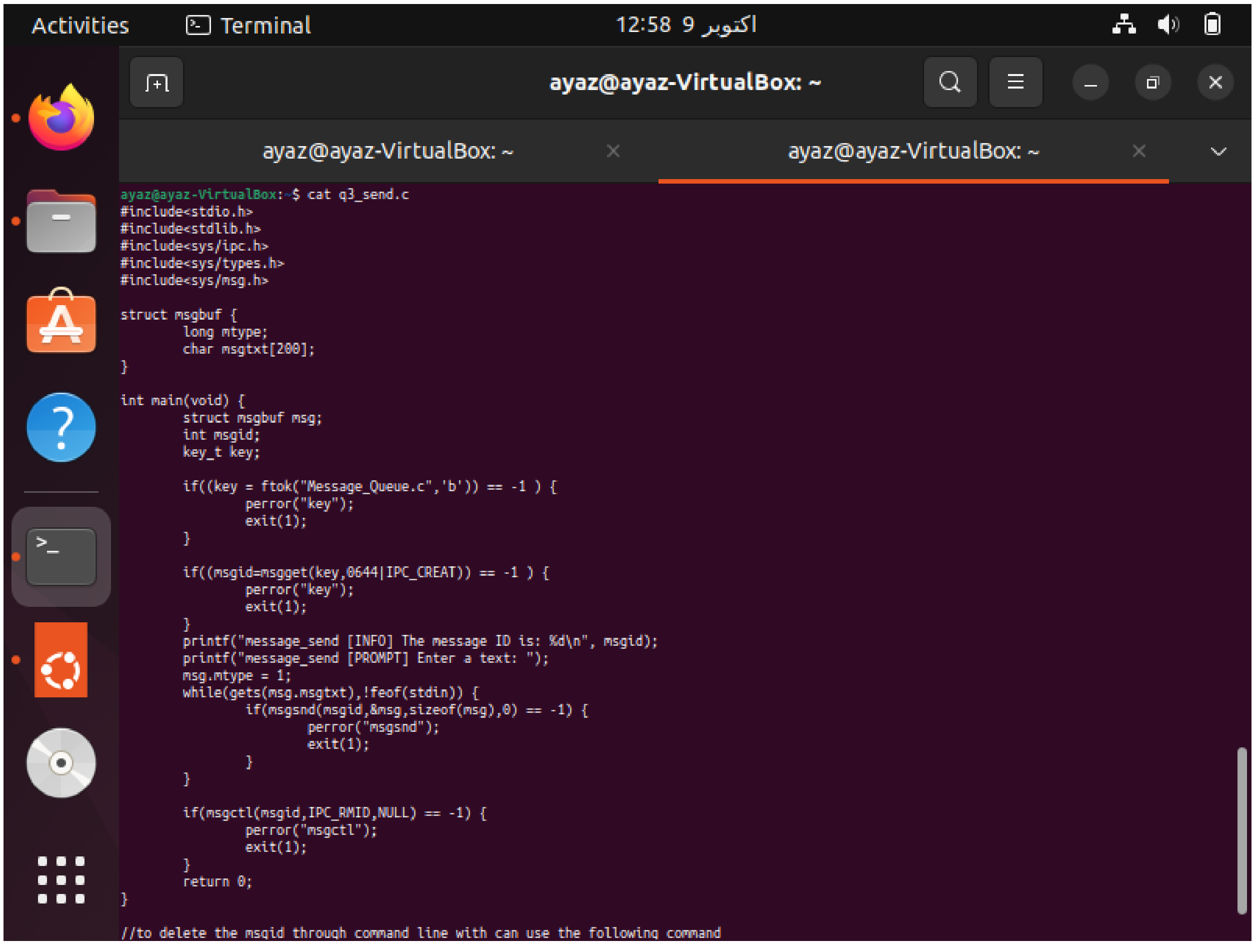


```
ayaz@ayaz-VirtualBox:~$ cat q2.c
# include <stdio.h>
# include <sys/stat.h>
# include <sys/types.h>
# include <fcntl.h>
# include <unistd.h>
int main () {
int fd, retval;
char buffer [8] = "TESTDATA";
fflush (stdin);
retval = mkfifo ("/tmp/myfifo", 0666);
// We have to create the fifo. we are permitted to create in
// /tmp directory alone. And we have given permissions 0666.
fd = open ("/tmp/myfifo",O_WRONLY);
// As the fifo is already created, we can just open it now.
// and the location of the file is "/tmp/myfifo"
// we can try the same step out there from Command Prompt as well.
write (fd,buffer,sizeof(buffer));
// we are now writing Into the fifo.
// That is the fd is the point of identification.
close (fd);
return 0;
}
ayaz@ayaz-VirtualBox:~$ 
```

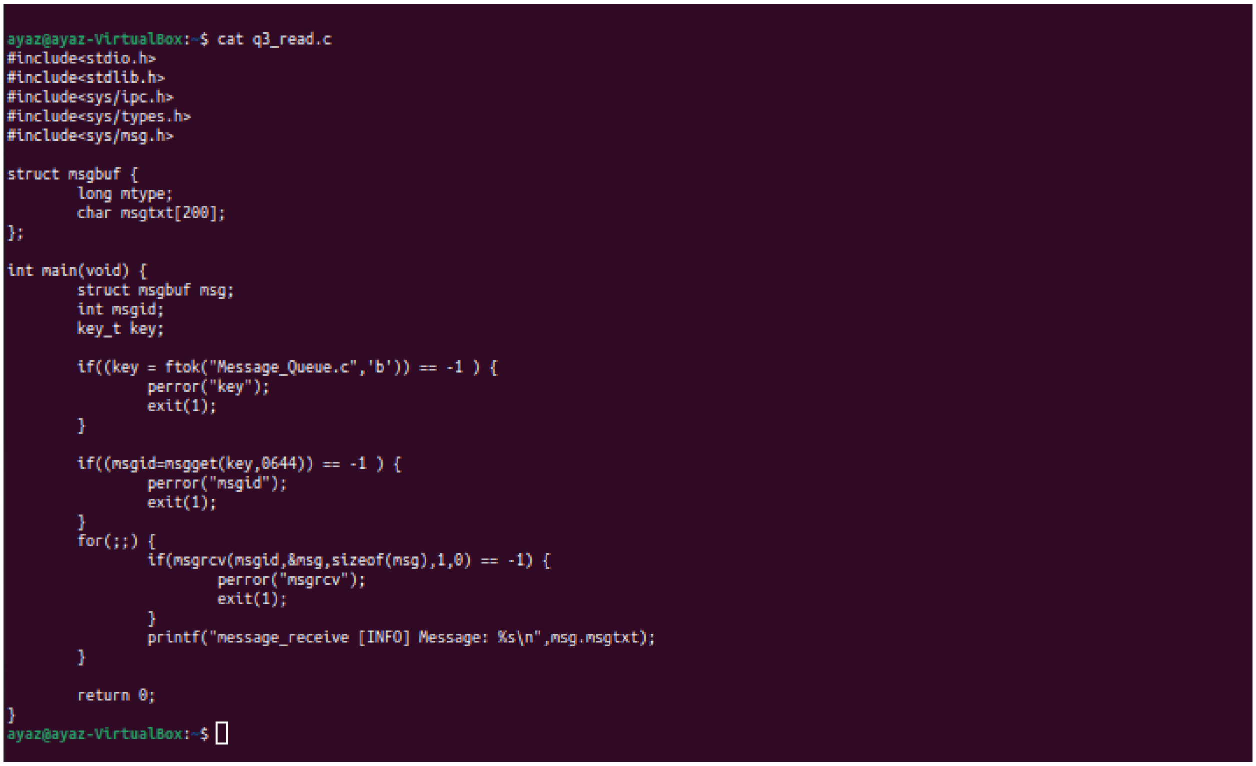


```
ayaz@ayaz-VirtualBox:~$ nano q2_write.c
ayaz@ayaz-VirtualBox:~$ gcc -o obj q2_write.c
ayaz@ayaz-VirtualBox:~$ ./obj
TESTDATAayaz@ayaz-VirtualBox:~$ cat q2_write.c
# include <sys/stat.h>
# include <sys/types.h>
# include <fcntl.h>
# include <unistd.h>
# include <stdio.h>
int main() {
int fd, retval;
char buffer[8];
fd = open ("/tmp/myfifo",O_RDONLY);
//As we have already created the fifo in write program, we have to  just open i
t.
retval = read (fd, buffer, sizeof (buffer));
// we are reading and putting the content into the buffer.
fflush(stdin); // we are just flushing the data, which might be present already

write(1, buffer, sizeof (buffer));
// we are now reading the content out from the buffer and putting it on the scr
een
close (fd);
return 0;
}
ayaz@ayaz-VirtualBox:~$ 
```

Task 3

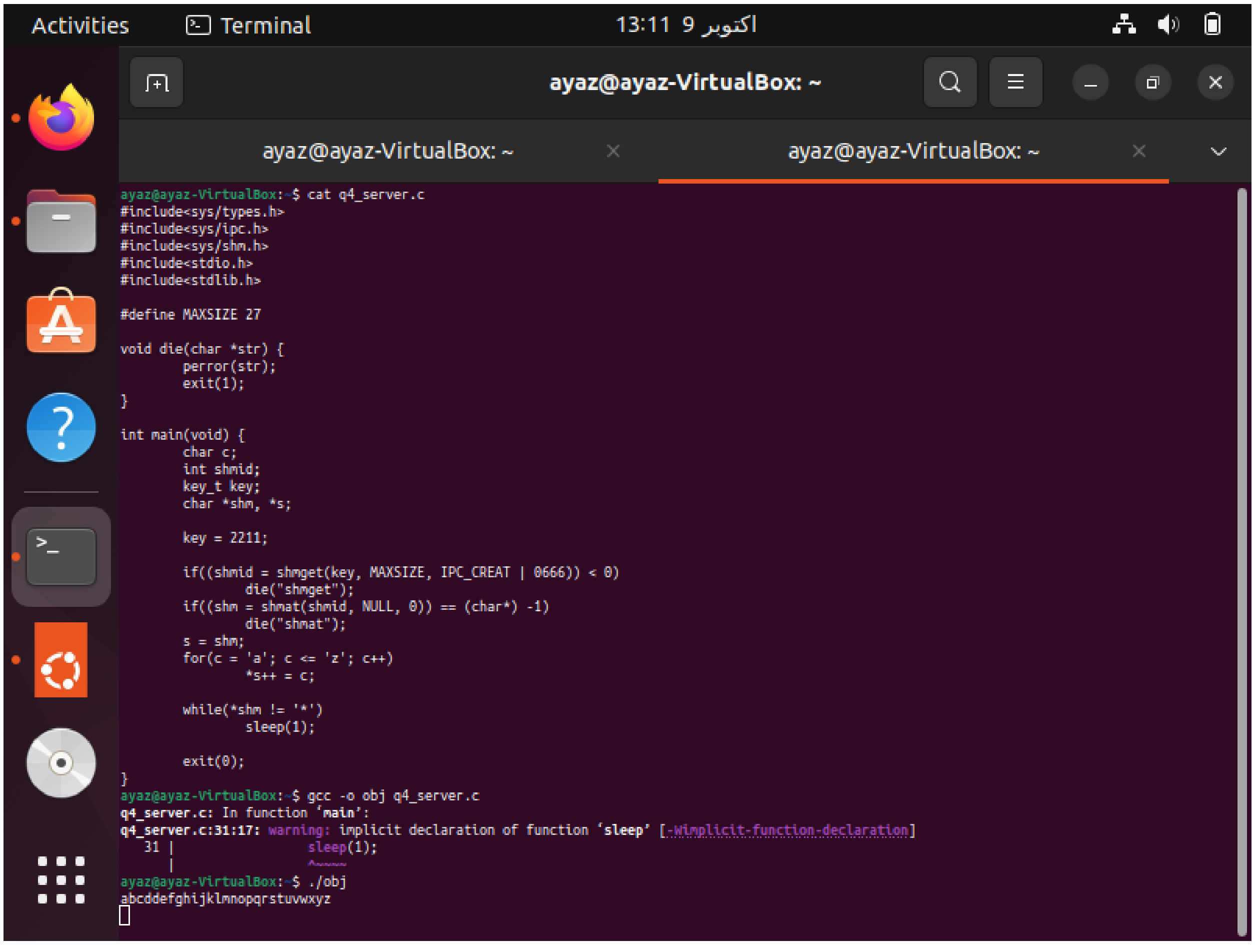


```
ayaz@ayaz-VirtualBox:~$ cat q3_send.c
#include<stdio.h>
#include<stdlib.h>
#include<sys/ipc.h>
#include<sys/types.h>
#include<sys/msg.h>

struct msgbuf {
        long mtype;
        char msgtxt[200];
}

int main(void) {
        struct msgbuf msg;
        int msgid;
        key_t key;

        if((key = ftok("Message_Queue.c",'b')) == -1 ) {
                perror("key");
                exit(1);
        }

        if((msgid=msgget(key,0644|IPC_CREAT)) == -1 ) {
                perror("key");
                exit(1);
        }
        printf("message_send [INFO] The message ID is: %d\n", msgid);
        printf("message_send [PROMPT] Enter a text: ");
        msg.mtype = 1;
        while(gets(msg.msgtxt),!feof(stdin)) {
                if(msgsnd(msgid,&msg,sizeof(msg),0) == -1) {
                        perror("msgsnd");
                        exit(1);
                }
        }

        if(msgctl(msgid,IPC_RMID,NULL) == -1) {
                perror("msgctl");
                exit(1);
        }
        return 0;
}

//to delete the msgid through command line with can use the following command
```

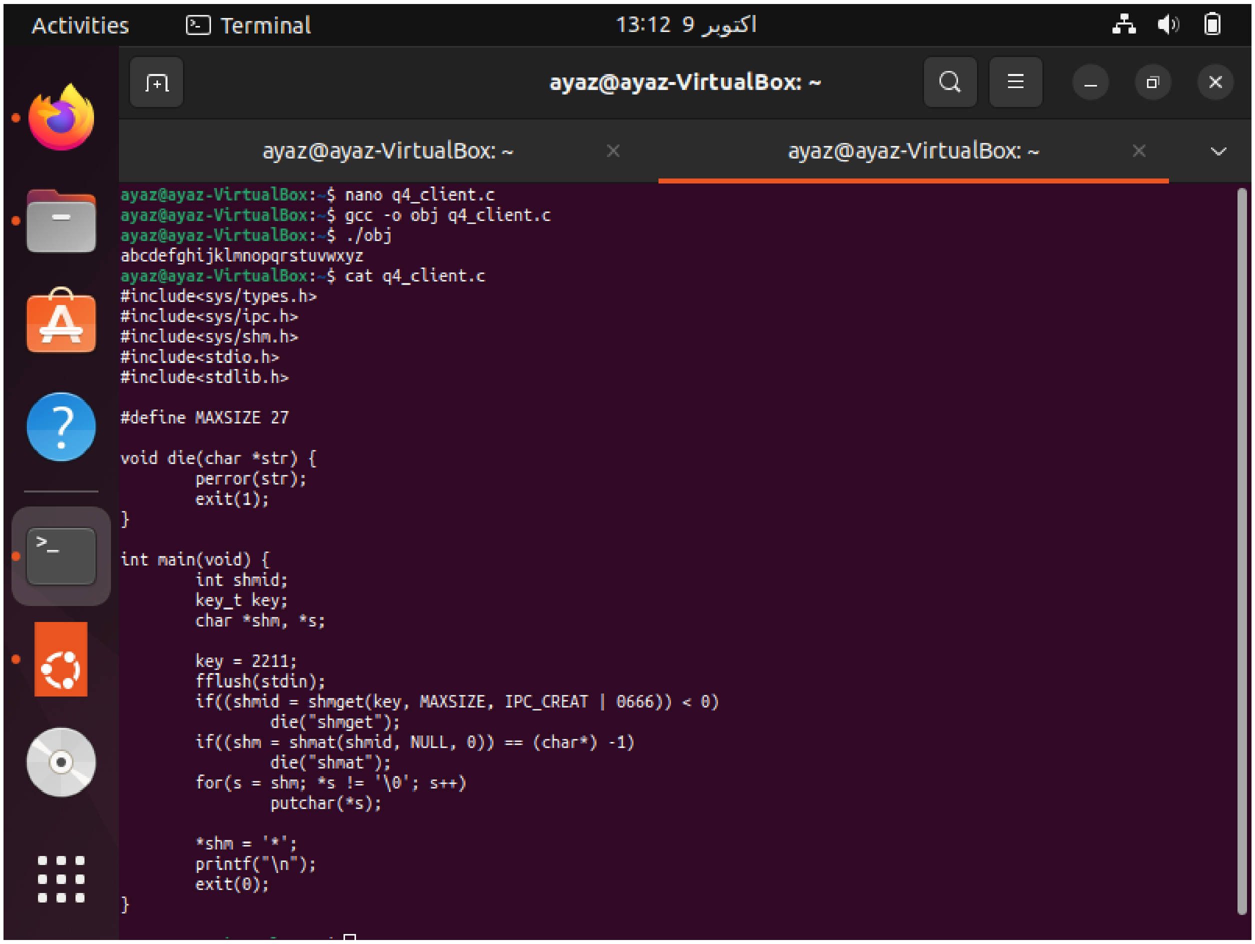


```
ayaz@ayaz-VirtualBox:~$ cat q3_read.c
#include<stdio.h>
#include<stdlib.h>
#include<sys/ipc.h>
#include<sys/types.h>
#include<sys/msg.h>

struct msgbuf {
        long mtype;
        char msgtxt[200];
};

int main(void) {
        struct msgbuf msg;
        int msgid;
        key_t key;

        if((key = ftok("Message_Queue.c",'b')) == -1 ) {
                perror("key");
                exit(1);
        }

        if((msgid=msgget(key,0644)) == -1 ) {
                perror("msgid");
                exit(1);
        }
        for(;;) {
                if(msgrcv(msgid,&msg,sizeof(msg),1,0) == -1) {
                        perror("msgrcv");
                        exit(1);
                }
                printf("message_receive [INFO] Message: %s\n",msg.msgtxt);
        }

        return 0;
}
ayaz@ayaz-VirtualBox:~$ 
```

Task 4



```
ayaz@ayaz-VirtualBox:~$ cat q4_server.c
#include<sys/types.h>
#include<sys/ipc.h>
#include<sys/shm.h>
#include<stdio.h>
#include<stdlib.h>

#define MAXSIZE 27

void die(char *str) {
        perror(str);
        exit(1);
}

int main(void) {
        char c;
        int shmid;
        key_t key;
        char *shm, *s;

        key = 2211;

        if((shmid = shmget(key, MAXSIZE, IPC_CREAT | 0666)) < 0)
                die("shmget");
        if((shm = shmat(shmid, NULL, 0)) == (char*) -1)
                die("shmat");
        s = shm;
        for(c = 'a'; c <= 'z'; c++)
                *s++ = c;

        while(*shm != '*')
                sleep(1);

        exit(0);
}
ayaz@ayaz-VirtualBox:~$ gcc -o obj q4_server.c
q4_server.c: In function ‘main’:
q4_server.c:31:17: warning: implicit declaration of function ‘sleep’ [-Wimplicit-function-declaration]
   31 |                 sleep(1);
      |                 ^~~~~
ayaz@ayaz-VirtualBox:~$ ./obj
abcddefghijklmnopqrstuvwxyz
```



```
ayaz@ayaz-VirtualBox:~$ nano q4_client.c
ayaz@ayaz-VirtualBox:~$ gcc -o obj q4_client.c
ayaz@ayaz-VirtualBox:~$ ./obj
abcdefghijklmnopqrstuvwxyz
ayaz@ayaz-VirtualBox:~$ cat q4_client.c
#include<sys/types.h>
#include<sys/ipc.h>
#include<sys/shm.h>
#include<stdio.h>
#include<stdlib.h>

#define MAXSIZE 27

void die(char *str) {
        perror(str);
        exit(1);
}

int main(void) {
        int shmid;
        key_t key;
        char *shm, *s;

        key = 2211;
        fflush(stdin);
        if((shmid = shmget(key, MAXSIZE, IPC_CREAT | 0666)) < 0)
                die("shmget");
        if((shm = shmat(shmid, NULL, 0)) == (char*) -1)
                die("shmat");
        for(s = shm; *s != '\0'; s++)
                putchar(*s);

        *shm = '*';
        printf("\n");
        exit(0);
}
```