

1,

[illegible]

2,

mycreate.c

```
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc msgcreate.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Üzenet: Szia első üzenetem D3U3EE
Üzenet: A másik üzenet pedig a Oravecz Aron
oraveczaron@oraveczaron-VirtualBox:~/Desktop$
```

msgcreate.c x

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>
#include <string.h>
#include <stdlib.h>
#define SZOVEGKULCS 654321L

struct szovegbuf {
    char sajatszoveg[1024];
    long entipus;
} sndbuf, *uzenetpont;

int main()
{
    int msgflg;
    int vissza, szovegmeret;
    int index;
    key_t kulcs;

    kulcs = SZOVEGKULCS;
    msgflg = 00666 | IPC_CREAT;
    index = msgget( kulcs, msgflg);

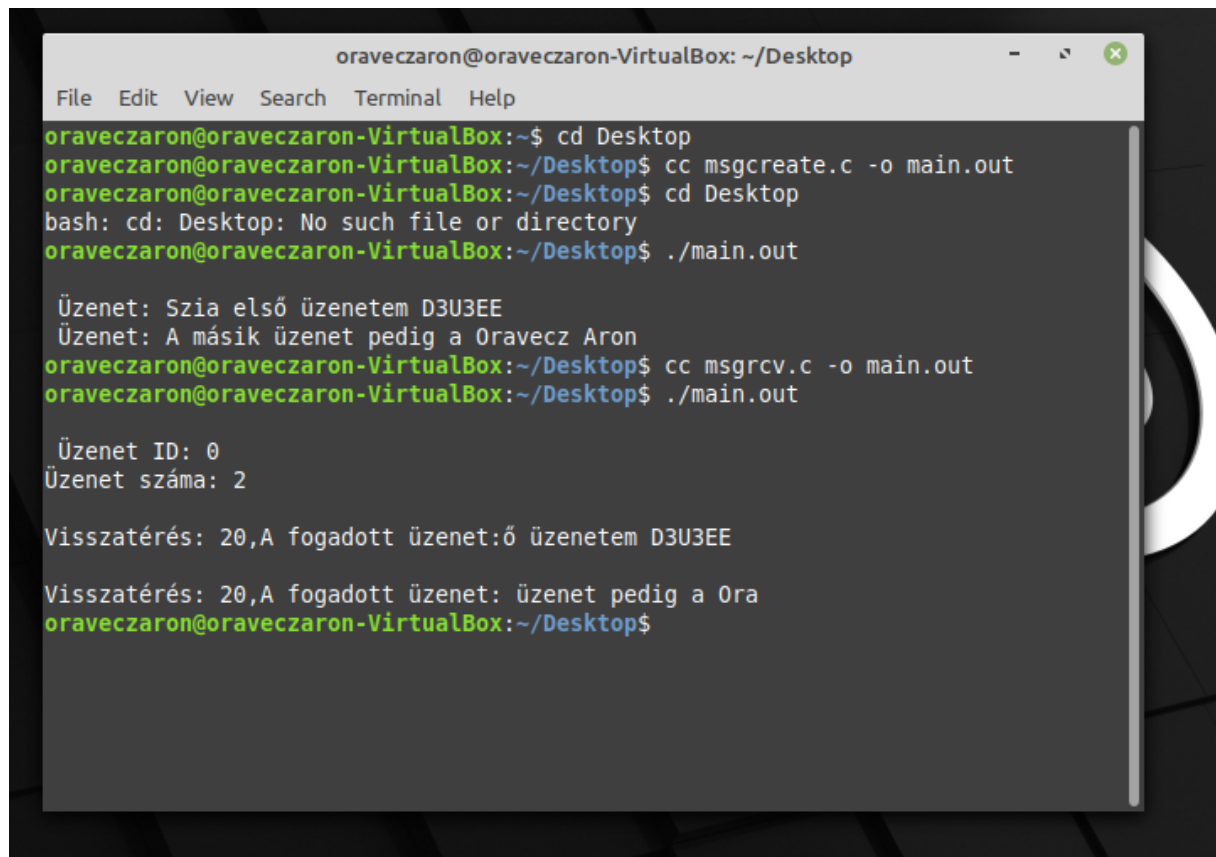
    if ( index == -1) {
        exit(-1);
        perror("Msgget nem volt sikeres a rendszerhívásban!");
    }
    uzenetpont = &sndbuf;
    uzenetpont->entipus = 1;

    strcpy(uzenetpont->sajatszoveg, "Szia első üzenetem D3U3EE");
    szovegmeret = strlen(uzenetpont->sajatszoveg) + 1;
    vissza = msgsnd(index, (struct szovegbuf *) uzenetpont, szovegmeret, msgflg);
    printf("\n Üzenet: %s", uzenetpont->sajatszoveg);

    strcpy(uzenetpont->sajatszoveg, "A másik üzenet pedig a Oravec Aron");
    szovegmeret = strlen(uzenetpont->sajatszoveg) + 1;
    vissza = msgsnd(index, (struct szovegbuf *) uzenetpont, szovegmeret, msgflg);
    printf("\n Üzenet: %s \n", uzenetpont->sajatszoveg);

    return (0);
}
```

msgrcv.c



```
oraveczaron@oraveczaron-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
oraveczaron@oraveczaron-VirtualBox:~$ cd Desktop
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc msgcreate.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cd Desktop
bash: cd: Desktop: No such file or directory
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Üzenet: Szia első üzenetem D3U3EE
Üzenet: A másik üzenet pedig a Oravecz Aron
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc msgrcv.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Üzenet ID: 0
Üzenet száma: 2

Visszatérés: 20,A fogadott üzenet:ő üzenetem D3U3EE
Visszatérés: 20,A fogadott üzenet: üzenet pedig a Ora
oraveczaron@oraveczaron-VirtualBox:~/Desktop$
```

```
msgcreate.c x msgrcv.c x msgctl.c x
#include <stdio.h>
#include <sys/types.h>
#define MSGKEY 654321L

struct msgbuf1 {
    long tipus;
    char mtext[1024];
} rcvbuf, *szovegpont;

struct msqid_ds ds, *buf;

int main()
{
    int index;
    int tipus;
    int msgflg;
    int vissza;
    int meret;

    key_t kulcs;

    kulcs = MSGKEY;
    msgflg = 00666 | IPC_CREAT | MSG_NOERROR;

    index = msgget(kulcs, msgflg);
    if ( index == -1) {
        perror("\n Msgget sikertelen volt a rendszerhívás!");
        exit(-1);
    }
    printf("\n Üzenet ID: %d",index);

    szovegpont = &rcvbuf;
    buf = &ds;
    tipus = 0;
    meret = 20;
    vissza = msgctl(index,IPC_STAT,buf);
    printf("\nÜzenet száma: %ld\n",buf->msg_qnum);

    while (buf->msg_qnum) {

        vissza = msgrcv(index,(struct msgbuf *)szovegpont, meret, tipus, msgflg);
        printf("\nVisszatérés: %d,A fogadott üzenet:%s\n",vissza, szovegpont->mtext);
        vissza = msgctl(index,IPC_STAT,buf);
    }
    return 0;
}

C Spaces: 4 Ln 38, Col 15 INS
```

msgctl.c

```
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc msgrcv.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out
```

```
Üzenet ID: 0
Üzenet száma: 2
```

```
Visszatérés: 20,A fogadott üzenet:ő üzenetem D3U3EE
```

```
Visszatérés: 20,A fogadott üzenet: üzenet pedig a Ora
```

```
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc msgctl.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out
```

```
Visszatérési érték: 0
```

```
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ █
```

```
msgctl.c x msgrcv.c x
#include <sys/ipc.h>
#include <sys/msg.h>
#include <stdlib.h>
#include <stdio.h>
#include <sys/types.h>
#define MSGKEY 654321L

int main()
{
    int index;
    int msgflg;
    int vissza;

    key_t kulcs;
    kulcs = MSGKEY;

    msgflg = 00666 | IPC_CREAT;
    index = msgget(kulcs, msgflg);

    vissza = msgctl(index, IPC_RMID, NULL);           // Törli az üzenetet
    printf("Visszatérési érték: %d\n", vissza);

    return 0;
}
```

2,

gyak10\_2.c

```
Az üzenet ID: 0

Válasszon a menüből
0. Üzenetek darabszáma
1. 1db üzenet kiolvasása
2. Üzenetsor megszüntetése
3. Kilépés
1

Kiküldött üzenet: első
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc gyak10_2.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Az üzenet ID: 0

Válasszon a menüből
0. Üzenetek darabszáma
1. 1db üzenet kiolvasása
2. Üzenetsor megszüntetése
3. Kilépés
0

Üzenetek száma: 2
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc gyak10_2.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Az üzenet ID: 0

Válasszon a menüből
0. Üzenetek darabszáma
1. 1db üzenet kiolvasása
2. Üzenetsor megszüntetése
3. Kilépés
2

Üzenetsor megszüntetve
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc gyak10_2.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Az üzenet ID: 0

Válasszon a menüből
0. Üzenetek darabszáma
1. 1db üzenet kiolvasása
2. Üzenetsor megszüntetése
3. Kilépés
3
```

```
File Edit View Search Tools Documents Help

gyak10_2.c x
#include <stdio.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>
#include <string.h>
#include <stdlib.h>
#define MSGKEY 654321L

struct msgbuf1 {
    long mtype;
    char mtext[512];
} sndbuf, *msgp;

struct msqid_ds ds, *buf;

int main()
{
    int msgid, rtn, size, flag;
    key_t key;

    key = MSGKEY;
    flag = 00666 | IPC_CREAT;
    msgid = msgget(key, flag);
    if (msgid == -1) {
        perror("\nAz msgget sikertelen!");
        return -1;
    }

    msgp = &sndbuf;
    msgp->mtype = 1;
    strcpy(msgp->mtext, "első");
    size = strlen(msgp->mtext) + 1;

    rtn = msgsnd(msgid, (struct msgbuf *) msgp, size, flag);

    strcpy(msgp->mtext, "második");
    size = strlen(msgp->mtext) + 1;
    rtn = msgsnd(msgid, (struct msgbuf *) msgp, size, flag);

    if (msgp->mtext == "exit"){
        return 0;
    }

    key = MSGKEY;
    flag = 00666 | IPC_CREAT | MSG_NOERROR;
```

## shmcreate.c illetve shmctl.c

```
oraveczaron@oraveczaron-VirtualBox: ~/Desktop
File Edit View Search Terminal Help

oraveczaron@oraveczaron-VirtualBox:~$ cd Desktop
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Az azonosító : 32811
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc shmcreate.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Az azonosító : 32811
oraveczaron@oraveczaron-VirtualBox:~/Desktop$
```

```
File Edit View Search Tools Documents Help

shmcreate.c x  shmctl.c x
#include <sys/ipc.h>
#include <sys/shm.h>
#include <stdio.h>
#include <sys/types.h>
#define SHMKEY 13

int main()
{
    int shmid;
    int flag;
    int vissza;
    int choose;
    int size = 256;
    key_t key;

    key = SHMKEY;
    flag = 0;

    if((shmid = shmget(key, size, flag)) < 0)
    {
        flag = 00666 | IPC_CREAT;
        if((shmid = shmget(key, size, flag)) < 0)
        {
            perror("\nA shmget sikertelen!");
            return -1;
        }
    }
    printf("\nAz azonosító : %d\n", shmid);

    return 0;
}
```

The image shows a terminal window on the left and a code editor on the right. The terminal window shows the user compiling and running a program named shmcreate.c. The code editor shows the source code of shmcreate.c, which includes headers for stdio, stdlib, types, ipc, and shm. It defines a SHMKEY and a struct shm\_id. The main function calls shmget to create a shared memory segment, checks for success, and then calls shmctl to get the size and pid of the segment. It prints the size and pid to the console.

```
oraveczaron@oraveczaron-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cd Desktop
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc shmctl.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cd Desktop
bash: cd: Desktop: No such file or directory
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

meret: 512
pid: 0
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc shmcreate.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Az azonosító : 32811
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc shmctl.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

meret: 512
pid: 0
oraveczaron@oraveczaron-VirtualBox:~/Desktop$
```

```
File Edit View Search Tools Documents Help
shmcreate.c x shmctl.c x
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#define SHMKEY 13

struct shm_id shm_id_ds, *buf;

int main()
{
    int schmid;
    int flag;
    int vissza;
    int size = 512;
    key_t key;

    buf = &shm_id_ds;
    key = SHMKEY;
    flag = 0;

    if((schmid = shmget(key, size, flag)) < 0)
    {
        perror("\nA shmget sikertelen!");
        return -1;
    }
    vissza = shmctl(schmid, IPC_STAT, buf);
    printf("\n meret: %ld", buf->shm_segsz);
    printf("\n pid: %d\n", buf->shm_lpid);

    return 0;
}
```

shmop.c

The image shows a terminal window with the user compiling and running a program named shmop.c. The program prints the size and pid of a shared memory segment, and then prompts the user to enter a new string. The user enters 'Teszt', and the program prints it back. The user then enters 'Teszt2', and the program prints it back.

```
File Edit View Search Terminal Help
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc shmctl.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

meret: 512
pid: 0
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc shmop.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Adja meg az új szöveget : Teszt

Új szöveg : Teszt
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc shmop.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

A régi szöveg : Teszt
Adja meg az új szöveget : Teszt2

Új szöveg : Teszt2
```

3a. gyakorlati feladat

```
oraveczaron@oraveczaron-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
oraveczaron@oraveczaron-VirtualBox:~$ cd Desktop
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc gyak10_3.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Az azonosító : 32779
A régi szöveg : teszt2
Új szöveget : teszt3

Az új szöveg: teszt3
A menü opciói:
0. Státusz lekérdezése
1.törlés a memóriából
1

Megszüntette a szegment
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ cc gyak10_3.c -o main.out
oraveczaron@oraveczaron-VirtualBox:~/Desktop$ ./main.out

Az azonosító : 32793
Új szöveget : Ötvenkettő

Az új szöveg: Ötvenkettő
A menü opciói:
0. Státusz lekérdezése
1.törlés a memóriából
0

Méret: 512
Pid: 2560
oraveczaron@oraveczaron-VirtualBox:~/Desktop$
```

```
gyak10_3.c
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define SHMKEY 13

struct str {
    int lng;
    char text[1024-sizeof(int)];
} *segment;

int main()
{
    int vissza;
    int choose;
    int shmid;
    int flag;
    int size = 512;
    key_t key;

    key = SHMKEY;
    flag = 0;

    if((shmid = shmget(key, size, flag)) < 0)
    {
        flag = 00666 | IPC_CREAT;
        if((shmid = shmget(key, size, flag)) < 0)
        {
            perror("\nA shmget sikertelen volt!");
            return -1;
        }
    }

    printf("\nAz azonosító : %d\n", shmid);

    flag = 00666 | SHM_RND;
    segment = (struct str *)shmat(shmid, NULL, flag);

    if(segment == (void *) - 1)
    {
        perror("\nCsatlakozás sikertelen volt!");
        return -1;
    }
}
```

C Spaces: 4 Ln 15, Col 11 INS