

## **Operációs rendszerek BSc**

11. Gyak.

2022. 04. 20.

**Készítette:**

Bodnár Máté László

Szak:

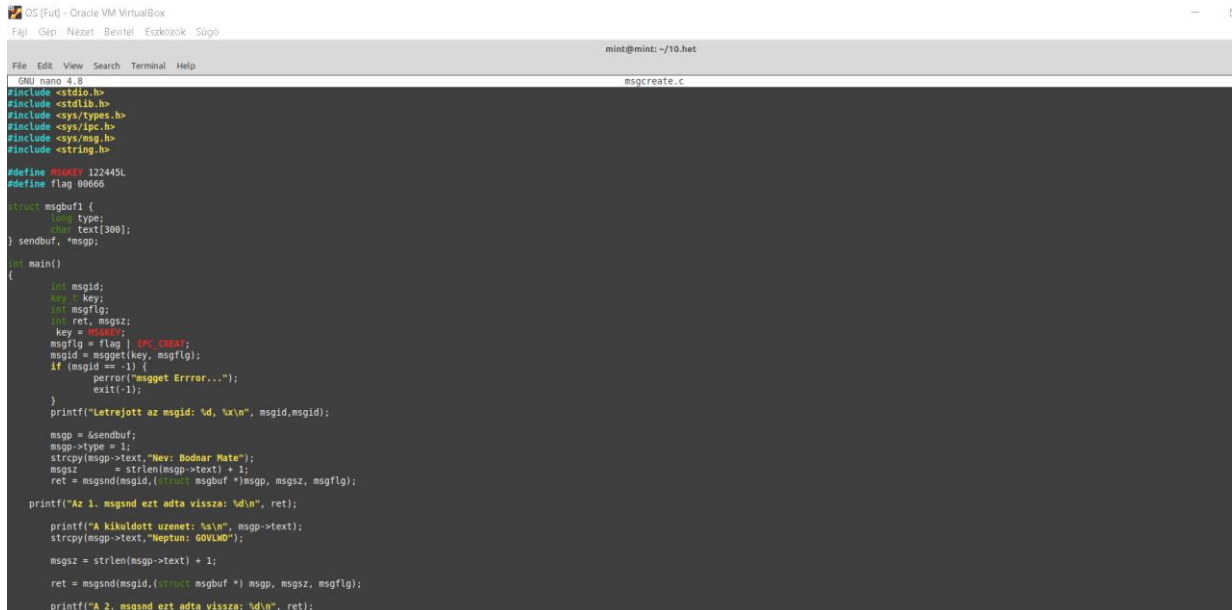
Mérnökinformatikus

Neptunkód: GOVLWD

**Miskolc, 2022**

## Feladatok

Először tanulmányozzák Vadász Dénes: Operációs rendszer jegyzet, a témához kapcsolódó fejezetét (5.3)., azaz Írjon három C nyelvű programot, ahol készít egy üzenetsort és ebbe két üzenetet tesz bele – msgcreate.c, majd olvassa ki az üzenetet - msgrcv.c, majd szüntesse meg az üzenetsort (takarít) - msgctl.c. A futtatás eredményét is tartalmazza a jegyzőkönyv. Mentés: msgcreate.c; msgrcv.c; msgctl.c.



```
OS [Fut] - Oracle VM VirtualBox
Fájl  Gép  Nézet  Bevitel  Eszközök  Súgó
File  Edit  View  Search  Terminal  Help
GNU nano 4.8
msgcreate.c
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>
#include <string.h>

#define MSGKEY 122445L
#define flag 00666

struct msgbuf {
    long type;
    char text[300];
} sendbuf, *msgp;


int main()
{
    int msgid;
    key_t key;
    int msgflg;
    int ret, msgsz;
    key = MSGKEY;
    msgflg = flag | IPC_CREAT;
    msgid = msgget(key, msgflg);
    if (msgid == -1) {
        perror("msgget Error...");
        exit(-1);
    }
    printf("Letrejött az msgid: %d, %x\n", msgid, msgid);

    msgp = &sendbuf;
    msgp->type = 1;
    strcpy(msgp->text, "Neved: Bodnar Mate");
    msgsz = strlen(msgp->text) + 1;
    ret = msgsnd(msgid, (struct msgbuf *)msgp, msgsz, msgflg);

    printf("Az 1. msgsnd ezt adta vissza: %d\n", ret);

    printf("A kiküldött üzenet: %s\n", msgp->text);
    strcpy(msgp->text, "Neptun: GOVLM0");
    msgsz = strlen(msgp->text) + 1;

    ret = msgsnd(msgid, (struct msgbuf *) msgp, msgsz, msgflg);
    printf("A 2. msgsnd ezt adta vissza: %d\n", ret);
}
```

 OS [Fut] - Oracle VM VirtualBox

Fájl Gép Nézet Bevitel Eszközök Súgó

Bekapcsoltad az **Automata billentyűzet-elkapás** lehetőséget. Ezáltal a virtuális gép automatikusan n  
File Edit View Search Terminal Help

```
mint@mint:~/10.het$ gcc msgcreate.c -o create
mint@mint:~/10.het$ ./create
Letrejott az msgid: 0, 0
Az 1. msgsnd ezt adta vissza: 0
A kikuldott uzenet: Nev: Bodnar Mate
A 2. msgsnd ezt adta vissza: 0
A kikuldott uzenet vegul: Neptun: GOVLWD
mint@mint:~/10.het$
```

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>

#define flag 00666
#define MSGKEY 12245L

struct msgbuf1 {
    long type;
    char text[300];
} recievebuf, *msgp;

struct msqid_ds ds, *buf;

int main()
{
    int msgid;
    key_t key;
    int type, msgflg;
    int ret, msgsz;

    key = MSGKEY;
    msgflg = flag | IPC_CREAT | MSG_NOERROR;

    msgid = msgget(key, msgflg);
    if (msgid == -1) {
        perror("Nem sikerult...");
        exit(-1);
    }
    printf("Az msgid-je: %d\n", msgid);

    msgp = &recievebuf;
    buf = &ds;
    msgsz = 32;
    type = 0;

    ret = msgctl(msgid, IPC_STAT, buf);
    printf("Az uzenetek szama az uzenetsorban: %ld\n", buf->msg_qnum);

    while (buf->msg_qnum) {
        ret = msgrcv(msgid, (struct msgbuf *)msgp, msgsz, type, msgflg);
        printf("Visszaadva: %d, A kapott uzenet: %s\n", ret, msgp->text);
        ret = msgctl(msgid, IPC_STAT, buf);
    }
}
```

OS [Fut] - Oracle VM VirtualBox

Fájl Gép Nézet Bevitel Eszközök Súgó

Bekapcsoltad az **Automata billentyűzet-elkapás** lehetőséget. Ezáltal a virtuális gép automatikusan mindig **elfogja** a billentyűzetet, ha a virtuális gép ablaka

File Edit View Search Terminal Help

```
mint@mint:~/10.het$ nano msgrcv.c
mint@mint:~/10.het$ gcc msgrcv.c -o rcv
mint@mint:~/10.het$ ./rcv
Az msgid-je: 1
Az uzenetek szama az uzenetsorban: 0
mint@mint:~/10.het$
```

```
OS [Fut] - Oracle VM VirtualBox
Fájl Gép Nézet Bevitel Eszközök Súgó

File Edit View Search Terminal Help

GNU nano 4.8 msgctl.c
#include <stdlib.h>
#include <stdio.h>
#include <sys/types.h>
#include <sys/msg.h>
#include <sys/ipc.h>

#define MSGKEY 122445L
#define flag 00666

int main()
{
    int msgid, msgflg, ret;
    key_t key;

    key = MSGKEY;
    msgflg = flag | IPC_CREAT;
    msgid = msgget(key, msgflg);

    ret = msgctl(msgid, IPC_RMID, 0);
    printf ("Visszatert: %d\n", ret);

    exit(0);
}

Read 24 Lines (Converted from DOS format)
Get Help Write Out Where Is Cut Text Justify Cut-Paste M-1 Undo M-2 Back Text M-3 To Bracket M-4 Previous M-5 Back M-6 Prev Word M-7
```

```
OS [Fut] - Oracle VM VirtualBox
Fájl Gép Nézet Bevitel Eszközök Súgó

Bekapcsolod az Automata billentyűzet-elkapás lehetőségét. Ezáltal a virtuális gép automatikusan mindig elfogja a billentyűzetet, ha a virtuális gép ablaka aktív és elérhetővé teszi a gazda gépen futó többi alkalmazás számára: mikor a billentyűzet el van
File Edit View Search Terminal Help

mint@mint:~/10.het$ gcc msgctl.c -o ctl
mint@mint:~/10.het$ ./ctl

Visszatert: 0
mint@mint:~/10.het$
```

5. Gyakorló feladat: Először tanulmányozzák Vadász Dénes: Operációs rendszer jegyzetet - a témához kapcsolódó fejezetét (5.3.2), azaz Írjon három C nyelvű programot, ahol készít egy osztott memóriát, melyben választott kulccsal kreál/azonosít osztott• memória szegmenst - shmcreate.c. az shmcreate.c készített osztott memória szegmens státusának lekérdezése – shmctl.c• opcionális: shmop.c shmid-del azonosít osztott memória szegmenst. Ezután a segm• nevű pointervál-tozót használva a processz virtuális címtartományába kapcsolja (attach) a szegmest (shmat() rendszerhívás). Olvassa, írja ezt a címtartományt, végül lekapcsolja (detach) a shmdt() rendszerhívással).

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#define flag 0b000
#define shmkey 12244L

int main()
{
    int shmid;
    key_t key;
    int size = 250;
    int shmflg;
    key = shmkey;
    shmflg = 0;
    if ((shmid=shmget(key, size, shmflg)) < 0) {
        printf("Nem letezik meg a szegmens... Keszitsuk el!\n");
        shmflg = flag | IPC_CREAT;
        if ((shmid=shmget(key, size, shmflg)) < 0) {
            perror("Sikertelen az shmget system call!\n");
            exit(-1);
        }
        else printf("Van mar ilyen shm szegmens!\n");
    }
    printf("\nAz shm szegmens azonositoja %d: %n", shmid);
    exit(0);
}
```

```
mint@mint:~/10.het$ nano shmcreate.c
mint@mint:~/10.het$ gcc shmcreate.c -o shmcreate
mint@mint:~/10.het$ ./shmcreate
bash: ./: Is a directory
mint@mint:~/10.het$ ./shmcreate
Nem letezik meg a szegmens... Keszitsuk el!

Az shm szegmens azonositoja 131128:
mint@mint:~/10.het$
```

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

#define SHMKEY 12244L
#define flag 00666

int main()
{
    int shmId;
    key_t key;
    int size = 300;
    int shmflg;
    struct shmId ds shmId_ds, *buf;
    buf = &shmId_ds;
    int rtn;
    int cmd;
    key = SHMKEY;
    shmflg = 0;
    if ((shmId=shmget(key, size, shmflg)) < 0) {
        perror("Sikeretelen az shmget system call!!!\n");
        exit(-1);
    }

    do
    {
        printf("\nAdd meg a parancs szamat:\n");
        printf("0. IPC STAT (statusz)\n");
        printf("1. IPC RMID (torles)\n");
        scanf("%d",&cmd);
    } while (cmd < 0 || cmd > 1);

    switch (cmd)
    {
        case 0:
            rtn = shmctl(shmId, IPC_STAT, buf);
            printf("A szegmens merete: %d\n",buf->shm_segsz);
            printf("Az utolso shmop processz pid-je: %d\n",buf->shm_lpid);
            break;
        case 1:
            rtn = shmctl(shmId, IPC_RMID, NULL);
            printf("A szegmens torolve... Hibakod: %d\n", rtn);
    }
}
```

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

#define SHMKEY 12244L
#define flag 00666

int main()
{
    int shmId;
    key_t key;
    int size=300;
    int shmflg;
    struct vmi {
        int hossz;
        char szoveg[300-sizeof(int)];
    } *segm;
    key = SHMKEY;
    shmflg = 0;

    if ((shmId=shmget(key, size, shmflg)) < 0) {
        perror("Az shmget system-call sikertelen!\n");
        exit(-1);
    }

    shmflg = flag | SHM_RWP;
    segm = (struct vmi *)shmatt(shmId, NULL, shmflg);
    if (segm == (struct vmi *)-1) {
        perror("Az attach sikertelen!\n");
        exit(-1);
    }

    if (strlen(segm->szoveg) > 0)
        printf("\n A regi szoveg: %s (%d hosszon)",segm->szoveg,segm->hossz);

    printf("\nKerem adjon meg egy uj szoveget!\n");
    gets(segm->szoveg);
    printf("Az uj szoveg: %s\n",segm->szoveg);
    segm->hossz=strlen(segm->szoveg);
    shmId(segm);
    exit(0);
}
```



```
mint@mint:~/10.het$ ./smctl
bash: ./smctl: No such file or directory
mint@mint:~/10.het$ ./shmctl
Sikeretelen az shmget system call!!
: Invalid argument
mint@mint:~/10.het$ ./shmop
Az shmget system-call sikertelen!
: Invalid argument
mint@mint:~/10.het$
```