

# **Operációs rendszerek BSc**

11.gyak.

2021. 05. 03.

**Készítette:**  
**Ábrahám Péter István**

Gazdaságinformatikus  
Neptunkód: **A4XIOV**

**Miskolc, 2021**

„1. Adott egy rendszer (foglalási stratégiák), melyben a következő

- Szabad területek: 30k, 35k, 15k, 25k, 75k, 45k és
- Foglalási igények: 39k, 40k, 33k, 20k, 21k állnak rendelkezésre.

Határozza meg *változó partíció esetén* a következő algoritmusok felhasználásával: first fit, next fit, best fit, worst fit a foglalási igényeknek megfelelő helyfoglalást!

Foglalási igények	Szabad területek:					
	30	35	15	25	75	45
39	30	35	15	25	75	45
40	30	35	15	25	75	45
33	30	35	15	25	75	45
20	30	35	15	25	75	45
21	30	35	15	25	75	45

First Fit						
Foglalási igények	Szabad területek:					
	30	35	15	25	75	45
39	30	35	15	25	39   36	45
40	30	35	15	25	75	40   5
33	30	33   2	15	25	75	45
20	20   10	35	15	25	75	45
21	30	35	15	21   4	75	45

Next Fit						
Foglalási igények	Szabad területek:					
	30	35	15	25	75	45
39	30	35	15	25	39   36	45
40	30	35	15	25	75	40   5
33	30	33   2	15	25	75	45
20	30	35	15	20   5	75	45
21	30	35	15	25	21   54	45

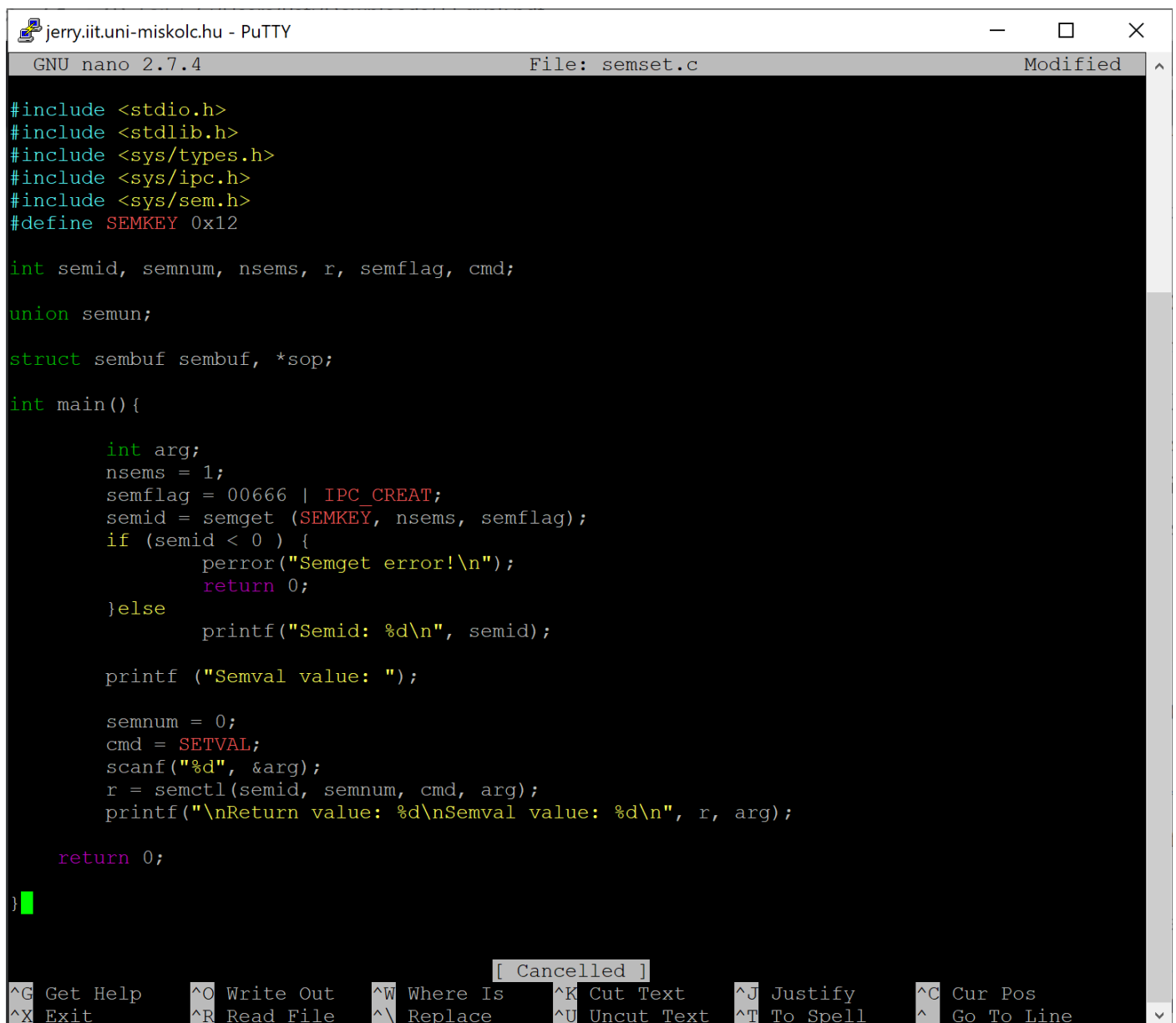
Best Fit						
Foglalási igények	Szabad területek:					
	30	35	15	25	75	45
39	30	35	15	25	75	39   6
40	30	35	15	25	40   35	45
33	30	33   2	15	25	75	45
20	30	35	15	20   5	75	45
21	21   9	35	15	25	75	45

Worst Fit						
Foglalási igények	Szabad területek:					
	<b>30</b>	<b>35</b>	<b>15</b>	<b>25</b>	<b>75</b>	<b>45</b>
<b>39</b>	30	35	15	25	39   36	45
<b>40</b>	30	35	15	25	75	40   5
<b>33</b>	30	35	15	25	39   36	45
					33   3	
<b>20</b>	30	20   15	15	25	75	45
<b>21</b>	21   9	35	15	25	75	45

**2. Gyakorló feladat:** A feladat megoldásához először tanulmányozza Vadász Dénes: Operációs rendszer jegyzet, a témához kapcsolódó fejezetét (6.4)., azaz Írjon C nyelvű programokat, ahol

- kreál/azonosít szemafor készletet, benne N szemafor-t. A kezdő értéket 0-ra állítja – **semset.c**,
- kérdezze le és írja ki a pillanatnyi szemafor értéket – **semval.c**
- szüntesse meg a példácskák szemafor készletét – **semkill.c**
- sembuf.sem\_op=1 értékkel inkrementálja a szemafort – **semup.c**

**A futtatás eredményét is tartalmazza a jegyzőkönyv.**



The screenshot shows a terminal window titled "jerry.iit.uni-miskolc.hu - PuTTY". The editor is GNU nano 2.7.4, editing the file "semset.c". The code is as follows:

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/sem.h>
#define SEMKEY 0x12

int semid, semnum, nsems, r, semflag, cmd;

union semun;

struct sembuf sembuf, *sop;

int main() {

    int arg;
    nsems = 1;
    semflag = 00666 | IPC_CREAT;
    semid = semget (SEMKEY, nsems, semflag);
    if (semid < 0 ) {
        perror("Semget error!\n");
        return 0;
    }else
        printf("Semid: %d\n", semid);

    printf ("Semval value: ");

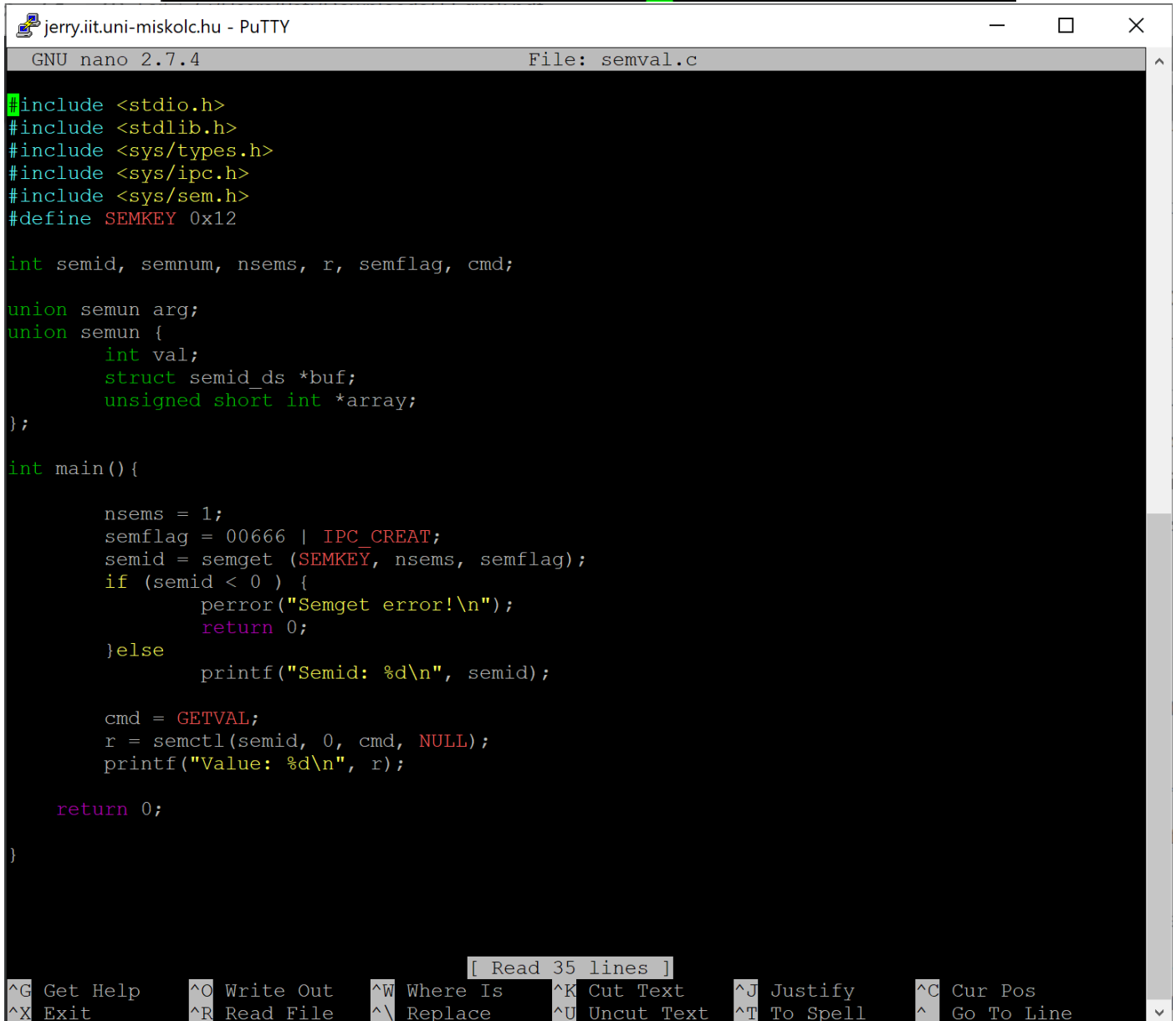
    semnum = 0;
    cmd = SETVAL;
    scanf("%d", &arg);
    r = semctl(semid, semnum, cmd, arg);
    printf("\nReturn value: %d\nSemval value: %d\n", r, arg);

    return 0;
}
```

The bottom of the terminal shows a status bar with various keyboard shortcuts: ^G Get Help, ^O Write Out, ^W Where Is, ^K Cut Text, ^J Justify, ^C Cur Pos, ^X Exit, ^R Read File, ^\ Replace, ^U Uncut Text, ^T To Spell, ^ Go To Line. A "[ Cancelled ]" message is also visible in the status bar.

```
abraham1@jerry:~$ ./semset.out
Semid: 950283
Semval value: 12

Return value: 0
Semval value: 12
```



The screenshot shows a terminal window titled "jerry.iit.uni-miskolc.hu - PuTTY" with a nano 2.7.4 editor open to the file "semval.c". The code is as follows:

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/sem.h>
#define SEMKEY 0x12

int semid, semnum, nsems, r, semflag, cmd;

union semun arg;
union semun {
    int val;
    struct semid_ds *buf;
    unsigned short int *array;
};

int main(){

    nsems = 1;
    semflag = 00666 | IPC_CREAT;
    semid = semget (SEMKEY, nsems, semflag);
    if (semid < 0 ) {
        perror("Semget error!\n");
        return 0;
    }else
        printf("Semid: %d\n", semid);

    cmd = GETVAL;
    r = semctl(semid, 0, cmd, NULL);
    printf("Value: %d\n", r);

    return 0;
}
```

The bottom of the window shows the nano editor's status bar with various shortcuts: ^G Get Help, ^O Write Out, ^W Where Is, ^K Cut Text, ^J Justify, ^C Cur Pos, ^X Exit, ^R Read File, ^\_ Replace, ^U Uncut Text, ^T To Spell, and ^\_ Go To Line. A status indicator "[ Read 35 lines ]" is also present.

```
abraham1@jerry:~$ ./semval.out
Semid: 950283
Value: 12
abraham1@jerry:~$
```

jerry.iit.uni-miskolc.hu - PuTTY

GNU nano 2.7.4 File: semkill.c

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/sem.h>
#define SEMKEY 0x10

int semid, semnum, nsems, r, semflag, cmd;

struct sembuf sembuf, *sop;

union semun {
    int val;
    struct semid_ds *buf;
    unsigned short int *array;
};

int main() {

    int arg;
    nsems = 1;
    semflag = 00666 | IPC_CREAT;
    semid = semget (SEMKEY, nsems, semflag);
    if (semid < 0 ) {
        perror("Semget error!\n");
        return 0;
    }else
        printf("Semid: %d\n", semid);

    cmd = IPC_RMID;
    r = semctl(semid, 0, cmd, arg);
    printf("Kill return value: %d\n", r);

    return 0;
}
```

[ Read 37 lines ]

^G Get Help    ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify    ^C Cur Pos  
^X Exit        ^R Read File    ^\ Replace    ^U Uncut Text    ^T To Spell    ^\_ Go To Line

```
abraham1@jerry:~$ ./semkill.out
Semid: 1376269
Kill return value: 0
```

jerry.iit.uni-miskolc.hu - PuTTY

GNU nano 2.7.4 File: semup.c

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/sem.h>
#define SEMKEY 0x12

int semid, nsems, r, semflag;

struct sembuf sembuf, *sop;

unsigned nsops;

int main(){

    nsems = 1;
    semflag = 00666 | IPC_CREAT;
    semid = semget (SEMKEY, nsems, semflag);
    if (semid < 0 ) {
        perror("Semget error!\n");
        return 0;
    }else
        printf("Semid: %d\n", semid);

    nsops = 1;
    sembuf.sem_flg = 0066;
    sembuf.sem_op = 1;
    sembuf.sem_num = 0;
    sop = &sembuf;
    r = semop(semid, sop, nsops);
    printf("Return value: %d\n", r);

    return 0;
}
```

[ Read 36 lines ]

^G Get Help	^O Write Out	^W Where Is	^K Cut Text	^J Justify	^C Cur Pos
^X Exit	^R Read File	^_ Replace	^U Uncut Text	^T To Spell	^_ Go To Line

```
abraham1@jerry:~$ ./semup.out
Semid: 950283
Return value: 0
```