

# **Operációs rendszerek BSc**

10. Gyak.

2022. 04. 13.

**Készítette:**

Honti Dániel BSc

Programtervező Informatikus

HR6121

**Miskolc, 2022**

## 1. feladat

Az előadáson bemutatott mintaprogram alapján készítse el a következő feladatot. Adott egy rendszerbe az alábbi erőforrások: R (R1: 10; R2: 5; R3: 7) A rendszerbe 5 processz van: P0, P1, P2, P3, P4

Kérdés: Kielégíthető-e P1 (1,0,2), P4 (3,3,0) ill. P0 (0,2,0) kérése úgy, hogy biztonságos legyen, holtpontmentesség szempontjából a rendszer - a következő kiinduló állapot alapján.

Külön-külön táblázatba oldja meg a feladatot!

- Határozza meg a processzek által igényelt erőforrások mátrixát?
- Határozza meg pillanatnyilag szabad erőforrások számát?
- Igazolja, magyarázza az egyes processzek végrehajtásának lehetséges sorrendjét számolással?

P1(1,0,2) kielégíthető kérés, P1-P3-P4-P0-P2 sorrendben a maximális igények kielégíthetők, a rendszer biztonságos állapotban marad.

Az IGÉNY mátrix és a KÉSZLET a képen láthatók.

A kép alján az új készlet számítások igazolják a felírt lehetséges sorrendet, annak biztonságosságát.

P1 (1,0,2), P4(3,3,0) ill. P0 (0,2,0)				P1	1	0	2										
P1(1,0,2)																	
	MAX IGÉNY				FOGLAL								Készlet Igény				
	R1	R2	R3		R1	R2	R3		R1	R2	R3		R1	R2	R3		
P0	7	5	3		0	1	0		7	4	3		4.	-5	-1	-3	
P1	3	2	2		3	0	2		0	2	0		1.	2	1	0	
P2	9	0	2		3	0	2		6	0	0		5.	-4	3	0	
P3	2	2	2		2	1	1		0	1	1		2.	2	2	-1	
P4	4	3	3		0	0	2		4	3	1		3.	-2	0	-1	
					8	2	7										
				Készlet	2	3	0	1.	Végrehajt: P1								
									Készlet	5	3	2					
								2.	Végrehajt: P3								
									Készlet	7	4	3					
								3.	Végrehajt: P4								
									Készlet	7	4	5					
								4.	Végrehajt:P0								
									Készlet	7	5	5					
								5.	Végrehajt: P2								
									Készlet	10	5	7					
									Biztonságos Állapot								

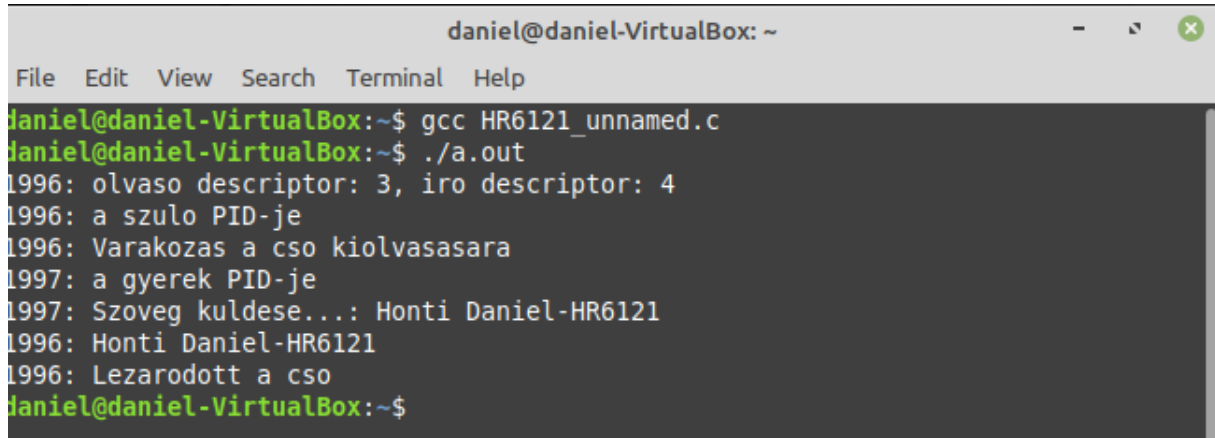
P4(3, 3, 0) nem teljesíthető, mert az IGÉNY mátrixban szereplő igények valamelyik erőforrásra nézve mindig nagyobbak, mint a rendelkezésre álló KÉSZLET, a rendszer nem lesz biztonságos állapotban.

[illegible]

[illegible]

## 2. feladat

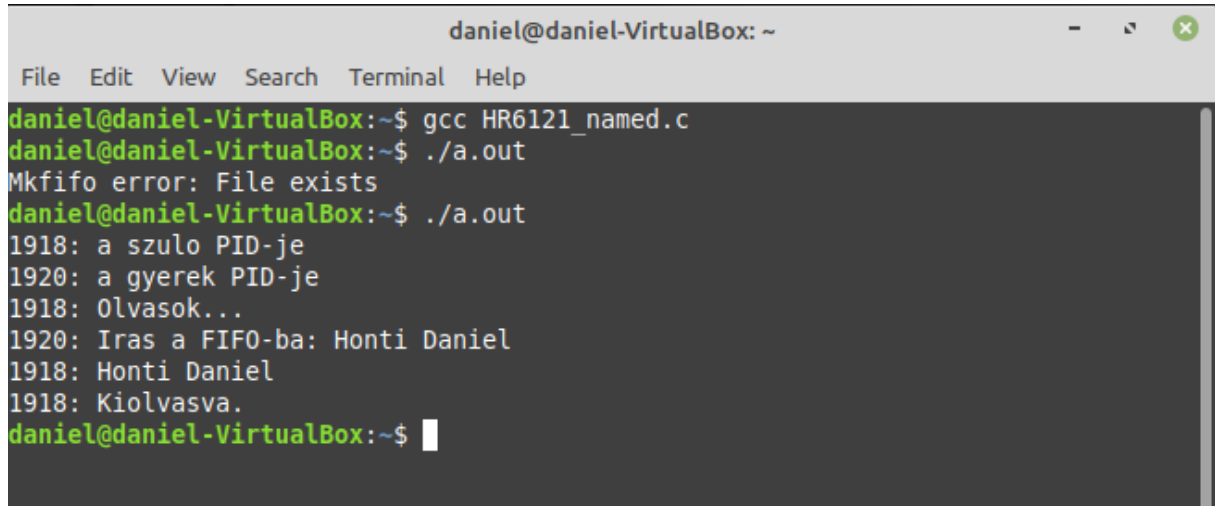
Készítsen C nyelvű programot, ahol egy szülő processz létrehoz egy csővezetékét, a gyerek processz beleír egy szöveget a csővezetékbe (A kiírt szöveg: XY neptunkod), a szülő processz ezt kiolvassa, és kiírja a standard kimenetre.



```
daniel@daniel-VirtualBox: ~
File Edit View Search Terminal Help
daniel@daniel-VirtualBox:~$ gcc HR6121_unnamed.c
daniel@daniel-VirtualBox:~$ ./a.out
1996: olvaso descriptor: 3, iro descriptor: 4
1996: a szulo PID-je
1996: Varakozas a cso kiolvasasara
1997: a gyerek PID-je
1997: Szoveg kuldesse...: Honti Daniel-HR6121
1996: Honti Daniel-HR6121
1996: Lezarodott a cso
daniel@daniel-VirtualBox:~$
```

### 3. feladat

Készítsen C nyelvű programot, ahol egy szülő processz létrehoz egy nevesített csővezetékét (neve: neptunkod), a gyerek processz beleír egy szöveget a csővezetékbe (A hallgató neve: pl.: Keserű Ottó), a szülő processz ezt kiolvassa, és kiírja a standard kimenetre.



```
daniel@daniel-VirtualBox: ~
File Edit View Search Terminal Help
daniel@daniel-VirtualBox:~$ gcc HR6121_named.c
daniel@daniel-VirtualBox:~$ ./a.out
Mkfifo error: File exists
daniel@daniel-VirtualBox:~$ ./a.out
1918: a szulo PID-je
1920: a gyerek PID-je
1918: Olvasok...
1920: Iras a FIFO-ba: Honti Daniel
1918: Honti Daniel
1918: Kiolvasva.
daniel@daniel-VirtualBox:~$
```

## 4. feladat

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.

```
daniel@daniel-VirtualBox:~$ gcc msgcreate.c
daniel@daniel-VirtualBox:~$ ./a.out
Az msgid létrejött: 0, 0
Az 1. msgsnd() visszaadott: 0
A kiküldött üzenet: Egyik üzenet
A 2. msgsnd visszaadott: 0
A kiküldött üzenet: Masik üzenet
daniel@daniel-VirtualBox:~$ ipcs

----- Message Queues -----
key          msqid      owner      perms      used-bytes   messages
0x0009fbf1  0           daniel     666        26           2

Shared Memory Segments
```

```
daniel@daniel-VirtualBox:~$ gcc msgrcv.c
daniel@daniel-VirtualBox:~$ ./a.out
Az msgid: 0
Az üzenetek száma az üzenetsorban: 2
msgrcv() visszaadta: 13, a kapott üzenet: Egyik üzenet
msgrcv() visszaadta: 13, a kapott üzenet: Masik üzenet
daniel@daniel-VirtualBox:~$ ipcs

----- Message Queues -----
key          msqid      owner      perms      used-bytes   messages
0x0009fbf1  0           daniel     666         0           0
```

```
daniel@daniel-VirtualBox:~$ gcc msgctl.c
daniel@daniel-VirtualBox:~$ ./a.out
Visszatért: 0
daniel@daniel-VirtualBox:~$ ipcs

----- Message Queues -----
key          msqid      owner      perms      used-bytes   messages
```

#### 4a.

Írjon egy C nyelvű programot, melyben

- az egyik processz létrehozza az üzenetsort, és szövegeket küld bele, exit üzenetre kilép,
- másik processzben lehet választani a feladatok közül: üzenetek darabszámának lekérdezése, 1 üzenet kiolvasása, összes üzenet kiolvasása, üzenetsor megszüntetése, kilépés.

Mentés: gyak10\_4.c

A futtatás eredményét is tartalmazza a jegyzőkönyv.

```
Az msgid létrejött: 1, 1
Irja be az üzenetet (kilepeshez exit)!
Uzenet
A msgsnd() visszaadott: 0
A kikuldott uzenet: Uzenet
Irja be az üzenetet (kilepeshez: exit)!
Ez egy uzenet
A msgsnd() visszaadott: 0
A kikuldott uzenet: Ez egy uzenet
Irja be az üzenetet (kilepeshez: exit)!
Ez is
A msgsnd() visszaadott: 0
A kikuldott uzenet: Ez is
Irja be az üzenetet (kilepeshez: exit)!
Hello
A msgsnd() visszaadott: 0
A kikuldott uzenet: Hello
Irja be az üzenetet (kilepeshez: exit)!
exit
daniel@daniel-VirtualBox:~$ ipcs

----- Message Queues -----
key          msqid      owner      perms      used-bytes  messages
0x0009fbf1  1          daniel     666        33          4
```

```

daniel@daniel-VirtualBox:~$ ./a.out
Az msgid: 1

1 - egy uzenetet kiolvasasa
2 - osszes uzenet kiolvasasa
3 - darabszam lekerdezese
4 - uzenetsor megszuntetese es kilepes
3
Az uzenetek szama az uzenetsorban: 4

1 - egy uzenetet kiolvasasa
2 - osszes uzenet kiolvasasa
3 - darabszam lekerdezese
4 - uzenetsor megszuntetese es kilepes
1
msgrcv() visszaadta: 7, a kapott uzenet: Uzenet

1 - egy uzenetet kiolvasasa
2 - osszes uzenet kiolvasasa
3 - darabszam lekerdezese
4 - uzenetsor megszuntetese es kilepes
2
msgrcv() visszaadta: 14, a kapott uzenet: Ez egy uzenet
msgrcv() visszaadta: 6, a kapott uzenet: Ez is

```

```

1 - egy uzenetet kiolvasasa
2 - osszes uzenet kiolvasasa
3 - darabszam lekerdezese
4 - uzenetsor megszuntetese es kilepes
2
msgrcv() visszaadta: 14, a kapott uzenet: Ez egy uzenet
msgrcv() visszaadta: 6, a kapott uzenet: Ez is
msgrcv() visszaadta: 6, a kapott uzenet: Hello

```

```

1 - egy uzenetet kiolvasasa
2 - osszes uzenet kiolvasasa
3 - darabszam lekerdezese
4 - uzenetsor megszuntetese es kilepes
4
Megszuntetes visszatert: 0

```

```

daniel@daniel-VirtualBox:~$ ipcs

```

```

----- Message Queues -----
key          msqid          owner          perms          used-bytes   messages

```



## 5. feladat

Először tanulmányozzák Vadász Dénes: Operációs rendszer jegyzetét - 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 shmctl-del azonosít osztott memória szegmenst. Ezután a segm nevű pointerváltozót használva a processz virtuális címtartományába kapcsolja (attach) a szegment (shmat() rendszerhívás). Olvassa, írja ezt a címtartományt, végül lekapcsolja (detach) a shmdt() rendszerhívással).

```
daniel@daniel-VirtualBox:~$ gcc shmcreate.c
daniel@daniel-VirtualBox:~$ ./a.out
Szegmens keszítése...

Az shm szegmens azonosítója 32780:
daniel@daniel-VirtualBox:~$ ipcs

----- Message Queues -----
key          msqid      owner      perms      used-bytes   messages

----- Shared Memory Segments -----
key          shmid      owner      perms      bytes       nattch     status
0x00000000   32769      daniel     600        524288      2         dest
0x00000000   32770      daniel     600        4194304     2         dest
0x00000000   32771      daniel     600        1622016     2         dest
0x00000000    6         daniel     600        524288      2         dest
0x00000000    8         daniel     600        16384       1         dest
0x00000000   32777      daniel     600        4194304     2         dest
0x00000000   32778      daniel     600        1441792     2         dest
0x00000000   11        daniel     600        33554432    2         dest
0x00011111   32780      daniel     666        512         0
```

```
daniel@daniel-VirtualBox:~$ gcc shmctl.c
daniel@daniel-VirtualBox:~$ ./a.out

Parancs:
0. IPC_STAT (status)
1. IPC_RMID (torles)
> 0
Szegmens merete: 512
Utolso shmop()-os processz pid-je: 0
daniel@daniel-VirtualBox:~$ ./a.out

Parancs:
0. IPC_STAT (status)
1. IPC_RMID (torles)
> 1
Szegmens torolve. Visszateresi ertek: 0
```

```
daniel@daniel-VirtualBox:~$ gcc shmop.c
daniel@daniel-VirtualBox:~$ ./a.out
Uj szoveg:
Az alma piros
Az uj szoveg: Az alma piros
daniel@daniel-VirtualBox:~$ ./a.out
Regi szoveg: Az alma piros (13 hosszon)
Uj szoveg:
A banan sarga
Az uj szoveg: A banan sarga
daniel@daniel-VirtualBox:~$ ./a.out
Regi szoveg: A banan sarga (13 hosszon)
Uj szoveg:
A citrom sarga
Az uj szoveg: A citrom sarga
daniel@daniel-VirtualBox:~$
```

## 5a.

Írjon egy C nyelvű programot, melyben

- egyik processz létrehozza az osztott memóriát,
- másik processz rácsatlakozik az osztott memóriára, ha van benne valamilyen szöveg, akkor kiolvassa, majd beleír új üzenetet,
- harmadik processznél lehet választani a feladatok közül: státusz lekérése (szegmens mérete, utolsó shmop-os proc. pid-je), osztott memória megszüntetése, kilépés (2. és 3. proc. lehet egyben is)”

A futtatás eredményét is tartalmazza a jegyzőkönyv.

Mentés: gyak10\_5.c

```
daniel@daniel-VirtualBox:~$ gcc gyak10_5.c
daniel@daniel-VirtualBox:~$ ./a.out
A szegmens elkeszult

Az shm szegmens azonositoja 32801:
daniel@daniel-VirtualBox:~$ ipcs
```

0x00000000	32	daniel	600	524288	2	dest
0x00011122	32801	daniel	666	512	0	
0x00000000	34	daniel	600	524288	2	dest

```
daniel@daniel-VirtualBox:~$ gcc gyak10_5b.c
daniel@daniel-VirtualBox:~$ ./a.out
1 - Kiolvasas es beiras
2 - Statusz lekerese
3 - Szegmens megszuntetese es kilepes
1
Uj szoveg:
Alma
Az uj szoveg: Alma
1 - Kiolvasas es beiras
2 - Statusz lekerese
3 - Szegmens megszuntetese es kilepes
2
Szegmens merete: 512
Utolso shmop()-os processz pid-je: 2429
1 - Kiolvasas es beiras
2 - Statusz lekerese
3 - Szegmens megszuntetese es kilepes
1
Regi szoveg: Alma (4 hosszon)
Uj szoveg:
Korte
Az uj szoveg: Korte
1 - Kiolvasas es beiras
2 - Statusz lekerese
3 - Szegmens megszuntetese es kilepes
3
Szegmens torolve. Visszateresi ertek: 0
```