

JEGYZŐKÖNYV

Operációs rendszerek BSc

2021 tavasz féléves feladat

Készítette: **Nyeste Ágoston**

Neptunkód: **WH85ZH**

1.feladat: Erőforrások: R1=10, R2=5, R3=7.

5 processz P0, P1, P2, P3, P4. Kielégíthető-e P4 (3,3,0) ill. P0 (0,2,0) kérése úgy, hogy biztonságos legyen, holtpontmentesség szempontjából a rendszer.

P0 (0,2,0)

Erőforrások: R1=10, R2=5, R3=7								
	1.lépés					2.lépés		
	MAX Igény					Foglal		
	R1	R2	R3			R1	R2	R3
P0	7	5	3		P0	0	1	0
P1	3	2	2		P1	2	0	0
P2	9	0	2		P2	3	0	2
P3	2	2	2		P3	2	1	1
P4	4	3	3		P4	0	0	2

A foglalthoz hozzá kell számolni a P0 plusz igényét.

Plusz	2.lépés				
P0(0,2,0)	Foglal				
	R1	R2	R3		
P0	0	3	0		
P1	2	0	0		
P2	3	0	2		
P3	2	1	1		
P4	0	0	2		
Összesen:	7	4	5		
Készlet:	3	1	2	(3,1,2)	

Kiszámoltuk a készletet.

	4.lépés		
Lépés	Készlet	Process	Uj készlet
1	3,1,2	P1	5,1,2
2	5,1,2	P3	7,2,3
3	7,2,3	P2	10,2,5
4	10,2,5	P4	10,2,7
5	10,2,7	P0	10,5,7

Eredeti forrásszám megvan: Biztosnágos

Végrehajtási sorrend: P1 P3 P2 P4 P0

Max igényből kivonjuk a foglalt igényt.

	3.lépés	Számítás				3.lépés	Számítás	
		Igény					Igény	
	R1	R2	R3			R1	R2	R3
P0	$7-0=7$	$5-3=2$	$3-0=3$		P0	7	2	3
P1	$3-2=1$	$2-0=2$	$2-0=2$		P1	1	0	0
P2	$9-3=6$	0	$2-2=0$		P2	6	0	0
P3	$2-2=0$	$2-1=1$	$2-1=1$		P3	0	1	1
P4	$4-0=4$	$3-0=3$	$3-2=1$		P4	4	3	1

Így próbálunk felállítani egy biztonságos processz sorrendet:

P4(3,3,0)

Erőforrás: R1=10, R2=5, R3 = 7								
	1.lépés					2.lépés		
		Max Igény					Foglal	
	R1	R2	R3			R1	R2	R3
P0	7	5	3		P0	0	1	0
P1	3	2	2		P1	2	0	0
P2	9	0	2		P2	3	0	2
P3	2	2	2		P3	2	1	1
P4	4	3	3		P4	0	0	2

A foglalthoz hozzá kell számolni a P0 plusz igényét

Plusz					
P4(3,3,0)		R1	R2	R3	
P0		0	1	0	
P1		2	0	0	
P2		3	0	2	
P3		2	1	1	
P4		3	3	2	
Összesen:		10	5	5	
Készlet:		$10-10=0$	$5-5=0$	$7-5=2$	(0,0,2)

Meghatározzuk az igényeket:

	3.lépés	Számítás				3.lépés	Számítás	
		Igény					Igény	
	R1	R2	R3			R1	R2	R3
P0	7-0=7	5-3=2	3-0=3		P0	7	4	3
P1	3-2=1	2-0=2	2-0=2		P1	1	0	0
P2	9-3=6	0	2-2=0		P2	6	0	0
P3	2-2=0	2-1=1	2-1=1		P3	0	1	1
P4	4-0=4	3-0=3	3-2=1		P4	1	0	1

A 0,0,2 nem elégíti ki az igényeket, nincs biztonságos sorrend.

	4.lépés				
Lépés	Készlet	Process	Uj készlet		
1	(0,0,2)				
2					
3					
4					
5					
0,0,2 -vel egyik process igény sem kielégíthető, nem biztonságos					

2. Gyakorló feladat: Először tanulmányozzák Vadász Dénes: Operációs rendszer jegyzet, a témához kapcsolódó fejezetét (5.3)., azaz

Írjanak három C nyelvű programot, ahol készít egy üzenetsort és ebbe két üzenetet tesz bele – **msgcreate.c**, majd olvassa ki az üzenetet - **msgrev.c**, majd szüntesse meg az üzenetsort (takarít) - **msgctl.c**.

```
Az msgid 2, 2 :
Az 1. msgsnd visszaadott 0-t
A kikuldott uzenet:Egyik uzenet
Az 2. msgsnd visszaadott 0-t
A kikuldott uzenet:Masik uzenet
```

```
Az msgid: 2
Az uzenetek szama: 4
Az rtn: 13, a vett uzenet:Egyik uzenet
Az rtn: 13, a vett uzenet:Masik uzenet
Az rtn: 13, a vett uzenet:Egyik uzenet
Az rtn: 13, a vett uzenet:Masik uzenet
```

```
Vissztert: 0
```

2a. 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.

```
agoston@agostons-ubuntu:~/Desktop/gyak10$ ./gyak10_2
Küldés sikeres
Üzenetek száma: 3
Válasz: 1
Válasz: 2
Válasz: 3
agoston@agostons-ubuntu:~/Desktop/gyak10$
```

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 szegmenst (shmat() rendszerhívás). Olvassa, írja ezt a címtartományt, végül lekapcsolja (detach) a shmdt() rendszerhívással).

```
agoston@agostons-ubuntu:~/Desktop/gyak10$ ./shmcreate
agoston@agostons-ubuntu:~/Desktop/gyak10$ ./shmctl
agoston@agostons-ubuntu:~/Desktop/gyak10$ ./shmop
Közös memória tartalom : Sample Textagoston@agostons-ubuntu:~/Desktop/gyak10$
```

3a. 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: státus lekérése (szegmens mérete, utolsó shmop-os proc. pid-je), osztott memória megszüntetése, kilépés (2. és 3. proc. lehet egyben is)”

```
agoston@agostons-ubuntu:~/Desktop/gyak10$ ./gyak10_3
Processz 2 olvas...
Processz 1 sikeresen lefoglalta a memóriát
Osztott memóriában a szöveg : Sample Text
Processz 2 sikeresen elküldte az üzenetet
agoston@agostons-ubuntu:~/Desktop/gyak10$ Processz 3
Szegmens méret: 256
Utolsó processz PID-je : 4101
```