

JEGYZŐKÖNYV

Operációs rendszerek BSc

2022. tavasz féléves feladat

Készítette: Honti Dániel

Neptunkód: HR6121

Miskolc, 2022

ALGORITMUS

A feladat leírása:

24. Adott egy számítógépes rendszer, melyben a következő

- szabad memória területek: 23KB, 64KB, 10KB, 80KB, 12KB, 50KB és 40KB melynek
- foglalási igénye: 65kB, 21kB, 48kB, 13kB, 62kB.

Határozza meg változó méretű partíció esetén a következő algoritmusok felhasználásával: *first fit*, *best fit* a foglalási igényeknek megfelelő helyfoglalást – táblázatos formában!

Magyarázza a kapott eredményeket és hogyan lehet az eredményeket javítani!

A feladat elkészítésének lépései:

Két külön táblázatban vizsgáltam a két algoritmus helyfoglalásának eredményét.

FIRST FIT

		Szabad területek						
		23	64	10	80	12	50	40
Foglalandó	65	23	64	10	65+15	12	50	40
	21	21+2	64	10	15	12	50	40
	48	2	48+16	10	15	12	50	40
	13	2	13+3	10	15	12	50	40
	62	2	3	10	15	12	50	40

Eredménye: [21], 2, [48], [13], 3, 10, [65], 15, 12, 50, 40 és a 62-es foglalás még várakozik.

Probléma adódik abból, hogy nem elégül ki minden foglalási igény, a 62kB igényű processz várakozik. A hamarabb beérkező igények a FIRST FIT algoritmus miatt elfoglalják azokat a területeket, amibe beleférne a processz.

Erre megoldást nyújthat ha felszabadítunk területeket, vagy egy másik algoritmust, például a BEST FIT-et használjuk.

BEST FIT

		Szabad területek						
		23	64	10	80	12	50	40
Foglalandó	65	23	64	10	65+15	12	50	40
	21	21+2	64	10	15	12	50	40
	48	2	64	10	15	12	48+2	40
	13	2	64	10	13+2	12	2	40
	62	2	62+2	10	2	12	2	40

Eredménye: [21], 2 [62], 2, 10, [65], [13], 2, 12, [48], 2, 40

Mindegyik foglalási igény kielégíthető.

A FIRST FIT algoritmushoz képest már az is előrelépés, hogy mindegyik foglalási igény kielégíthető. Az összes fennmaradó hely 70 kB, ami 7 részre oszlik, de ezek közül az egyetlen „nagyobb” szabad terület a 40 kB-os, a többi mind 2, 10 vagy 12.

A két algoritmus közül muszáj a BEST FIT-et választanunk, ha mindegyik igényt ki szeretnénk elégíteni, különben nem mindegyik kap szabad helyet.

IPC

A feladat leírása:

6. Írjon C nyelvű programot, ami:

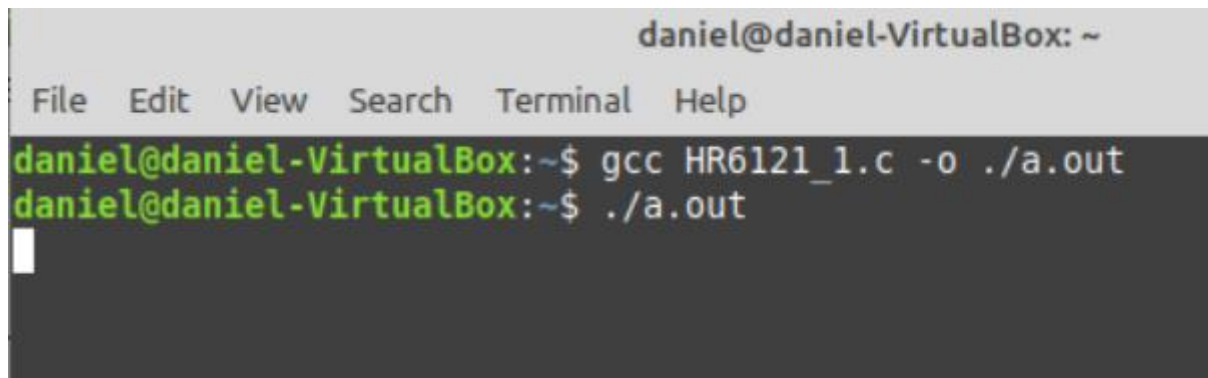
- létrehoz egy csővezeték, megnyitja
- olvasni próbál belőle
- közben egy második program ír bele egy random számot
- az első program kiírja a kapott számot és visszaküldi ennek a dupláját
- a második program kiolvassa és kiírja a képernyőre
- a processzek megszűnnek és a második program eltünteti a nevesített csővezeték

Elkészítések lépései:

1. Az első program létrehozása (HR6121_1.c), csővezeték deklarálása és létrehozása mkfifo() parancs segítségével.
2. Az első programban csővezeték megnyitása olvasásra open()-nel.
3. A második program létrehozása (HR6121_2.c), ugyan arra a csővezetékre csatlakozás.
4. A második programban rand() segítségével véletlen szám generálás.
5. A második programban a csővezeték megnyitása írásra, beleírjuk a rand()-al generált véletlen számot.
6. A második programban bezárjuk a csővezeték close()-al, majd újra megnyitjuk, ezalkalommal olvasásra, várva a szám dupláját.
7. Az első program megkapja a véletlen számot, kiírja, majd bezáródik a csővezeték.
8. Az első program megduplázza a számot.
9. Az első program megnyitja írásra a csővezeték, és beleírja a megduplázott számot.
10. Az első program bezárja a csővezeték, majd leáll.
11. A második program megkapja a megduplázott számot, kiírja, majd bezárja a csővezeték.
12. A második program eltávolítja a csővezeték, majd leáll.

Futtatás:

Futtatjuk az első programot (HR6121_1.c), ez vár egy számra.



```
daniel@daniel-VirtualBox: ~  
File Edit View Search Terminal Help  
daniel@daniel-VirtualBox:~$ gcc HR6121_1.c -o ./a.out  
daniel@daniel-VirtualBox:~$ ./a.out  
█
```

Futtatjuk a második programot (HR6121_2.c), ez generál egy véletlen számot, beírja a csővezetékbe, majd vár egy visszaadott számra.

```
daniel@daniel-VirtualBox: ~  
File Edit View Search Terminal Help  
daniel@daniel-VirtualBox:~$ gcc HR6121_2.c -o b.out  
daniel@daniel-VirtualBox:~$ ./b.out
```

Az első program megkapja a véletlen számot, kiírja, megduplázza, beírja a csővezetékbe, bezárja a csővezetékét, majd befejeződik.

```
daniel@daniel-VirtualBox: ~  
File Edit View Search Terminal Help  
daniel@daniel-VirtualBox:~$ gcc HR6121_1.c -o ./a.out  
daniel@daniel-VirtualBox:~$ ./a.out  
Number: 62  
daniel@daniel-VirtualBox:~$ █
```

A második program megkapja a megduplázott számot, kiírja, bezárja a csővezetékét és kitörli a csővezetékét, majd befejeződik.

```
daniel@daniel-VirtualBox: ~  
File Edit View Search Terminal Help  
daniel@daniel-VirtualBox:~$ gcc HR6121_2.c -o b.out  
daniel@daniel-VirtualBox:~$ ./b.out  
Number doubled: 124  
daniel@daniel-VirtualBox:~$ █
```