

JEGYZŐKÖNYV

Operációs rendszerek BSc

2022. tavasz féléves feladat

Készítette: Jónis Bálint Rudolf
Neptunkód: LN40BG

A feladat leírása:

- Írjon egy C programot, amely
egy szűlőprocessz révén készít egy gyermekprocesszt,
a gyermekben futtasson egy másik programot az exec() hívással(Environemnten keresztül kapja meg,
hogy mit indítson el a program), mely kiírja a PID-jét
és szülöje PID-jét, majd a szülő is kiírja mi a PID-je és a gyermeké PID-je.

A feladat elkészítésének lépései: program megírása

A futtatás eredménye: fork()-al létrehozzuk a gyerek processzt, kiíratjuk a 2 pidet, futtatjuk a gyerek processzben execl()-el az environmenten keresztül megkapott (jelen esetben LN40BG_openclose.c) programot, majd kiíratjuk a szülővel is a 2 pidet.

```
jb@jb:~/LN408G$ ./beadando
Szulo vagyok, pidem:2047, gyerek pidje:2048
Gyerek vagyok, pidem:2048, szulom pidje:2047
write() beirt: 48 byteot
lseek() helye: 0
read() olvasott 48 byteot, ami a kovetkezo: Jonis Balint Rudolf, mernokinformatikus,
LN408G
```

A feladat leírása:

1. Adott egy igény szerinti lapozást használó számítógéprendszer, melyben futás közben egy processz számára a következő laphivatkozással lehet hivatkozni: 6, 5, 4, 3, 5, 6, 2, 8, 5, 6, 5, 4, 7, 8, 4, 5 6, 5, 5, 8

Memóriakeret (igényelt lapok): 3, ill. 4 memóriakeret.

Készítse el a laphivatkozások betöltését külön-külön táblázatba 3. ill. 4 memóriakeret esetén.

Mennyi laphiba keletkezik az alábbi algoritmusok esetén: SC, LRU?

Hasonlitsa össze és magyarázza az eredményeket!

A feladat elkészítésének lépései: laphivatkozások, -hibák kiszámolása táblázatban

Ha több lapunk van, kevesebbszer kell kilaposzni, így kevesebb laphiba keletkezik, illetve az SC FIFO és LRU algoritmusokkal közel azonos mennyiségekű laphiba keletkezik.