

Operációs rendszerek BSc

2. Gyak.

2022. 03. 24.

Készítette:

Nagy Gergely Bsc

Mérnökinformatikus

G13RFP

Miskolc, 2022

Operációs rendszerek – 2. Gyakorlat

Linux OS - Rendszerhívások, processz ütemezés

Töltse fel az aktuális mappába: **Neptunkod_....**

Jegyzőkönyv neve: *Neptunkod_Gyak2.pdf és a* forrás file-k.

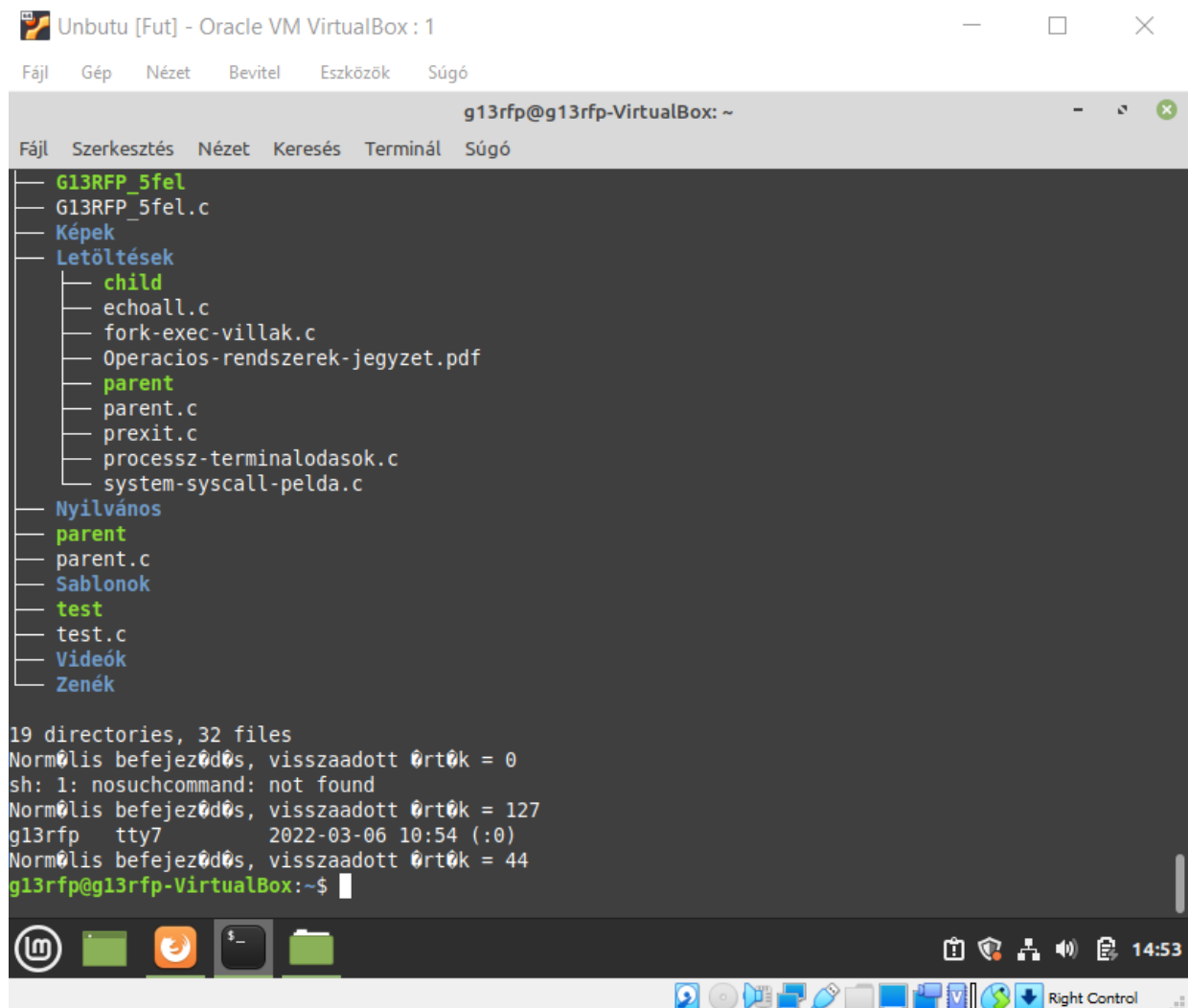
Határidő: 2022.04.07.

Feladatok

„1. A `system()` rendszerhívással hajtson végre létező és nem létező parancsot, és vizsgálja a visszatérési értéket, magyarázza egy-egy mondattal

A fordítás/futtatás után készítsen egy képernyőképet (minden parancs esetén) és illessze be a dokumentumba.

Mentés: *neptunkod_1fel.c*



```
Unbutu [Fut] - Oracle VM VirtualBox : 1
Fájl  Gép  Nézet  Bevitel  Eszközök  Súgó

g13rfp@g13rfp-VirtualBox: ~
Fájl  Szerkesztés  Nézet  Keresés  Terminál  Súgó

G13RFP_5fel
G13RFP_5fel.c
Képek
Letöltések
  child
  echoall.c
  fork-exec-villak.c
  Operacios-rendszerek-jegyzet.pdf
  parent
  parent.c
  prexit.c
  processz-terminalodasok.c
  system-syscall-pelda.c
Nyilvános
  parent
  parent.c
Sablonok
  test
  test.c
Videók
Zenék

19 directories, 32 files
Normális befejezés, visszaadott 0rtók = 0
sh: 1: nosuchcommand: not found
Normális befejezés, visszaadott 0rtók = 127
g13rfp  tty7      2022-03-06 10:54 (:0)
Normális befejezés, visszaadott 0rtók = 44
g13rfp@g13rfp-VirtualBox:~$
```

Magyarázat:

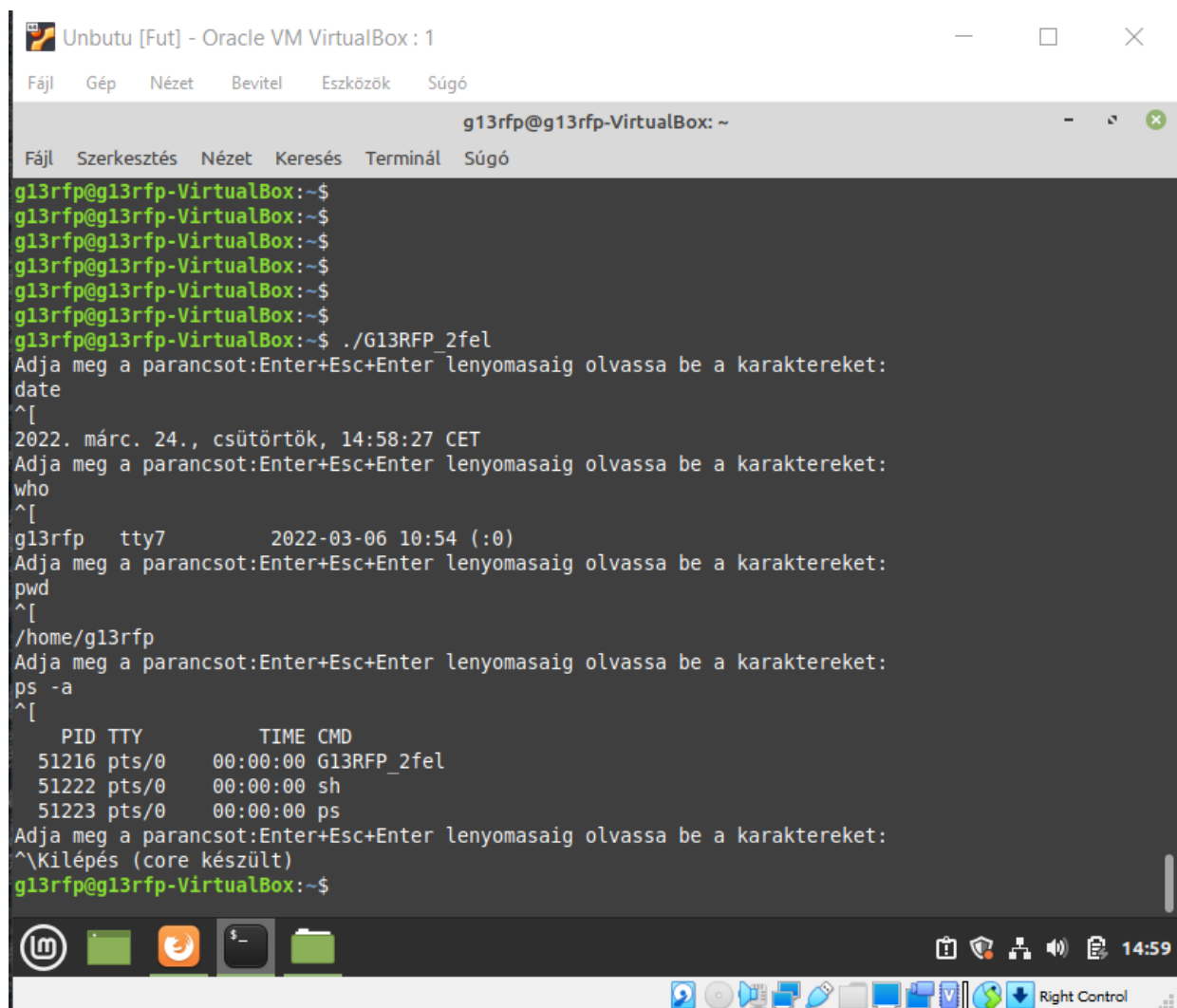
Státusz információk: WIF makrók

1. WIFEXITED (status): gyerek rendesen kilépett
 - WEXITSTATUS (status): visszaadja a kódot, amikor a gyermek kilép
2. WIFSIGNALED(status): gyermek kilépett, mert a jel nem lett elkapva
 - WTERMSIG(status): megadja a termináló jel számát

2. Írjon programot, amely billentyűzetről bekér Unix parancsokat és végrehajtja őket, majd kiírja a szabványos kimenetre. (pl.: amit bekér: date, pwd, who etc.; kilépés: CTRL-\) - magyarázza egy-egy mondattal

A fordítás/futtatás után készítsen egy képernyőképet (minden parancs esetén) és illessze be a dokumentumba.

Mentés: *neptunkod_2fel.c*



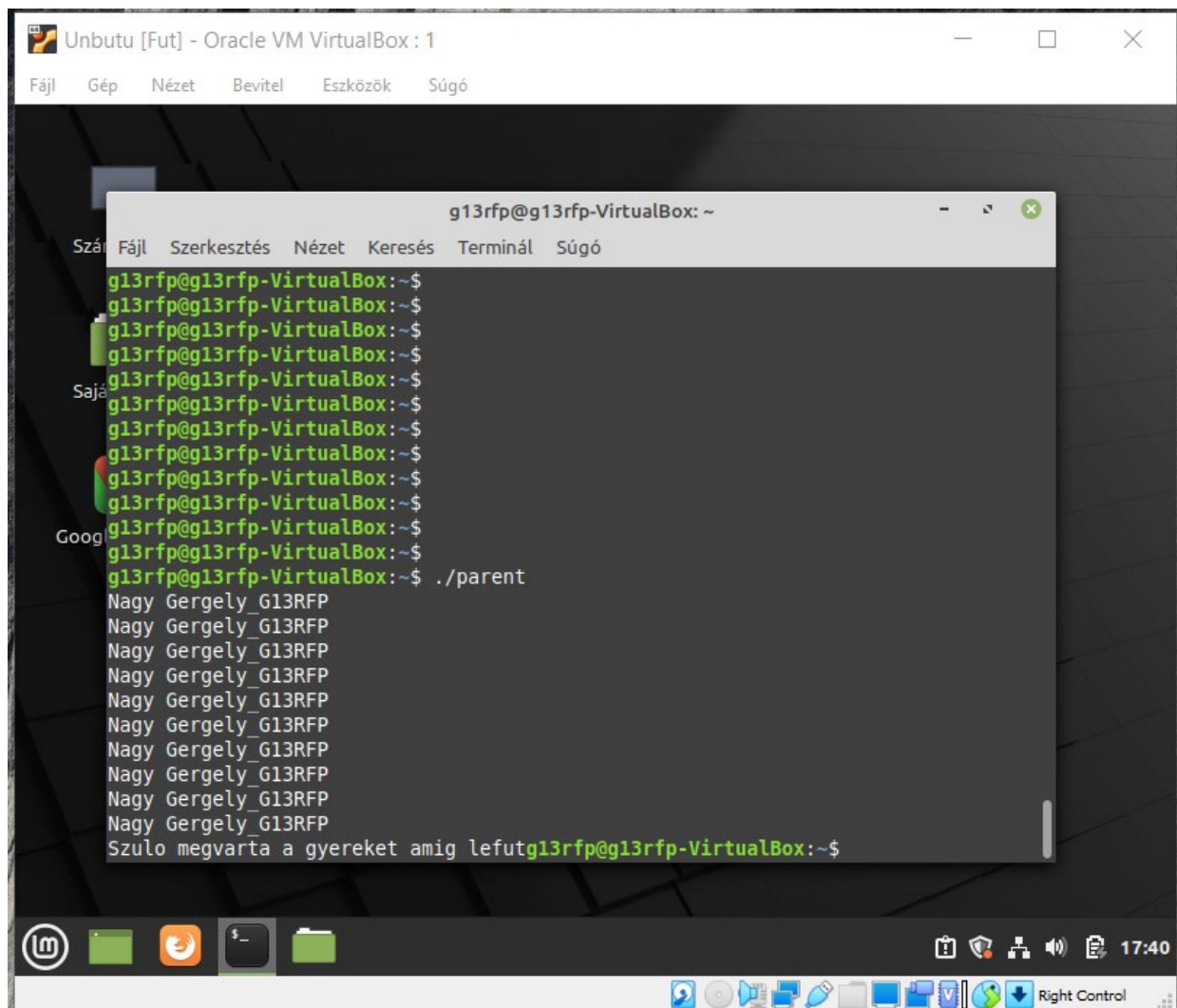
```
g13rfp@g13rfp-VirtualBox: ~  
g13rfp@g13rfp-VirtualBox: ~$  
g13rfp@g13rfp-VirtualBox: ~$  
g13rfp@g13rfp-VirtualBox: ~$  
g13rfp@g13rfp-VirtualBox: ~$  
g13rfp@g13rfp-VirtualBox: ~$  
g13rfp@g13rfp-VirtualBox: ~$ ./G13RFP_2fel  
Adja meg a parancsot:Enter+Esc+Enter lenyomasaig olvassa be a karaktereket:  
date  
^[  
2022. márc. 24., csütörtök, 14:58:27 CET  
Adja meg a parancsot:Enter+Esc+Enter lenyomasaig olvassa be a karaktereket:  
who  
^[  
g13rfp  tty7          2022-03-06 10:54 (:0)  
Adja meg a parancsot:Enter+Esc+Enter lenyomasaig olvassa be a karaktereket:  
pwd  
^[  
/home/g13rfp  
Adja meg a parancsot:Enter+Esc+Enter lenyomasaig olvassa be a karaktereket:  
ps -a  
^[  
  PID TTY          TIME CMD  
  51216 pts/0        00:00:00 G13RFP_2fel  
  51222 pts/0        00:00:00 sh  
  51223 pts/0        00:00:00 ps  
Adja meg a parancsot:Enter+Esc+Enter lenyomasaig olvassa be a karaktereket:  
^\\Kilépés (core készült)  
g13rfp@g13rfp-VirtualBox: ~$
```

Magyarázat:

A program addig kéri be a parancsokat és futtatja azokat amíg CTRL+ALT+GR+\\ billentyű kombinációval termináljuk.

3. Készítsen egy `parent.c` és a `child.c` programokat. A `parent.c` elindít egy gyermek processzt, ami különbözik a szülőtől. A szülő megvárja a gyermek lefutását. A gyermek szöveget ír a szabványos kimenetre (10-ször) (pl. a hallgató neve és a neptunkód)! - magyarázza egy-egy mondattal. A fordítás/futtatás után készítsen egy képernyőképet (minden parancs esetén) és illessze be a dokumentumba.

Mentés: `parent.c`, ill. `child.c`



```
g13rfp@g13rfp-VirtualBox: ~$
g13rfp@g13rfp-VirtualBox: ~$
g13rfp@g13rfp-VirtualBox: ~$
g13rfp@g13rfp-VirtualBox: ~$
g13rfp@g13rfp-VirtualBox: ~$
g13rfp@g13rfp-VirtualBox: ~$
g13rfp@g13rfp-VirtualBox: ~$
g13rfp@g13rfp-VirtualBox: ~$
g13rfp@g13rfp-VirtualBox: ~$
g13rfp@g13rfp-VirtualBox: ~$
g13rfp@g13rfp-VirtualBox: ~$ ./parent
Nagy Gergely_G13RFP
Nagy Gergely_G13RFP
Nagy Gergely_G13RFP
Nagy Gergely_G13RFP
Nagy Gergely_G13RFP
Nagy Gergely_G13RFP
Nagy Gergely_G13RFP
Nagy Gergely_G13RFP
Nagy Gergely_G13RFP
Nagy Gergely_G13RFP
Szulo megvárta a gyereket amig lefutg13rfp@g13rfp-VirtualBox: ~$
```

Magyarázat:

A `parent.c` elindít egy gyermek processzt (`child.c`) ami 10-szer kiírja a nevem+neptunkódom, megvárja annak befejezését.

4. A `fork()` rendszerhívással hozzon létre egy gyerek processzt-t és abban hívjon meg egy `exec` családbeli rendszerhívást (pl. `execlp`). A szülő várja meg a gyerek futását! - magyarázza egy-egy mondattal.

A fordítás/futtatás után készítsen egy képernyőképet (minden parancs esetén) és illessze be a dokumentumba.

```
g13rfp@g13rfp-VirtualBox: ~  
Fájl Szerkesztés Nézet Keresés Terminál Súgó  
PID TTY TIME CMD  
54068 pts/0 00:00:00 bash  
54513 pts/0 00:00:00 G13RFP_4fel  
54514 pts/0 00:00:00 child2  
54515 pts/0 00:00:00 sh  
54516 pts/0 00:00:00 ps  
g13rfp@g13rfp-VirtualBox:~$ ./G13RFP_4fel  
54517.sz.szulo var a 54518.sz.gyerekre.  
gyerek pid = 54518  
PID TTY TIME CMD  
54068 pts/0 00:00:00 bash  
54517 pts/0 00:00:00 G13RFP_4fel  
54518 pts/0 00:00:00 child2  
54519 pts/0 00:00:00 sh  
54520 pts/0 00:00:00 ps  
g13rfp@g13rfp-VirtualBox:~$ ./G13RFP_4fel  
54521.sz.szulo var a 54522.sz.gyerekre.  
gyerek pid = 54522  
PID TTY TIME CMD  
54068 pts/0 00:00:00 bash  
54521 pts/0 00:00:00 G13RFP_4fel  
54522 pts/0 00:00:00 child2  
54523 pts/0 00:00:00 sh  
54524 pts/0 00:00:00 ps  
g13rfp@g13rfp-VirtualBox:~$
```

Magyarázat:

A program `fork()` rendszerhívással léterhoz egy gyermek processzt ami elindít egy programot(`child2.c`) amiben `execl()` hívással meghív egy shell `ps` parancsot.

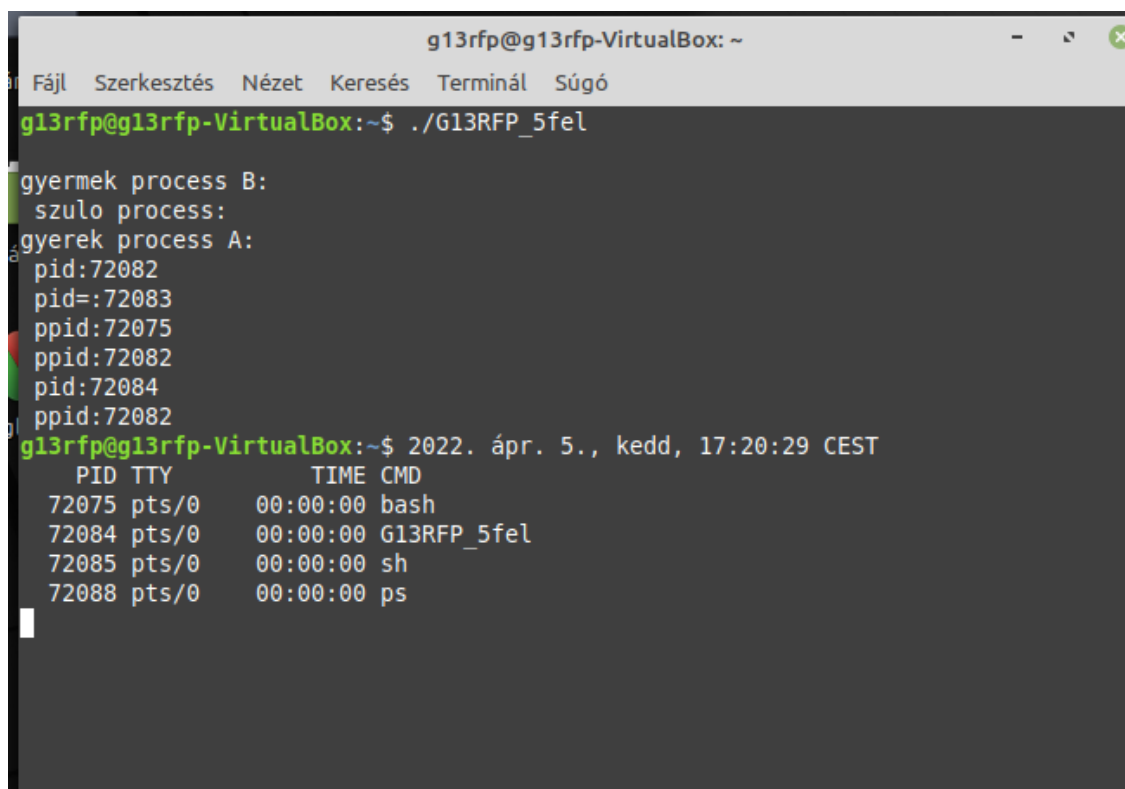
Mentés: *neptunkod_4fel.c*

5. A `fork()` rendszerhívással hozzon létre gyerekeket, várja meg és vizsgálja a befejeződési állapotokat (gyerekekben: `exit`, `abort`, nullával való osztás)! - magyarázza egy-egy mondatral!

A fordítás/futtatás után készítsen egy képernyőképet (minden parancs esetén) és illessze be a dokumentumba.

Mentés: *neptunkod_5fel.c*

exit—normál terminálódás:



```
g13rfp@g13rfp-VirtualBox: ~  
Fájl Szerkesztés Nézet Keresés Terminál Súgó  
g13rfp@g13rfp-VirtualBox:~$ ./G13RFP_5fel  
gyermek process B:  
szulo process:  
gyerek process A:  
pid:72082  
pid=:72083  
ppid:72075  
ppid:72082  
pid:72084  
ppid:72082  
g13rfp@g13rfp-VirtualBox:~$ 2022. ápr. 5., kedd, 17:20:29 CEST  
PID TTY      TIME CMD  
72075 pts/0      00:00:00 bash  
72084 pts/0      00:00:00 G13RFP_5fel  
72085 pts/0      00:00:00 sh  
72088 pts/0      00:00:00 ps
```

abort:

```
g13rfp@g13rfp-VirtualBox: ~
Fájl Szerkesztés Nézet Keresés Terminál Súgó
g13rfp@g13rfp-VirtualBox:~$
g13rfp@g13rfp-VirtualBox:~$ gedit G13RFP_5fel.c
g13rfp@g13rfp-VirtualBox:~$ gcc -o G13RFP_5fel G13RFP_5fel.c
g13rfp@g13rfp-VirtualBox:~$ ./G13RFP_5fel
gyermek process B:

szulo process:
gyerek process A:
pid:72156
pid=:72157
ppid:72075
ppid:72156
pid:72158
ppid:72156
2022. ápr. 5., kedd, 17:31:34 CEST
PID TTY      TIME CMD
 72075 pts/0      00:00:00 bash
 72156 pts/0      00:00:00 G13RFP_5fel
 72157 pts/0      00:00:00 G13RFP_5fel <defunct>
 72158 pts/0      00:00:00 G13RFP_5fel
 72159 pts/0      00:00:00 sh
 72163 pts/0      00:00:00 ps
Félbeszakítva (core készült)
g13rfp@g13rfp-VirtualBox:~$
```

6. Adott a következő ütemezési feladat, amit a FCFS, SJF és Round Robin (RR) ütemezési algoritmus használatával készítsen el (külön-külön táblázatba):

I. Határozza meg FCFS és SJF esetén

- a.) A befejezési időt?
- b.) A várakozási/átlagos várakozási időt?
- c.) Ábrázolja Gantt diagram segítségével az *aktív/várakozó processzek* futásának menetét.

Megj.: a Gantt diagram ábrázolása szerkesztő program segítségével vagy Excel programmal.

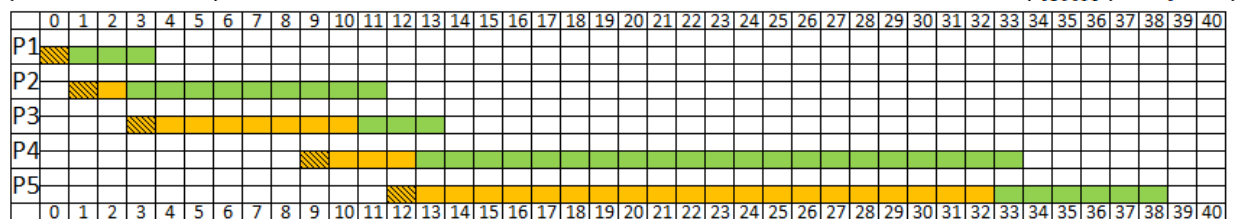
Mentés: *neptunkod_6fel.pdf*

FCFS

FCFS	Érkezés	CPU idő
P1	0	3
P2	1	8
P3	3	2
P4	9	20
P5	12	5

FCFS megoldás

FCFS					
Folyamat	Érkezési idő	CPU igény	Kezdési idő	Befejezési i.	Várakozási i.
P1	0	3	0	3	0
P2	1	8	3	11	2
P3	3	2	11	13	8
P4	9	20	13	33	4
P5	12	5	33	38	21
				átl.:	7

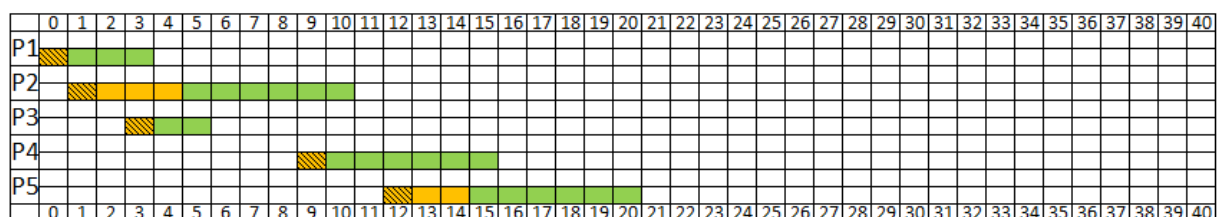


SJF

SJF	Érkezés	CPU idő
P1	0	3
P2	1	5
P3	3	2
P4	9	5
P5	12	5

SJF megoldás

SJF					
Folyamat	Érkezési idő	CPU igény	Kezdési idő	Befejezési i.	Várakozási i.
P1	0	3	0	3	0
P2	1	5	5	10	4
P3	3	2	3	5	0
P4	9	5	10	15	1
P5	12	5	15	20	3
				átl.:	1.6



II. Round Robin (RR) esetén

a.) Ütemezze az adott időszakot (5ms) alapján az egyes processzek (befejezési és várakozási/átlagos várakozási idő) paramétereit (ms)!

b.) A rendszerben lévő processzek végrehajtásának sorrendjét?

c.) Ábrázolja Gantt diagram segítségével az *aktív/várakozó processzek* futásának menetét!”

Megj.: a Gantt diagram ábrázolása szerkesztő program segítségével vagy Excel programmal.

Mentés: *neptunkod6fel.pdf*

RR: 5ms	Érkezés	CPU idő
P1	0	3
P2	1	8
P3	3	2
P4	9	20
P5	12	5

RR megoldás

RR:5ms	átl.: 1,6					
Folyamat	Érkezési idő	CPU igény	Kezdési idő	Befejezési i.	Várakozási i.	Várakozó process
P1	0	3	0	3	0	p2,p3
P2	1	8	3	8	2	p2,p4
P3	3	2	8	10	5	p2,p4
P2*	8	3	10	13	2	p4,p5
P4	9	20	13	18	4	p4,p5
P5	12	5	18	23	6	p4
P4*	18	15	23	38	7	-
átl.: 3,714						

