

**Operációs rendszerek BSc**  
**2. Gyak.**  
**2022. 03. 24.**

Készítette:  
Forgó Krisztián  
Mérnökinformatikus  
BSC - levelező  
DQ5AAK

Miskolc, 2022

## Operációs rendszerek – 2. Gyakorlat

### Linux OS - Rendszerhívások, processz ütemezés

Tölts 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 érteket, magyarázza egy-egy mondattal

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

Mentés: *neptunkod\_1fel.c*

```
forgo@forgo-VirtualBox:~/DQ5AAK/Gyak2$ ./DQ5AAK_1fel
.
├── child2.c
└── child.c
DQ5AAK_1fel
├── DQ5AAK_1fel.c
├── DQ5AAK_2fel.c
├── DQ5AAK_4fel.c
└── DQ5AAK_5fel.c
parent.c

0 directories, 8 files
Normalis befejezés, visszaadott 0rtók = 0
sh: 1: nosuchcommand: not found
Normalis befejezés, visszaadott 0rtók = 127
forgo    tty7        2022-03-03 16:49 (:0)
Normalis befejezés, visszaadott 0rtók = 44
forgo@forgo-VirtualBox:~/DQ5AAK/Gyak2$
```

#### Magyarázat:

Státusz információk: WIF makrók

1. WIFEXITED (status): gyerek rendesen kilépett
  - WEXITSTATUS (status): visszaadja a kódöt, amikor a gyermek kilépett
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*

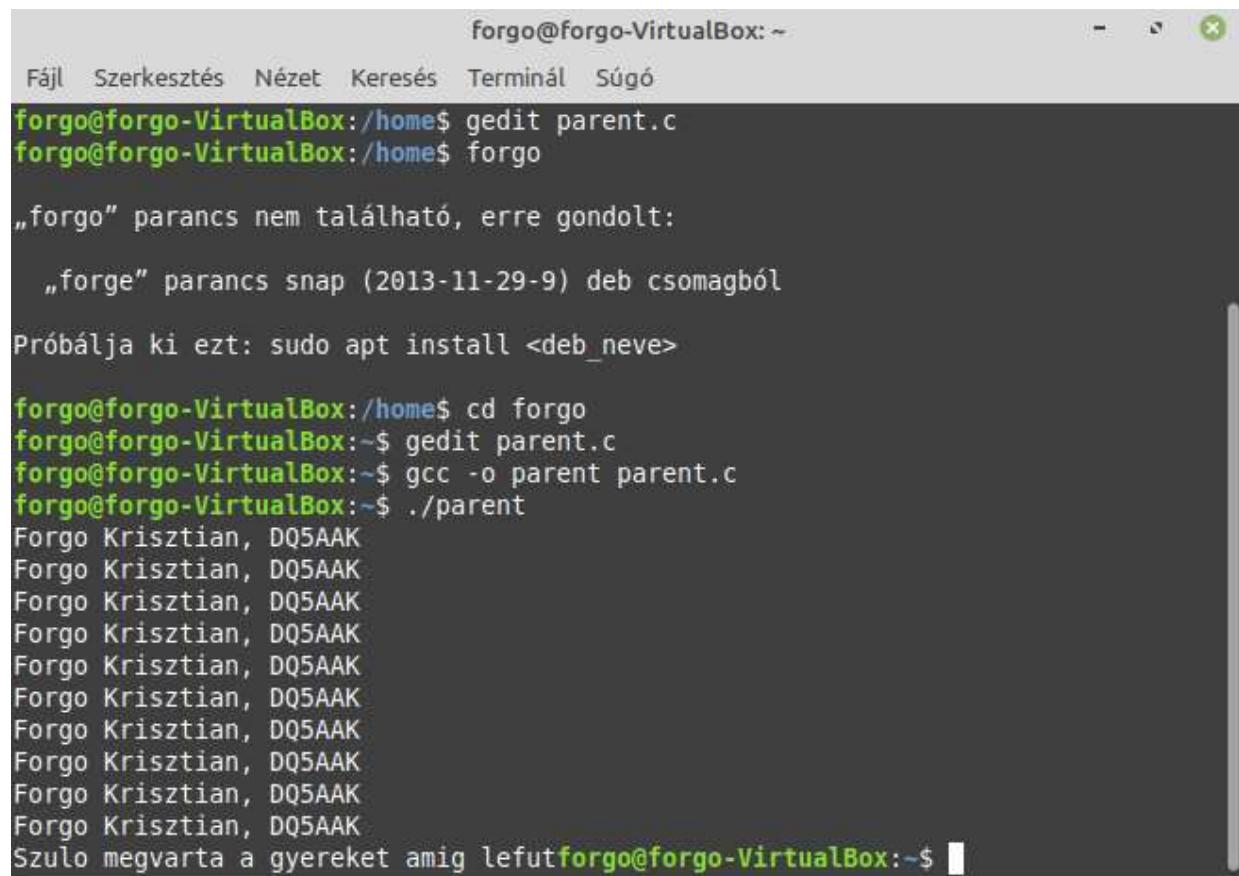
```
forgo@forgo-VirtualBox:~/DQ5AAK/Gyak2$ ./DQ5AAK_2fel
Adja meg a parancsot:Enter+Esc+Enter lenyomasaig olvassa be a karaktereket:
who
^[
forgo    tty7          2022-03-03 16:49 (:0)
Adja meg a parancsot:Enter+Esc+Enter lenyomasaig olvassa be a karaktereket:
```

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*



forgo@forgo-VirtualBox: ~

Fájl Szerkesztés Nézet Keresés Terminál Súgó

```
forgo@forgo-VirtualBox:/home$ gedit parent.c
forgo@forgo-VirtualBox:/home$ forgo
„forgo” parancs nem található, erre gondolt:
„forge” parancs snap (2013-11-29-9) deb csomagból
Próbálja ki ezt: sudo apt install <deb_neve>

forgo@forgo-VirtualBox:/home$ cd forgo
forgo@forgo-VirtualBox:~$ gedit parent.c
forgo@forgo-VirtualBox:~$ gcc -o parent parent.c
forgo@forgo-VirtualBox:~$ ./parent
Forgo Krisztian, DQ5AAK
Szulo megvarta a gyereket amig lefutforgo@forgo-VirtualBox:~$
```

#### Magyarázat:

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

4. A `fork()` rendszerhívással hozzon létre egy gyerek processzt 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.

```
forgo@forgo-VirtualBox:~$ ./DQ5AAK_4fel
22620.sz.szulo var a 22621.sz.gyerekre.

gyerek pid = 22621
 PID TTY          TIME CMD
22579 pts/0    00:00:00 bash
22620 pts/0    00:00:00 DQ5AAK_4fel
22621 pts/0    00:00:00 child2
22622 pts/0    00:00:00 sh
22623 pts/0    00:00:00 ps
forgo@forgo-VirtualBox:~$
```

Magyarázat:

A program `fork()` rendszerhívással léterhoz egy gyermek processzt ami elindít egy programot(`chid2.c`) amiben `execl()` hívással meghív egy shell `ps` parancsot.

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

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

Mentés: `neptunkod_5fel.c`

**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 aktiv/várakozó processzek futásának menetét.

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

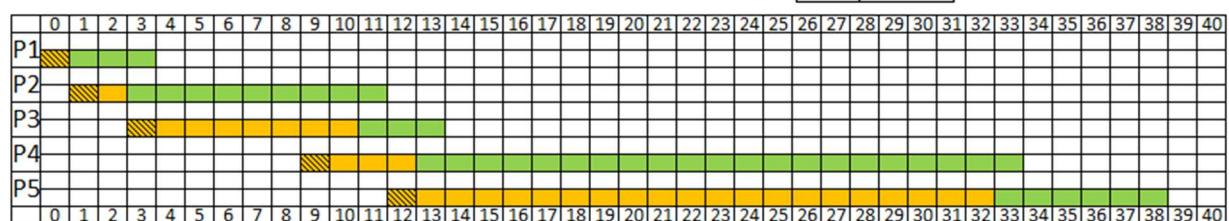
Mentés: *neptunkod\_6fel.pdf*

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
				átł.: 7	

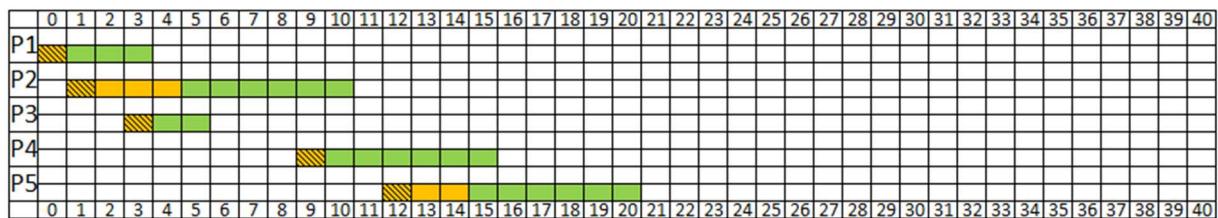


SJF

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

SJF megoldás

SFJ	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
					átł.:	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						
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	-
				átł.: 3,714		

