

Operációs rendszerek BSc

5. Gyak.

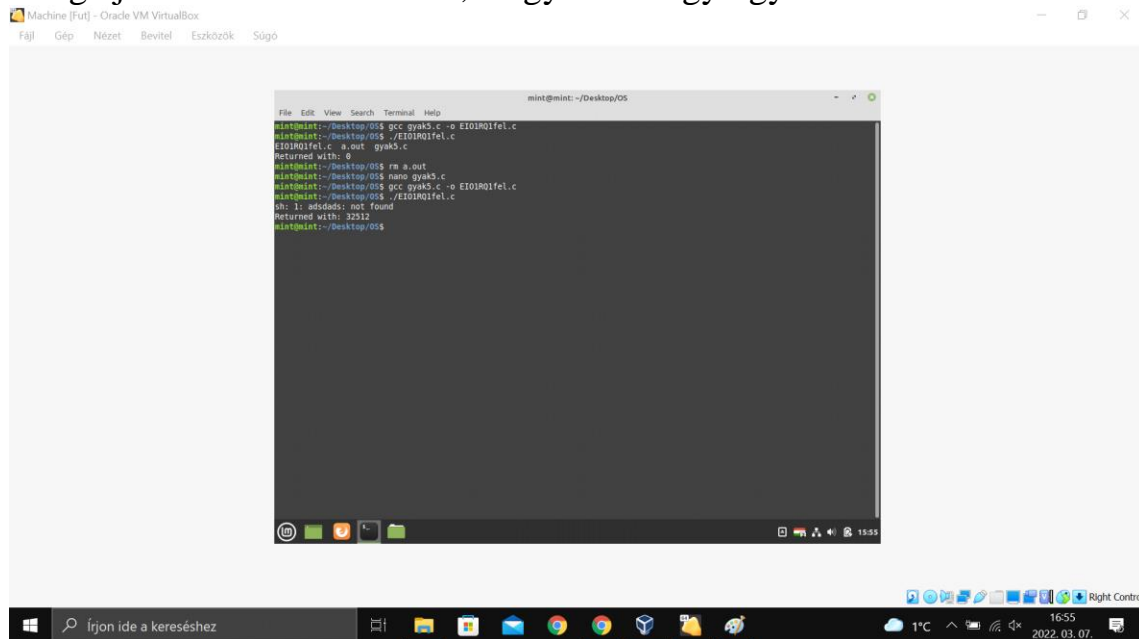
2022. 03. 07.

Készítette:

Nagy Balázs Bsc
Programtervező
informatikus
EIO1RQ

Miskolc, 2022

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

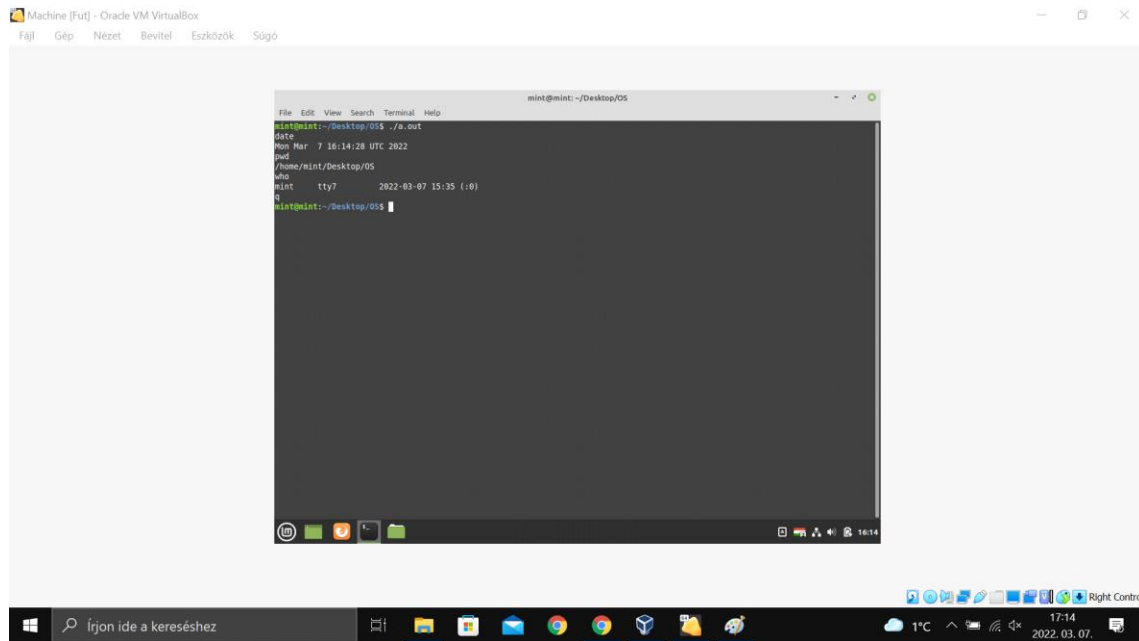


```
Machine [Fut] - Oracle VM VirtualBox
Fájl  Gép  Nézet  Bevitel  Eszközök  Súgó

mint@mint: ~/Desktop/OS
File Edit View Search Terminal Help
mint@mint:~/Desktop/OS$ gcc gyak5.c -o EIOIRQ1fel.c
mint@mint:~/Desktop/OS$ ./EIOIRQ1fel.c
EIOIRQ1fel.c: a.out: gyak5.c
returned with: 0
mint@mint:~/Desktop/OS$ rm a.out
mint@mint:~/Desktop/OS$ nano gyak5.c
mint@mint:~/Desktop/OS$ gcc gyak5.c -o EIOIRQ1fel.c
mint@mint:~/Desktop/OS$ ./EIOIRQ1fel.c
sh: 1: sdsdss: not found
returned with: 255
mint@mint:~/Desktop/OS$
```

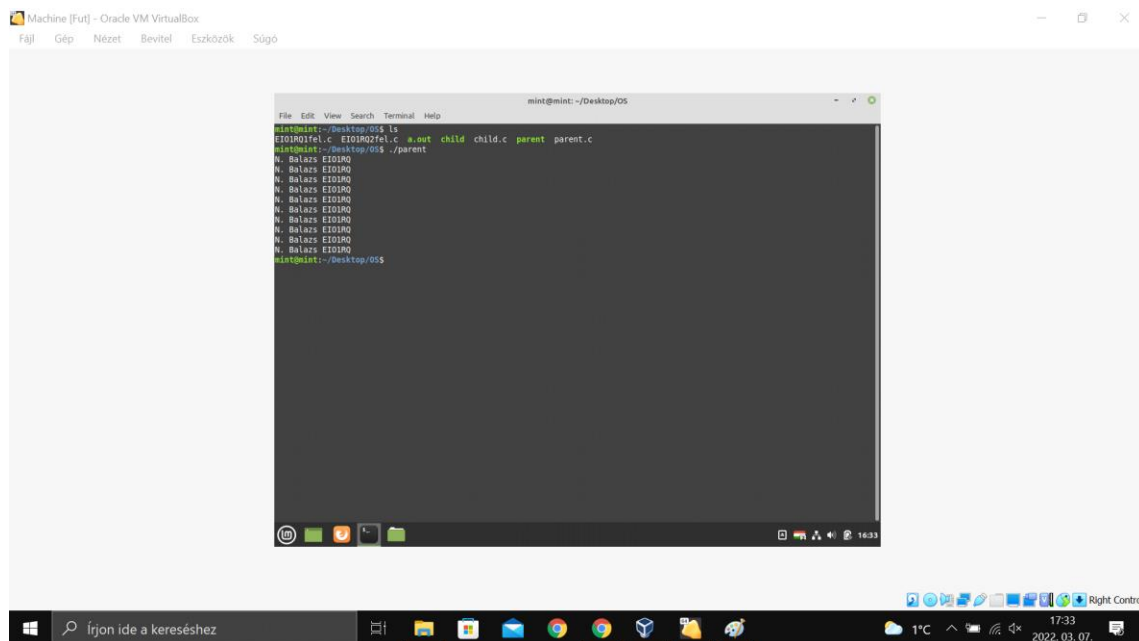
A rendszerhívás átadja a paraméterben megadott string-et, mint parancs.

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.



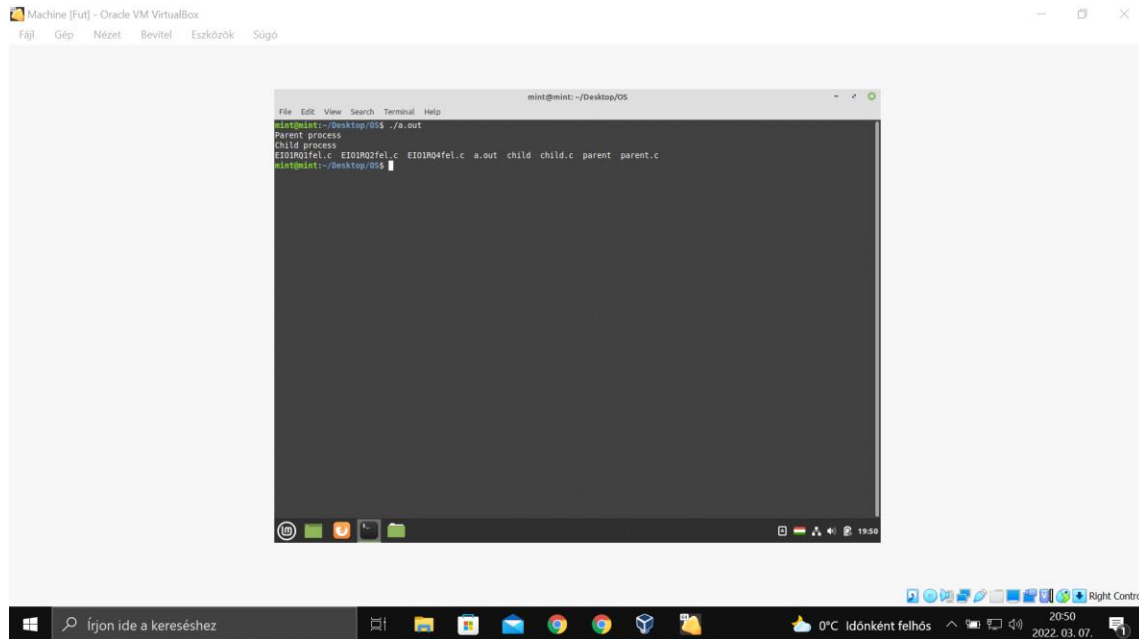
Egy végtelen ciklus, amely végrehajtja a parancsokat. A program leáll, ha a felhasználó *q*-t üt be.

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.



Parent.c futtatja a child.c programot, ami kiírja a megadott szöveget.

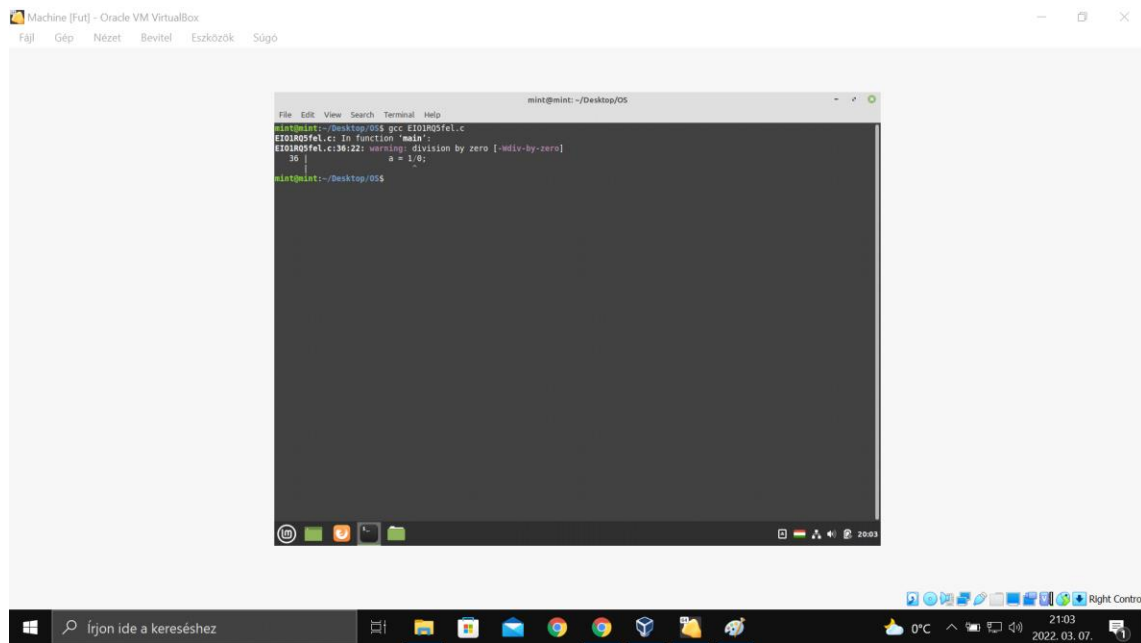
4. A *fork()* rendszerhívással hozzon létre egy gyerek processzt-t és abban hívjon meg egy *exec* családbeli rendszerhívást (pl. *execlp*). A szülő várja meg a gyerek futását! - magyarázza egy-egy mondattal.



```
mint@mint: ~/Desktop/OS
File Edit View Search Terminal Help
mint@mint:~/Desktop/OS$ ./a.out
Parent process
Child process
EIOIM01fel.c EIOIM02fel.c EIOIM04fel.c a.out child child.c parent parent.c
mint@mint:~/Desktop/OS$
```

Execlp parancsokat ad át a rendszernek; az execl-el szemben itt paraméterként nem kell megadni a teljes elérési utat, csupán a parancsot.

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

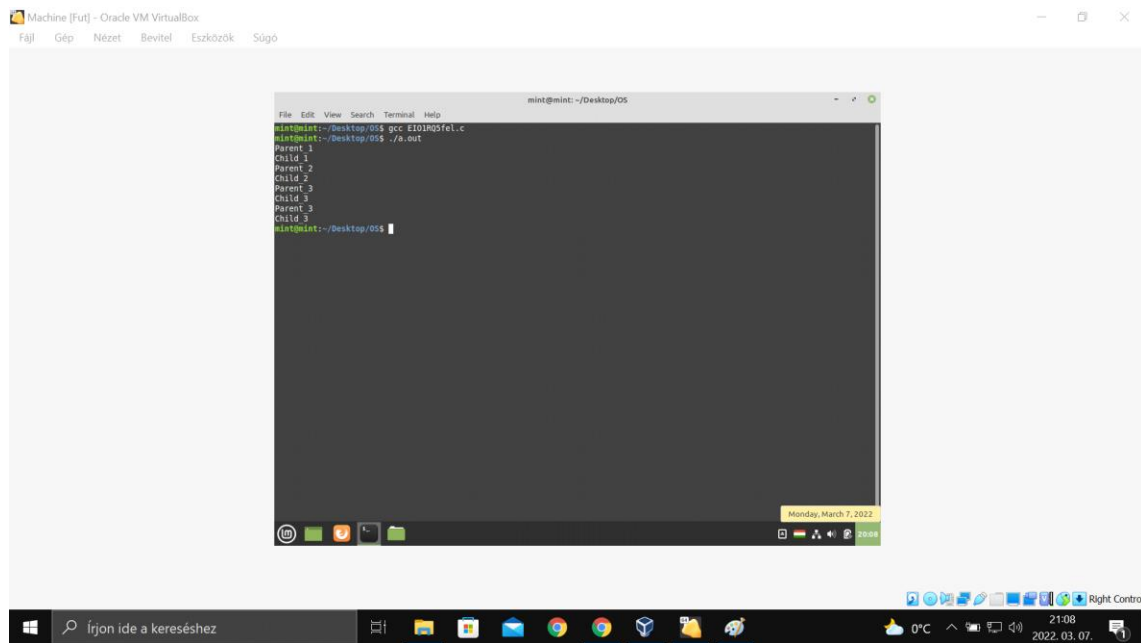


The screenshot shows a terminal window titled 'mint@mint: ~/Desktop/OS' within an Oracle VM VirtualBox environment. The terminal output is as follows:

```
mint@mint:~/Desktop/OS$ gcc EIO1Q05fel.c
EIO1Q05fel.c: In function 'main':
EIO1Q05fel.c:30:22: warning: division by zero [-Wdiv-by-zero]
   30 |         a = 1/0;
      |                  ^
mint@mint:~/Desktop/OS$
```

The Windows taskbar at the bottom shows the search bar with the text 'Írjon ide a kereséshez', several application icons, and system tray information including '0°C', '21:03', and '2022. 03. 07.'.

Nullával való osztás esetén a Compiler hibát jelez, így magát a program futtani sem lehet.



The screenshot shows the same terminal window as above, but now displaying the output of a program execution:

```
mint@mint:~/Desktop/OS$ gcc EIO1Q05fel.c
mint@mint:~/Desktop/OS$ ./a.out
Parent_1
Child_1
Parent_2
Child_2
Parent_3
Child_3
Parent_3
Child_3
mint@mint:~/Desktop/OS$
```

The Windows taskbar at the bottom shows the search bar with the text 'Írjon ide a kereséshez', several application icons, and system tray information including '0°C', '21:08', and '2022. 03. 07.'.

Ellenkező esetben, ha nincs compilation error a program lefut hibátlanul. A szülő mindig megvárja míg a gyermek befejezi a folyamatát és kilép.

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

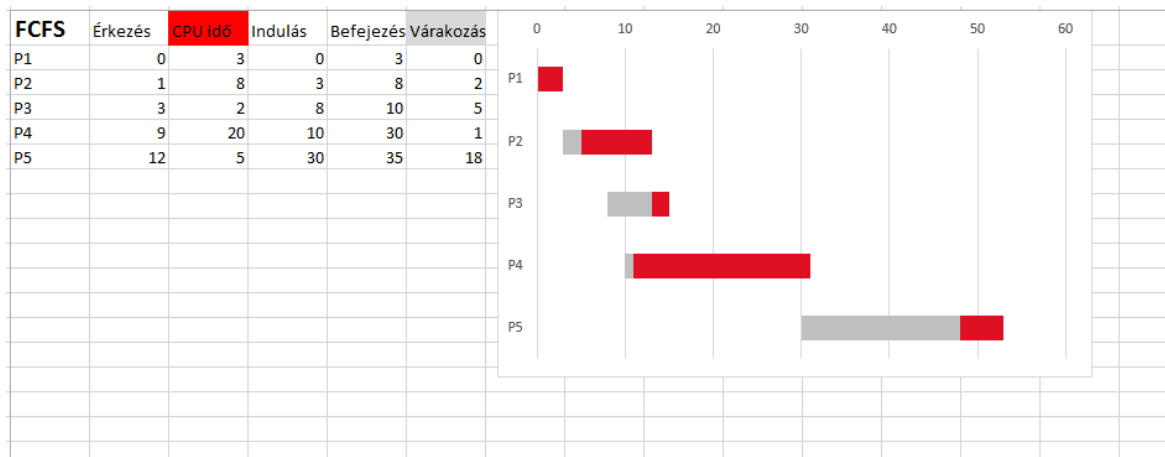
a.) A befejezési időt?

b.) A várakozási/átlagos várakozási időt?

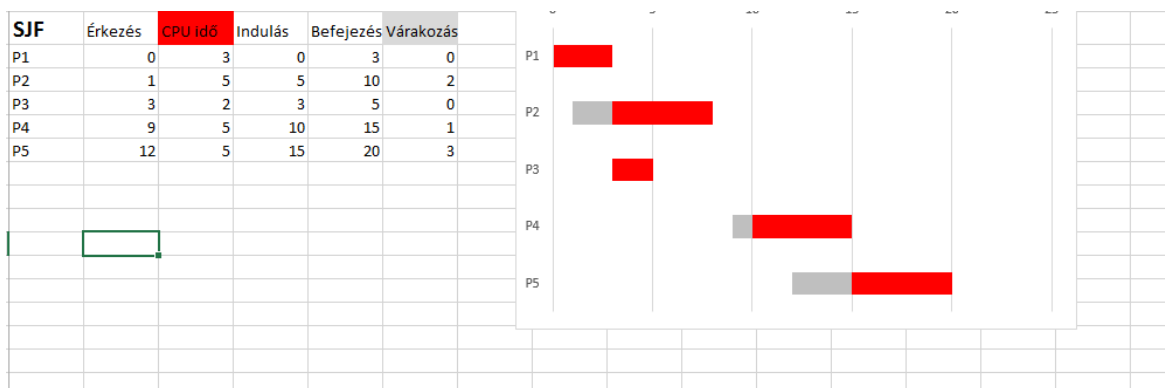
c.) Ábrázolja Gantt diagram segítségével az aktív/várakozó processzek futásának menetét.

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

FCFS



SJF



II. Round Robin (RR) esetén

- a.) Ütemezze az adott időszel (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!

5 ms							
RR	Érkezés	CPU idő	Indulás	Befejezés	Várakozás		
P1	0	3	0	3	0		
P2	1 8, 3		3, 10	8, 13	2, 2		
P3	3	2	8	10	5		
P4	9 20, 15, 10, 5		13, 23, 28, 33	18, 28, 33, 38	4, 5, 0, 0		
P5	12	5	18	23	6		

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