

Operációs rendszerek BSc

2. Gyak.
2022. 03. 18.

Készítette:
Kovács Donát
ME-GEIK

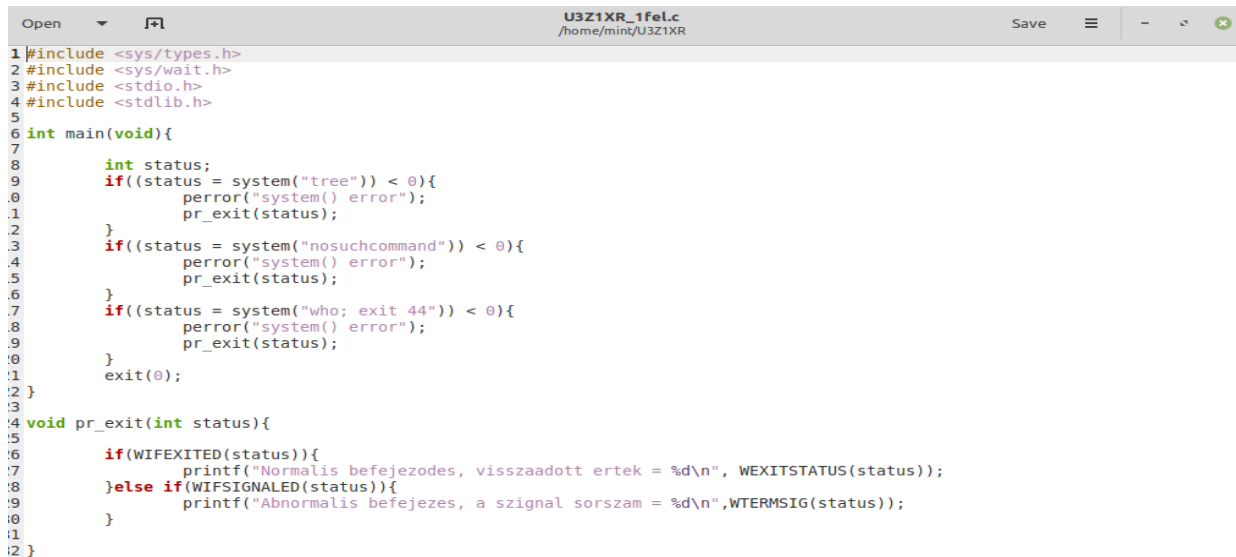
U3Z1XR

Miskolc, 2022

Feladatok

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

```
root@mint:/home/mint/U3Z1XR# ./U3Z1XR_1fel
Mon Mar 28 16:38:37 UTC 2022
sh: 1: nosuchcommand: not found
mint      tty7          2022-03-21 17:28 (:0)
```



```
1#include <sys/types.h>
2#include <sys/wait.h>
3#include <stdio.h>
4#include <stdlib.h>
5
6int main(void){
7
8    int status;
9    if((status = system("tree")) < 0){
10        perror("system() error");
11        pr_exit(status);
12    }
13    if((status = system("nosuchcommand")) < 0){
14        perror("system() error");
15        pr_exit(status);
16    }
17    if((status = system("who; exit 44")) < 0){
18        perror("system() error");
19        pr_exit(status);
20    }
21    exit(0);
22}
23
24void pr_exit(int status){
25
26    if(WIFEXITED(status)){
27        printf("Normalis befejezodes, visszaadott ertek = %d\n", WEXITSTATUS(status));
28    }else if(WIFSIGNALED(status)){
29        printf("Abnormalis befejezes, a szignal sorszam = %d\n", WTERMSIG(status));
30    }
31}
32}
```

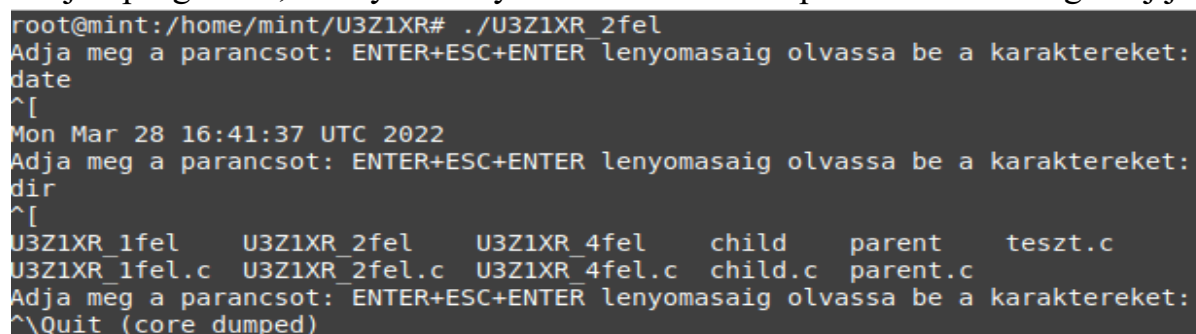
WIFEXITED(status): gyerek rendesen kilépett.

WEXITSTATUS(status): visszaadja a kódot, amikor a gyermek kilép.

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



```
root@mint:/home/mint/U3Z1XR# ./U3Z1XR_2fel
Adja meg a parancsot: ENTER+ESC+ENTER lenyomasaig olvassa be a karaktereket:
date
^[
Mon Mar 28 16:41:37 UTC 2022
Adja meg a parancsot: ENTER+ESC+ENTER lenyomasaig olvassa be a karaktereket:
dir
^[
U3Z1XR_1fel  U3Z1XR_2fel  U3Z1XR_4fel  child  parent  teszt.c
U3Z1XR_1fel.c U3Z1XR_2fel.c U3Z1XR_4fel.c child.c parent.c
Adja meg a parancsot: ENTER+ESC+ENTER lenyomasaig olvassa be a karaktereket:
^QQuit (core dumped)
```

őket, majd kiírja a szabványos kimenetre. (pl.: amit bekér: date, pwd, who etc.; kilépés: CTRL-\\)

A program addig kér 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)!

```
root@mint:/home/mint/U3Z1XR# ./parent
root@mint:/home/mint/U3Z1XR# Név: Kovács Donát, Neptunkód: U3Z1XR
Név: Kovács Donát, Neptunkód: U3Z1XR
Név: Kovács Donát, Neptunkód: U3Z1XR
Név: Kovács Donát, Neptunkód: U3Z1XR
Név: Kovács Donát, Neptunkód: U3Z1XR
Név: Kovács Donát, Neptunkód: U3Z1XR
Név: Kovács Donát, Neptunkód: U3Z1XR
Név: Kovács Donát, Neptunkód: U3Z1XR
Név: Kovács Donát, Neptunkód: U3Z1XR
Név: Kovács Donát, Neptunkód: U3Z1XR
```

A parent.c elindít egy gyermek processzt (child.c) ami 10-szer kiírja a nevet és neptunkódot.

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.

```
root@mint:/home/mint/U3Z1XR# ./U3Z1XR_4fel
Ez egy parent process-> ParentPID=5618
Varjon amig befejezodik a child process...
Ez egy parent process-> ParentPID=5618, ez egy child process-> PID=5619
A child process befejezodott...
```

A program fork() rendszerhívással létrehoz egy child 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 (gyerekekben: exit, abort, nullával való osztás)! - magyarázza egy-egy mondattal!

```
root@mint:/home/mint# ./U3Z1XR_5fel
Parent process:
pid:5478
ppid:5302
root@mint:/home/mint# Child process A:
pid:5480
ppid:1083
Child process B:
pid=5479
ppid:1083
Mon Mar 28 16:37:25 UTC 2022
  PID TTY          TIME CMD
  5300 pts/0        00:00:00 sudo
  5301 pts/0        00:00:00 su
  5302 pts/0        00:00:00 bash
  5480 pts/0        00:00:00 U3Z1XR_5fel
  5481 pts/0        00:00:00 sh
  5483 pts/0        00:00:00 ps
```

