

Jegyzőkönyv
Operációs
rendszerek
BSc
2022. tavasz
féléves feladat

Timkó András
HZS05V
ge-BGI

A feladat leírása:

Írjon C nyelvű programot, ami:

létrehoz két gyermekprocesszt

ezek a gyermekprocesszek létrehoznak 3-3 további gyereket

ezek az unokák varakoznak néhány másodpercet és szunjenek meg

a szülők várják meg a gyerekek befejeződését és csak utána szunjenek meg

Vagyis, megvárja a főprocessz amíg a (2)gyerekei elvégzik a feladatukat és azok is megvárják még az őket (3-3)gyerekei elvégzik és csak az után fejeződnek be.

A feladat elkészítésének lépései:

```
//3 gyerek forkolása ciklusban;  
//2 szülő hozzáadása egy másik ciklusban;  
//szülők várják meg a gyerekeiket wait(NULL);  
//ne maradjon gyerek processz és sikeresen kilépjen a program;
```

A futtatás eredménye:

root@elegemvan:/home/elegemvan/Desktop/HZS05V0sGyak2/OSSemTask

```
Szulo procesz => PID=7367  
Varok hogy a gyerek befejezze...  
Gyerek procesz => PPID=7367, PID=7368  
a gyerek procesz befejezte.  
Szulo procesz => PID=7367  
Varok hogy a gyerek befejezze...  
Gyerek procesz => PPID=7367, PID=7369  
a gyerek procesz befejezte.  
Szulo procesz => PID=7367  
Varok hogy a gyerek befejezze...  
Gyerek procesz => PPID=7367, PID=7370  
a gyerek procesz befejezte.
```

```
Szulo procesz => PID=7371  
Varok hogy a gyerek befejezze...  
Gyerek procesz => PPID=7371, PID=7372  
a gyerek procesz befejezte.  
Szulo procesz => PID=7371  
Varok hogy a gyerek befejezze...  
Gyerek procesz => PPID=7371, PID=7373  
a gyerek procesz befejezte.  
Szulo procesz => PID=7371  
Varok hogy a gyerek befejezze...  
Gyerek procesz => PPID=7371, PID=7374  
a gyerek procesz befejezte.
```

```
Szulo proceszek befejeztek.  
root@elegemvan:/home/elegemvan/Desktop/HZS05V0sGyak2/OSSemTask_HZS05V#
```

```
G: > EGYETEM > 2021-22-2 > OS > HZS05V0sGyak > OSSemTask_HZS05V > C hzs05v.c > main(void)  
1  #include <stdio.h>  
2  #include <stdlib.h>  
3  #include <sys/types.h>  
4  #include <sys/wait.h>  
5  #include <unistd.h>  
6  #include <signal.h>  
7  
8  int main(void)  
9  {  
10     pid_t pid_par;  
11     pid_t pid;  
12  
13     for (size_t i = 1; i <= 2; i++)  
14     {  
15         pid_par = fork();  
16  
17         if (pid_par != 0)  
18         {  
19             //szulo  
20             printf("\n");  
21             wait(NULL);  
22             exit(0);  
23         }  
24         else  
25         {  
26             //gyerek  
27             printf("\n");  
28             for (int i = 1; i <= 3; i++)  
29             {  
30                 pid = fork();  
31  
32                 if (pid == 0)  
33                 {  
34                     //gyerek(unoka)  
35                     printf("Gyerek procesz => PPID=%d, PID=%d\n", getppid(), getpid());  
36                     //var 2 mp-t es befejezi  
37                     sleep(2);  
38                     exit(0);  
39                 }  
40                 else  
41                 {  
42                     //szulo  
43                     printf("Szulo procesz => PID=%d\n", getpid());  
44                     printf("Varok hogy a gyerek befejezze...\n");  
45                     wait(NULL);  
46                     printf("a gyerek procesz befejezte.\n");  
47                 }  
48             }  
49         }  
50     }  
51  
52     printf("\nSzulo proceszek befejeztek.\n");  
53     //megvarja mig mind a ket szulo befejezi  
54     wait(NULL);  
55     return(0);  
56 }  
57  
58  
59  
60  
61
```