

Operációs rendszerek BSc

5. Gyak.

2022. 03. 09.

Készítette:

Holczer Vanda Bsc

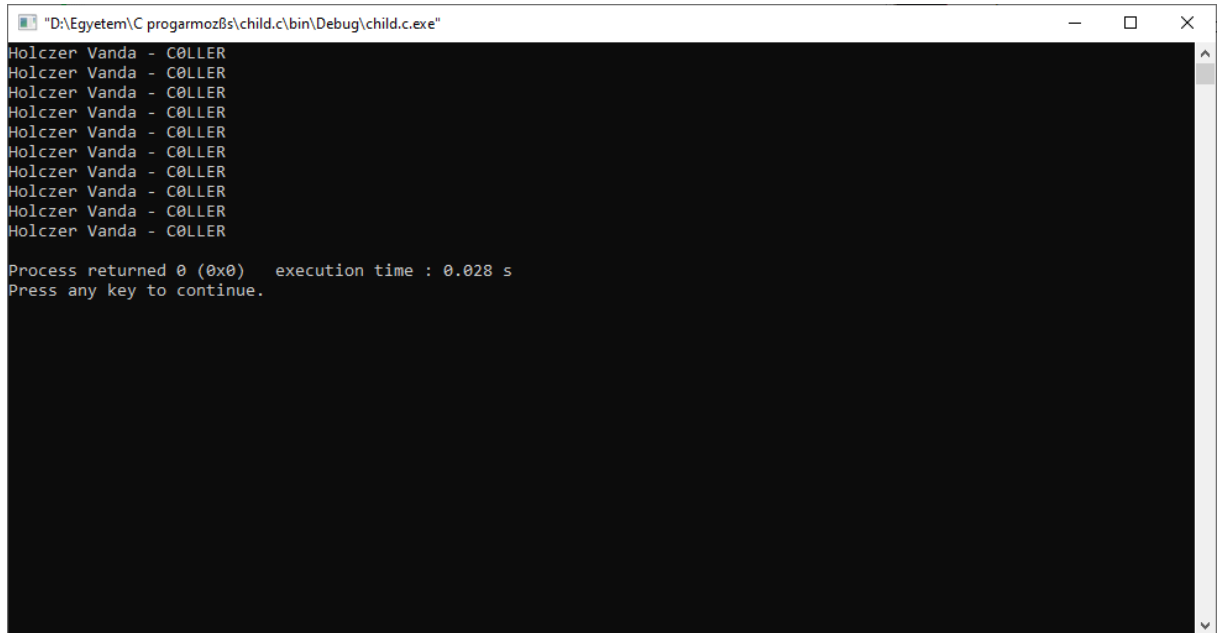
Programtervező informatikus

COLLER

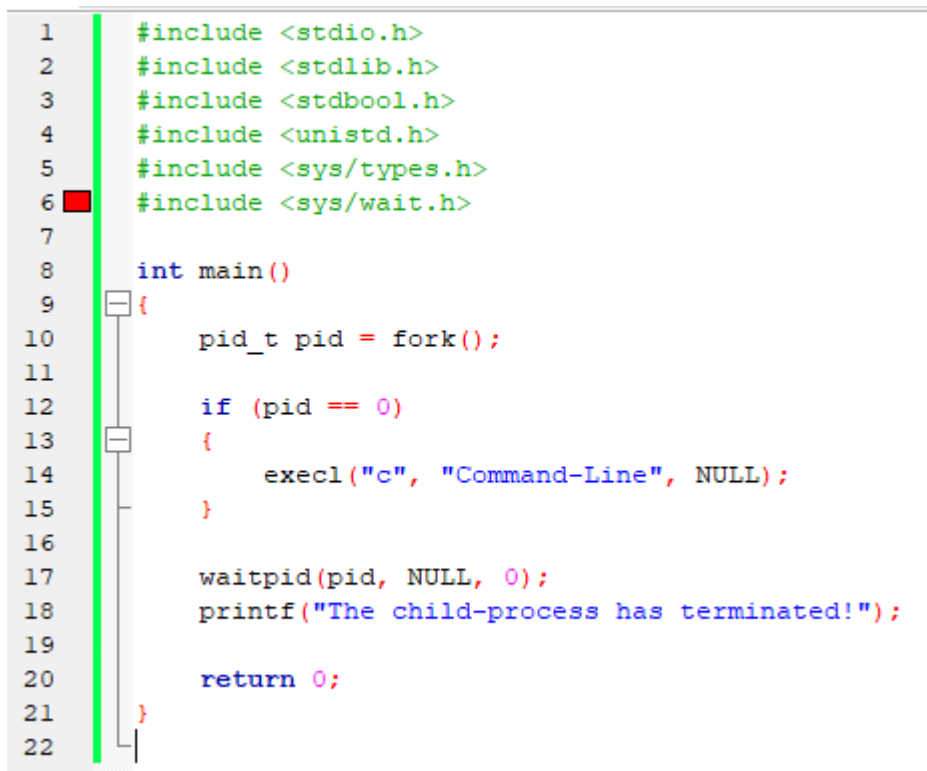
Miskolc, 2022

```
Process returned -1073741819 (0xC0000005)   execution time : 7.400 s
Press any key to continue.
```

3. feladat – Gyermek és szülő processz létrehozása



```
"D:\Egyetem\C programozás\child.c\bin\Debug\child.c.exe"
Holczer Vanda - CÖLLER
Holczer Vanda - CÖLLER
Holczer Vanda - CÖLLER
Holczer Vanda - CÖLLER
Holczer Vanda - CÖLLER
Holczer Vanda - CÖLLER
Holczer Vanda - CÖLLER
Holczer Vanda - CÖLLER
Holczer Vanda - CÖLLER
Holczer Vanda - CÖLLER
Process returned 0 (0x0) execution time : 0.028 s
Press any key to continue.
```



```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <stdbool.h>
4  #include <unistd.h>
5  #include <sys/types.h>
6  #include <sys/wait.h>
7
8  int main()
9  {
10     pid_t pid = fork();
11
12     if (pid == 0)
13     {
14         execl("c", "Command-Line", NULL);
15     }
16
17     waitpid(pid, NULL, 0);
18     printf("The child-process has terminated!");
19
20     return 0;
21 }
22
```

4. feladat – fork()-al gyerek processz létrehozása, exec családbeli rendszerhívás

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <stdbool.h>
4  #include <unistd.h>
5  #include <sys/types.h>
6  #include <sys/wait.h>
7
8  int main()
9  {
10     char *argv[3] = {"Command-line", ".", NULL};
11
12     int pid = fork();
13
14     if ( pid == 0 ) {
15         execlp( "find", argv );
16     }
17     wait(2);
18
19     printf("Child process terminated\n");
20
21     return 0;
22 }
23

```

5. feladat – fork()-al gyerek processz létrehozása, befejeződési állapotok

```

#include <stdio.h>
#include <stdlib.h>

#include <unistd.h>
pid_t fork(void);

#include <sys/types.h>
#include <sys/wait.h>
pid_t wait(int *wstatus);
pid_t waitpid(pid_t pid, int* wstatus, int options);

int main(void)
{
    pid_t pid;

    if((pid=fork())<0) perror("fork hiba");
    else if(pid==0)
        exit(7);
    if(wait(&status)!=pid) perror("wait hiba");
    if(WIFEXITED(status))
        printf("Normalis befejezodes, visszaadott ertekek= %d\n", WEXITSTATUS(status));

    if((pid=fork())<0) perror("fork hiba");
    else if(pid==0)
        abort();

    if((wait()
}

```