

# **Operációs Rendszerek BSc**

9.gyak.

2021.04.07

**Készítette:**

Orosz Dániel Bsc

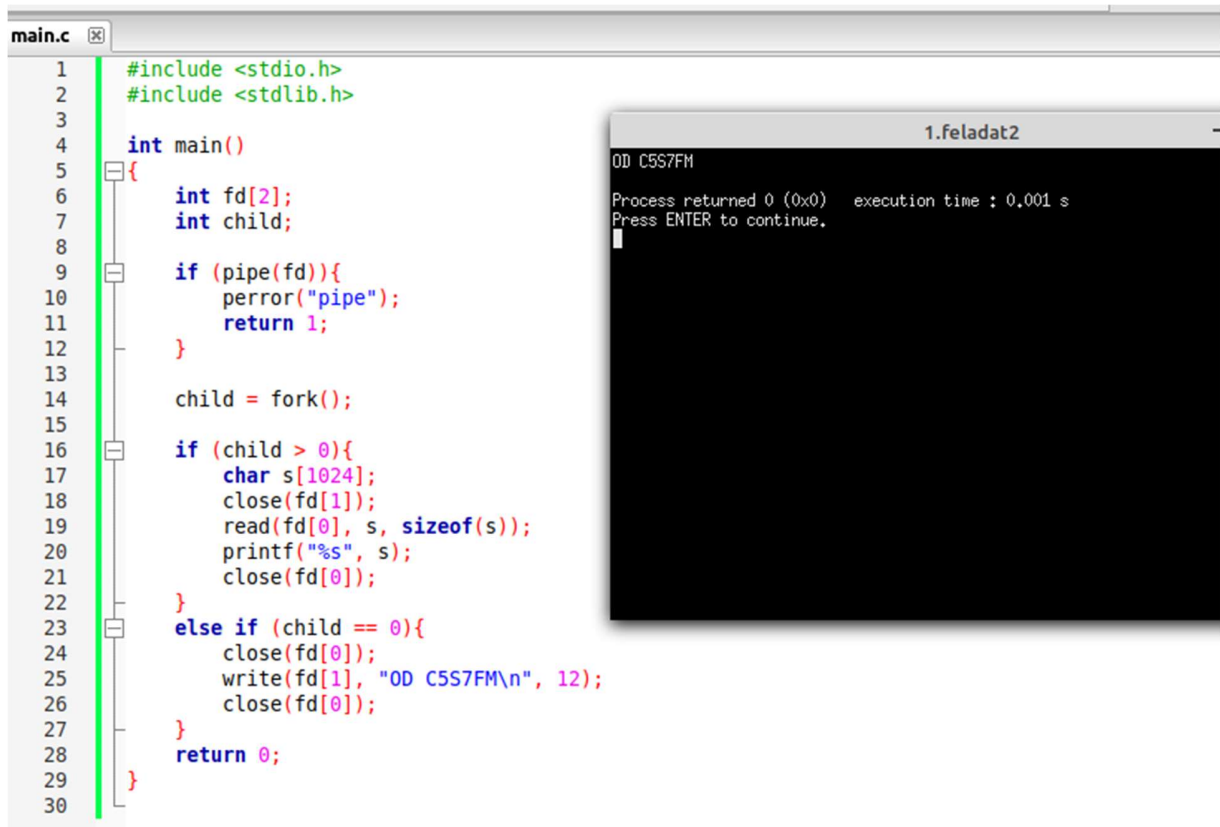
Üzemmérnök-informatikus

C5S7FM

**Miskolc, 2021**

1. Készítsen C nyelvű programot, ahol egy szülő processz létrehoz egy csővezetékét, a gyerek processz beleír egy szöveget a csővezetékbe (A kiírt szöveg: XY neptunkod), a szülő processz ezt kiolvassa, és kiírja a standard kimenetre.

Mentés: neptunkod\_unnamed.c



The image shows a C program in a text editor and its execution in a terminal window.

**main.c**

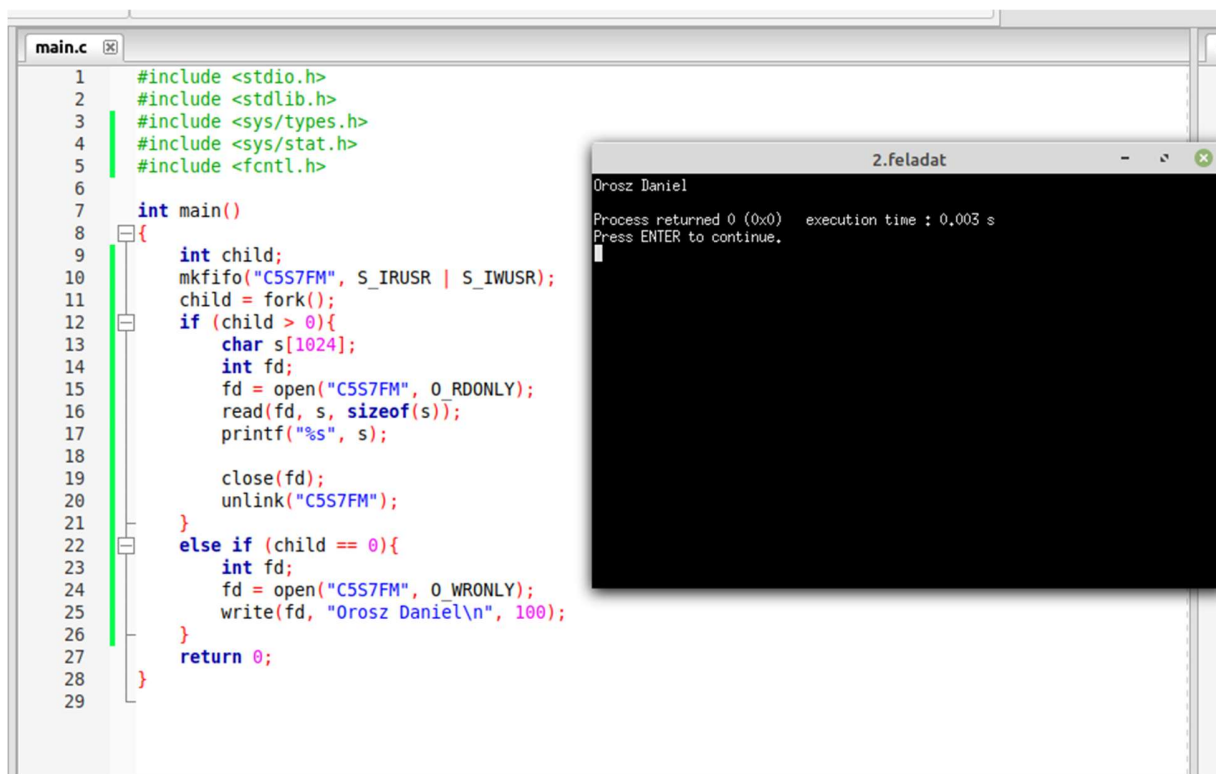
```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      int fd[2];
7      int child;
8
9      if (pipe(fd)){
10         perror("pipe");
11         return 1;
12     }
13
14     child = fork();
15
16     if (child > 0){
17         char s[1024];
18         close(fd[1]);
19         read(fd[0], s, sizeof(s));
20         printf("%s", s);
21         close(fd[0]);
22     }
23     else if (child == 0){
24         close(fd[0]);
25         write(fd[1], "0D C5S7FM\n", 12);
26         close(fd[1]);
27     }
28     return 0;
29 }
30
```

**1.feladat2**

```
0D C5S7FM
Process returned 0 (0x0)   execution time : 0.001 s
Press ENTER to continue.
```

2. Készítsen C nyelvű programot, ahol egy szülő processz létrehoz egy nevesített csővezetékét (neve: neptunkod), a gyerek processz beleír egy szöveget a csővezetékbe (A hallgató neve:pl. Keserű Ottó), a szülő processz ezt kiolvassa, és kiírja a standard kimenetre.

Mentés: neptunkod\_named.c



The image shows a C program in a text editor and its execution in a terminal window. The C program, named `main.c`, implements a named pipe (FIFO) for inter-process communication. It uses `mkfifo` to create the pipe `"C5S7FM"` with permissions `S_IRUSR | S_IWUSR`. The parent process (child ID `> 0`) opens the pipe in read-only mode (`O_RDONLY`), reads data from it, and prints it. The child process (child ID `== 0`) opens the pipe in write-only mode (`O_WRONLY`) and writes the string `"Orosz Daniel\n"` to it. The program returns `0` after successful execution.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <sys/types.h>
4  #include <sys/stat.h>
5  #include <fcntl.h>
6
7  int main()
8  {
9      int child;
10     mkfifo("C5S7FM", S_IRUSR | S_IWUSR);
11     child = fork();
12     if (child > 0){
13         char s[1024];
14         int fd;
15         fd = open("C5S7FM", O_RDONLY);
16         read(fd, s, sizeof(s));
17         printf("%s", s);
18
19         close(fd);
20         unlink("C5S7FM");
21     }
22     else if (child == 0){
23         int fd;
24         fd = open("C5S7FM", O_WRONLY);
25         write(fd, "Orosz Daniel\n", 100);
26     }
27     return 0;
28 }
29
```

The terminal window, titled `2.feladat`, shows the output of the program: `Orosz Daniel`. Below the output, it displays `Process returned 0 (0x0) execution time : 0.003 s` and `Press ENTER to continue.`