

Operációs rendszerek BSc

8.gyak.

2021. 03. 31.

Készítette:

Hegyesi Kristóf

Programtervező

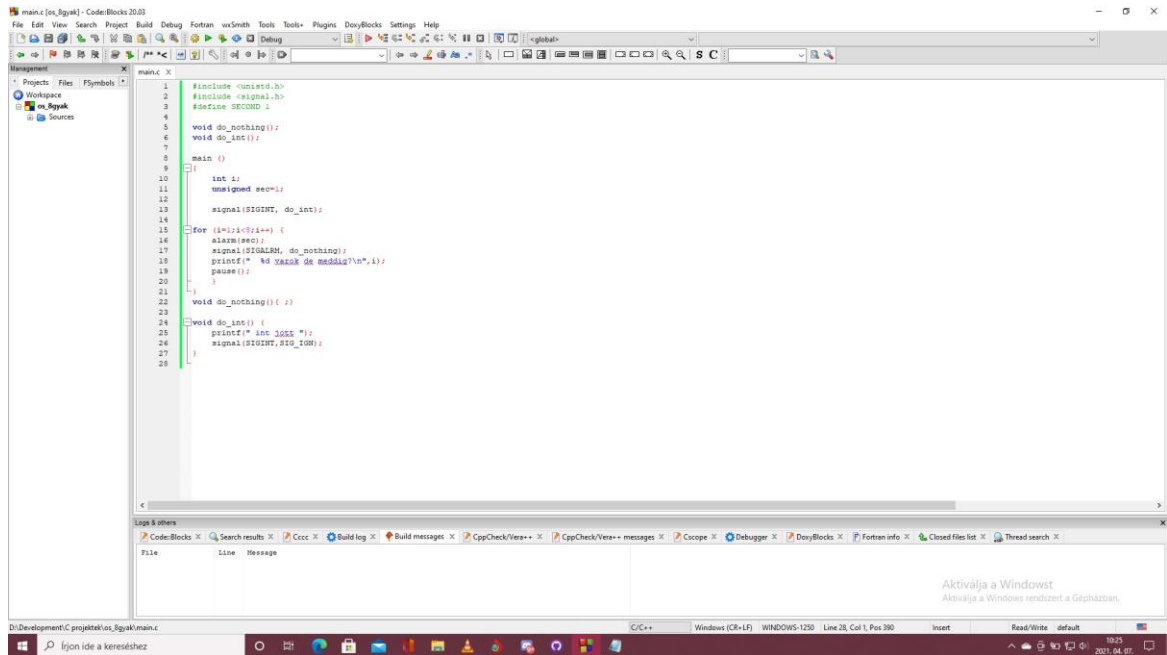
informatikus

Neptunkód: JPTJY2

Miskolc, 2021

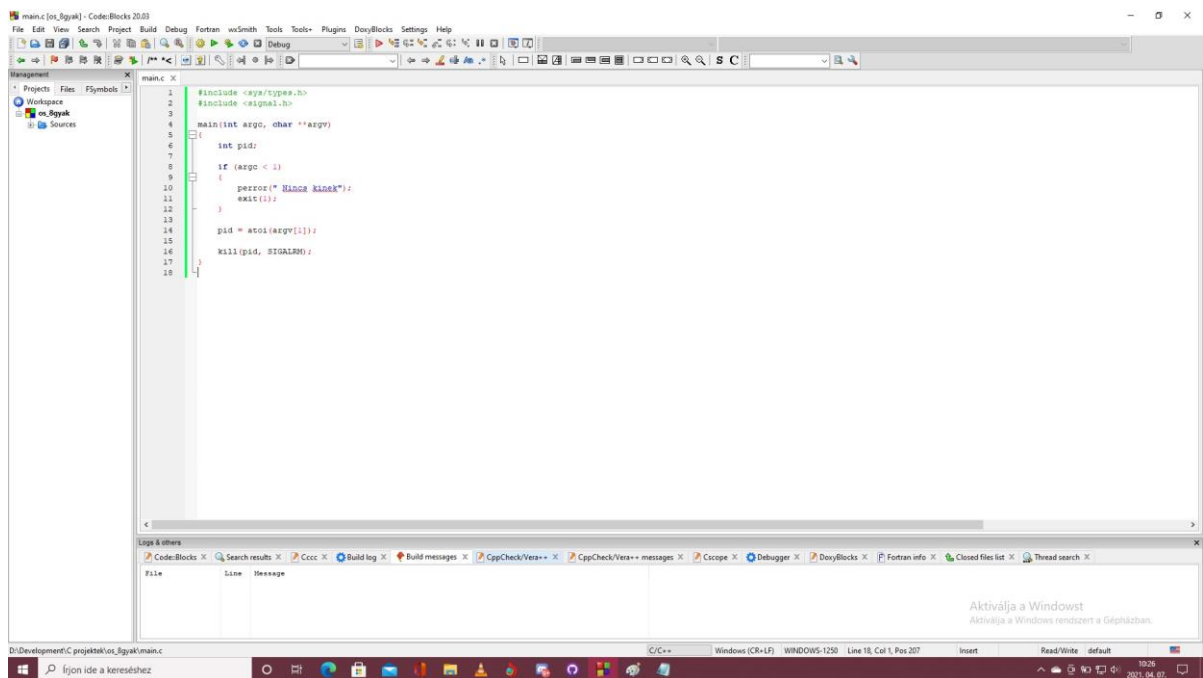
1.feladat

Alarm:



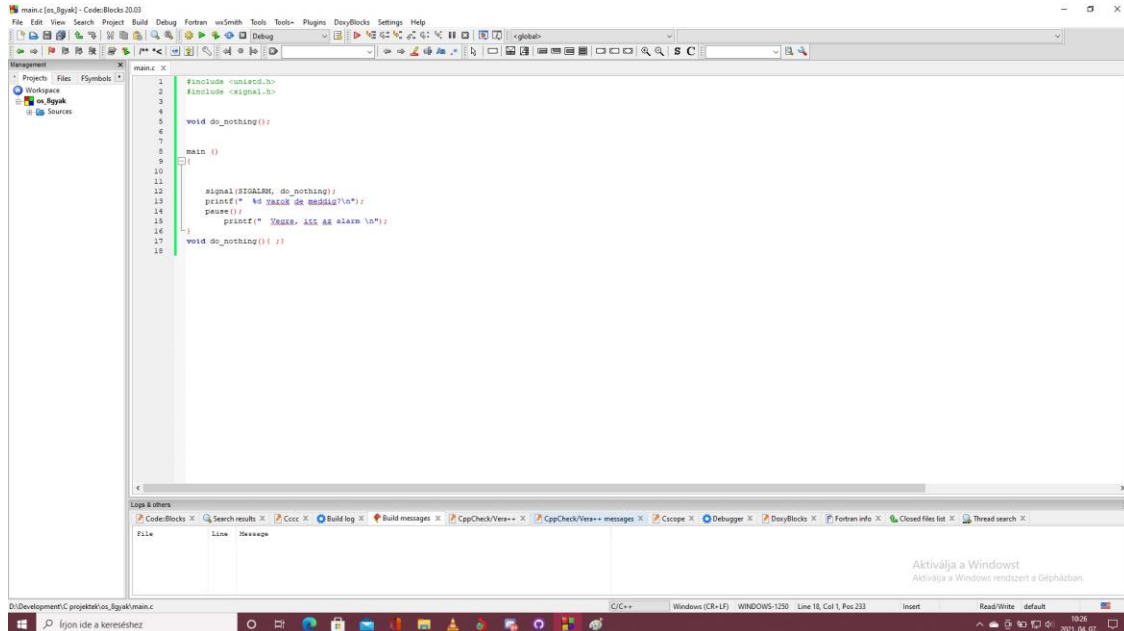
```
1 #include <unistd.h>
2 #include <signal.h>
3 #include <stdio.h>
4
5 void do_nothing();
6 void do_int();
7
8 main()
9 {
10     int i;
11     unsigned sec=1;
12     signal(SIGINT, do_int);
13
14     for (i=1;i<=10;i++) {
15         alarm(sec);
16         signal(SIGALRM, do_nothing);
17         printf("A YAKA de meddse?\\n",i);
18         pause();
19     }
20
21     void do_nothing() { }
22
23     void do_int() {
24         printf("Int SOS! ");
25         signal(SIGINT, SIG_IGN);
26     }
27 }
```

Alarm adó:



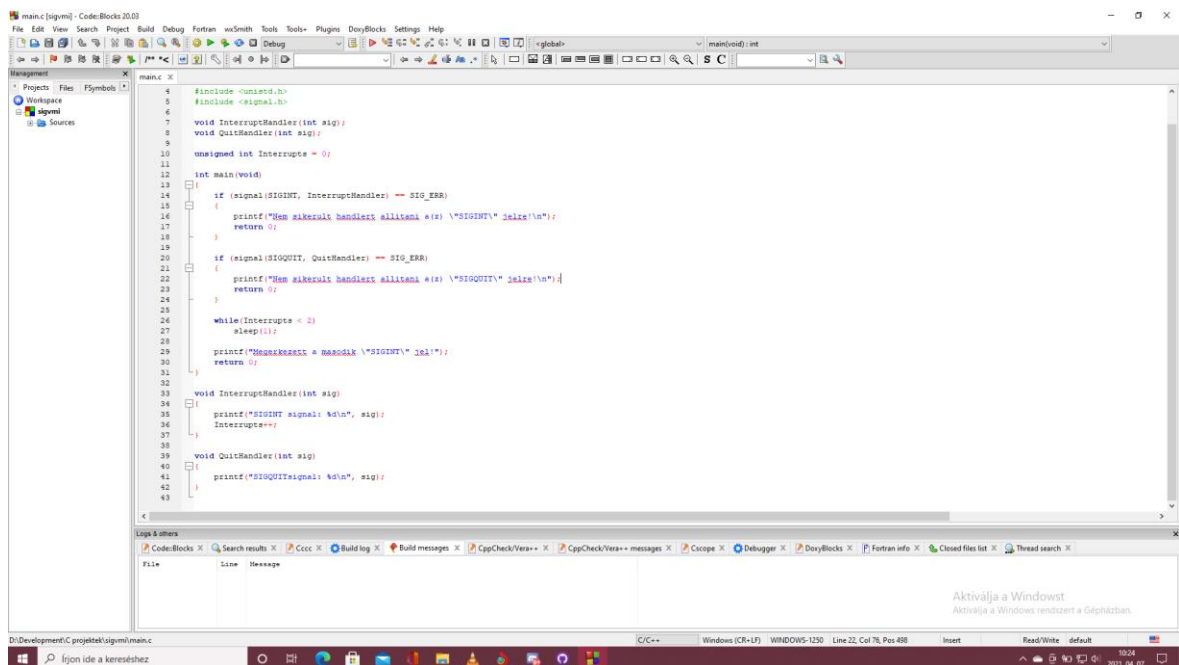
```
1 #include <sys/types.h>
2 #include <signal.h>
3 #include <unistd.h>
4
5 main(int argc, char **argv)
6 {
7     int pid;
8
9     if (argc < 2)
10     {
11         perror("Nincs kinek");
12         exit(1);
13     }
14
15     pid = atoi(argv[1]);
16     kill(pid, SIGALRM);
17 }
18 }
```

Alarmra vár:



```
1 #include <unistd.h>
2 #include <signal.h>
3
4 void do_nothing();
5
6
7 main ()
8 {
9     signal(SIGALRM, do_nothing);
10    printf(" VARSZAK: ARAZAS elkezdés\n");
11    pause();
12    printf(" VARSZAK: ARAZAS vége\n");
13}
14 void do_nothing() {}
```

2. feladat(Szingál kezelés)



```
4 #include <unistd.h>
5 #include <signal.h>
6
7 void InterruptHandler(int sig);
8 void QuitHandler(int sig);
9
10 unsigned int Interrupts = 0;
11
12 int main(void)
13 {
14     if (signal(SIGINT, InterruptHandler) == SIG_ERR)
15     {
16         printf("Nem sikerült handler állítani a(z) SIGINT-hez\n");
17         return 0;
18     }
19     if (signal(SIGQUIT, QuitHandler) == SIG_ERR)
20     {
21         printf("Nem sikerült handler állítani a(z) SIGQUIT-hez\n");
22         return 0;
23     }
24     while(Interrupts < 2)
25     {
26         sleep(1);
27         printf("Megkezdett a második SIGINT jel\n");
28         return 0;
29     }
30     void InterruptHandler(int sig)
31     {
32         printf("SIGINT signal: %d\n", sig);
33         Interrupts++;
34     }
35     void QuitHandler(int sig)
36     {
37         printf("SIGQUIT signal: %d\n", sig);
38     }
39 }
40
41
42
43
```