



# 1. feladat

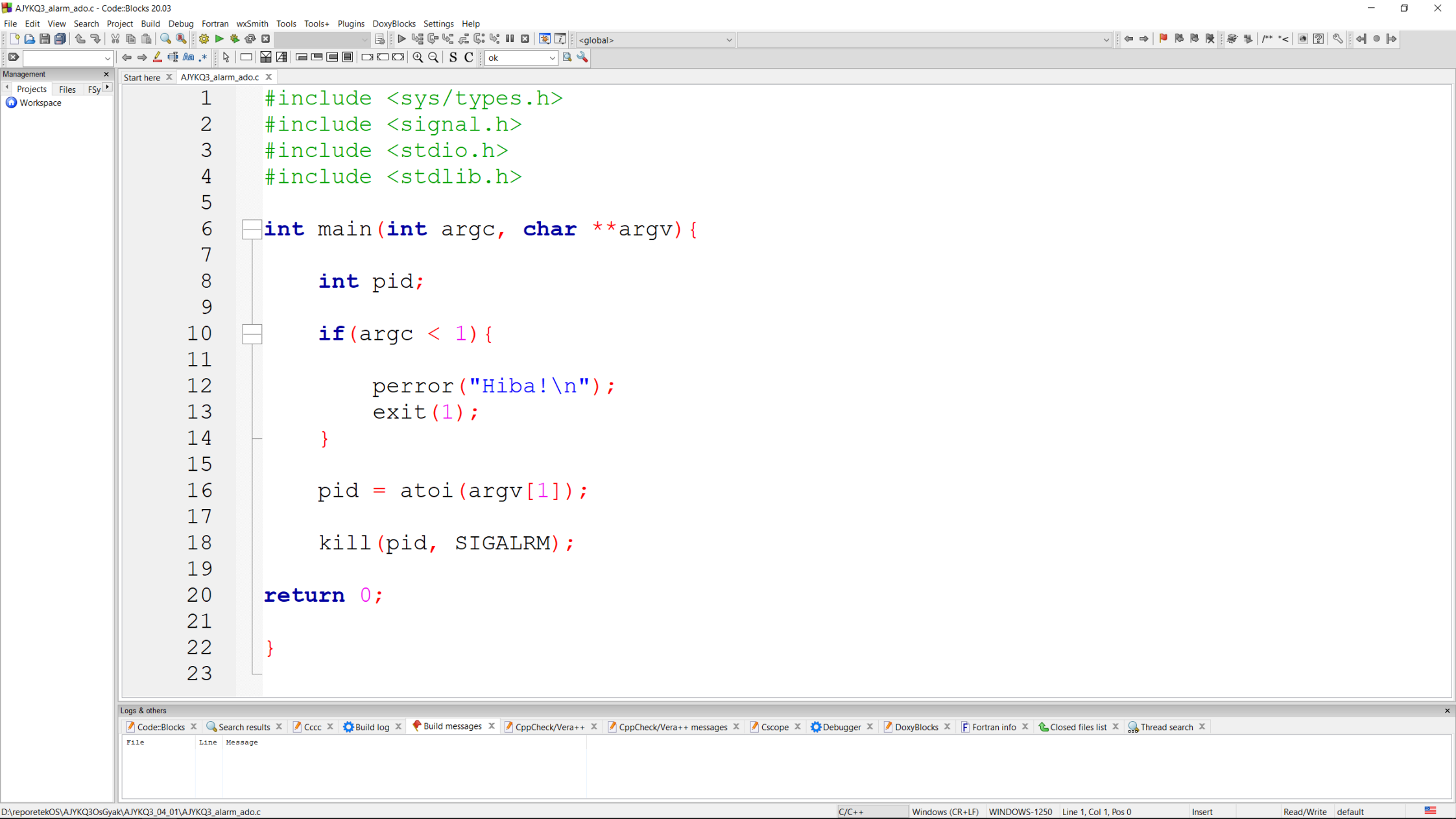
7-szer ismétlődő alarm + pause a beállított értékig

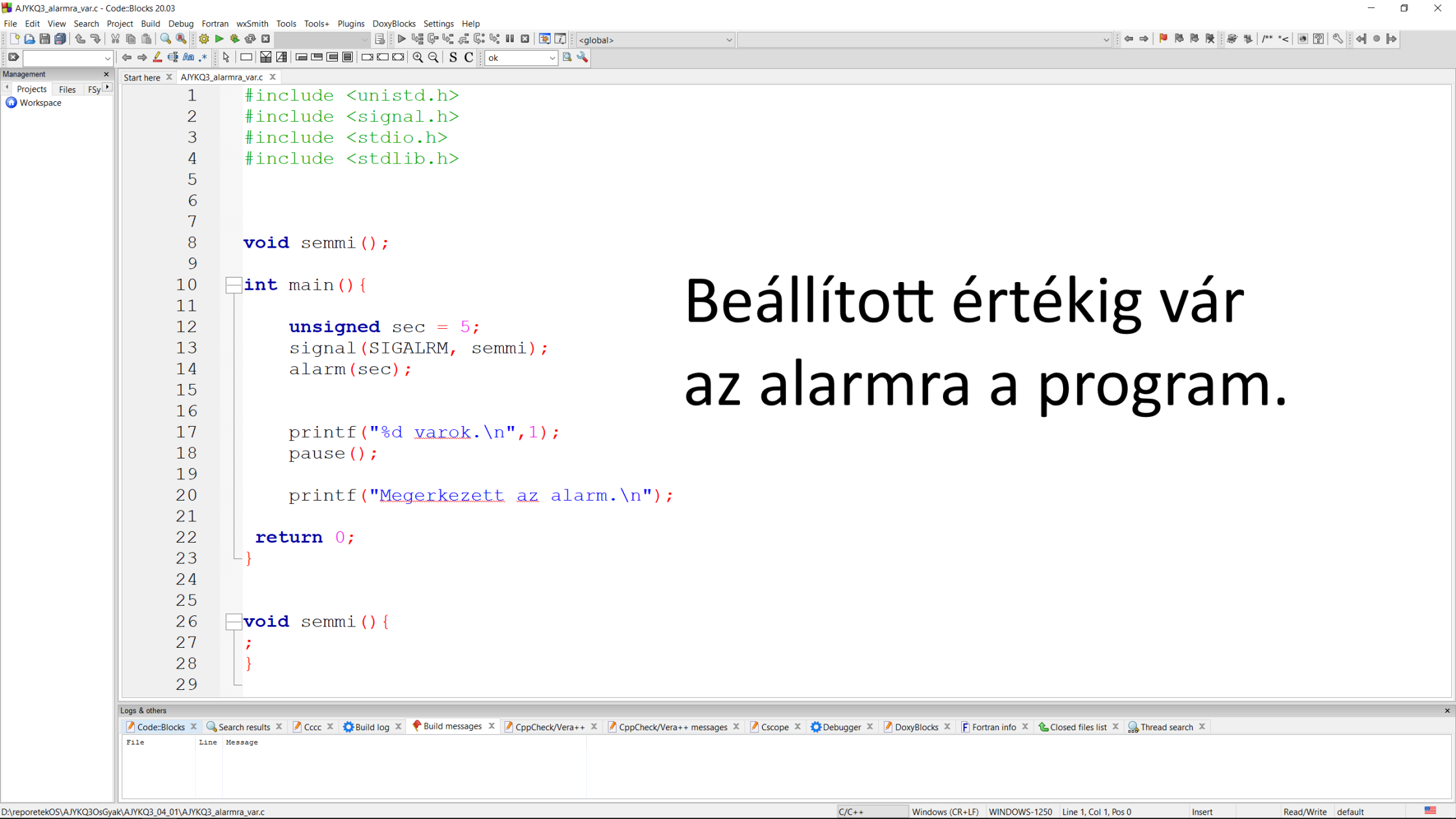
```
1  #include <unistd.h>
2  #include <signal.h>
3  #include <stdio.h>
4  #include <stdlib.h>
5
6  #define SECOND 1
7
8  void semmi();
9  void do_int();
10
11
12 int main(){
13
14     int i;
15     unsigned sec = 1;
16
17     signal(SIGINT, int_erkezik);
18
19     for(i = 1; i < 8; i++){
20         alarm(sec);
21
22
23         signal(SIGALRM, semmi);
24         printf("%d\n", i);
25         pause();
26     }
27     return 0;
28 }
29
30
31
32 void semmi(){
33 ;
34 }
35
36 void int_erkezik(){
37     printf("int megerkezett.\n");
38     signal(SIGINT, SIG_IGN);
39 }
40
```

Logs & others

Code::Blocks x Search results x Cccc x Build log x Build messages x CppCheck/Vera++ x CppCheck/Vera++ messages x Cscope x Debugger x DoxyBlocks x Fortran info x Closed files list x Thread search x

File	Line	Message





Beállított értékig vár  
az alarmra a program.

```
1  #include <unistd.h>
2  #include <signal.h>
3  #include <stdio.h>
4  #include <stdlib.h>
5
6
7
8  void semmi();
9
10 int main() {
11
12     unsigned sec = 5;
13     signal(SIGALRM, semmi);
14     alarm(sec);
15
16
17     printf("%d varok.\n", 1);
18     pause();
19
20     printf("Megerkezett az alarm.\n");
21
22     return 0;
23 }
24
25
26 void semmi() {
27 ;
28 }
29
```

Logs & others

Code::Blocks x Search results x Cccc x Build log x Build messages x CppCheck/Vera++ x CppCheck/Vera++ messages x Cscope x Debugger x DoxyBlocks x Fortran info x Closed files list x Thread search x

File	Line	Message

