

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  #include <unistd.h>
5  #include <signal.h>
6
7  void do_nothing();
8  void do_int();
9
10 int main(void)
11 {
12     unsigned sec = 1;
13     signal(SIGINT, do_int);
14
15     for (int i = 1; i < 8; i++)
16     {
17         alarm(sec);
18         signal(SIGALRM, do_nothing);
19         printf("%d Varok de meddig?\n", i);
20         pause();
21     }
22 }
23
24 void do_nothing() { }
25
26 void do_int()
27 {
28     printf("int jott");
29     signal(SIGINT, SIG_IGN);
30 }
31
```

Alarm

```
1 Varok de meddig?
2 Varok de meddig?
3 Varok de meddig?
4 Varok de meddig?
5 Varok de meddig?
6 Varok de meddig?
7 Varok de meddig?
```

Process returned 0 (0x0) execution time : 7.009 s
Press ENTER to continue.

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

Segmentation fault (core dumped)

Process returned 139 (0x8B) execution time : 0,310 s

Press ENTER to continue.

```
1  #include <unistd.h>
2  #include <signal.h>
3
4  void do_nothing();
5
6  int main(void)
7  {
8      signal(SIGALRM, do_nothing);
9      printf("%d Varok de meddig?\n");
10     pause();
11     printf("Vegre, itt az alarm \n");
12 }
13
14 void do_nothing() {}
15
```

-1342330960 Varok de meddig?


```

1 #include <stdio.h>
2 #include <stdlib.h>
3
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 sikerult handlert allitani a(z) \"SIGINT\" jelre!\n");
17         return 0;
18     }
19
20     if (signal(SIGQUIT, QuitHandler) == SIG_ERR)
21     {
22         printf("Nem sikerult handlert allitani a(z) \"SIGQUIT\" jelre!\n");
23         return 0;
24     }
25
26     while(Interrupts < 2)
27     {
28         printf("Varakozas jelre...\n");
29         sleep(1);
30     }
31
32     printf("Megerkezett a masodik \"SIGINT\" jel!");
33     return 0;
34 }
35
36 void InterruptHandler(int sig)
37 {
38     printf("SIGINT signal: %d\n", sig);
39     Interrupts++;
40 }
41
42 void QuitHandler(int sig)
43 {
44     printf("SIGQUIT signal: %d\n", sig);
45 }
46

```

[illegible]