


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

`}`

```
1  #include <sys/types.h>
2  #include <signal.h>
3  #include <stdio.h>
4  #include <stdlib.h>
5
6  int main(int argc, char **argv){
7
8      int pid;
9
10     if(argc < 1){
11
12         perror("Hiba!\n");
13         exit(1);
14     }
15
16     pid = atoi(argv[1]);
17
18     kill(pid, SIGALRM);
19
20     return 0;
21
22 }
```

```
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 }
```

```

8 void megszakit(int jel);
9 void kilep(int jel);
10
11 unsigned int megszakitások = 0;
12
13 int main(void)
14 {
15     if (signal(SIGINT, megszakit) == SIG_ERR)
16     {
17         printf("Nem sikerült handlert allitani a(s) SIGINT jelre!\n");
18         return 0;
19     }
20
21     if (signal(SIGQUIT, kilep) == SIG_ERR)
22     {
23         printf("Nem sikerült handlert allitani a(s) SIGQUIT jelre!\n");
24         return 0;
25     }
26
27     while(meszakitások < 2)
28     {
29         printf("Várakozás a jelre...\n");
30         sleep(1);
31     }
32
33     printf("Mégvan a maradék SIGINT jel!");
34     return 0;
35 }
36
37 void megszakit(int jel)
38 {
39     printf("SIGINT jel: %d\n", jel);
40     megszakitások++;
41 }
42
43 void kilep(int jel)
44 {
45     printf("SIGQUIT jel: %d\n", jel);
46 }

```