

Feladatok

1. Tanulmányozzák a Dr. Vadász Dénes: Operációs rendszerek, 2006. ME, jegyzet Szignálok fejezetet 61- 69 oldalig, majd...

Értelmezzék a mintapéldákat és oldják meg: alarm.c.; alarm_ado.c;

alarmra_var.c - szintén a jegyzet 68. oldalán található.

Mentés: neptunkod_alarm.c.; neptunkod_alarm_ado.c;

neptunkod_alarmra_var.c

```
abraham1@jerry:~$ ./a4xiov_alarm.out
1 Waiting.
2 Waiting.
3 Waiting.
4 Waiting.
5 Waiting.
6 Waiting.
7 Waiting.
abraham1@jerry:~$
```

```
GNU nano 2.7.4 File: a4xiov_alarm.c

#include <stdio.h>
#include <unistd.h>
#include <signal.h>
#define SECOND 1

void do_nothing();
void do_int();

int main(){
    int i;
    unsigned sec = 1;
    signal(SIGINT, do_int);

    for(i=1; i<8; i++){
        alarm(sec);
        signal(SIGALRM, do_nothing);
        printf("%d Waiting.\n", i);
        pause();
    }
}

void do_nothing(){ ;}

void do_int(){
    printf("Int came");
    signal(SIGINT, SIG_IGN);
}
```

0



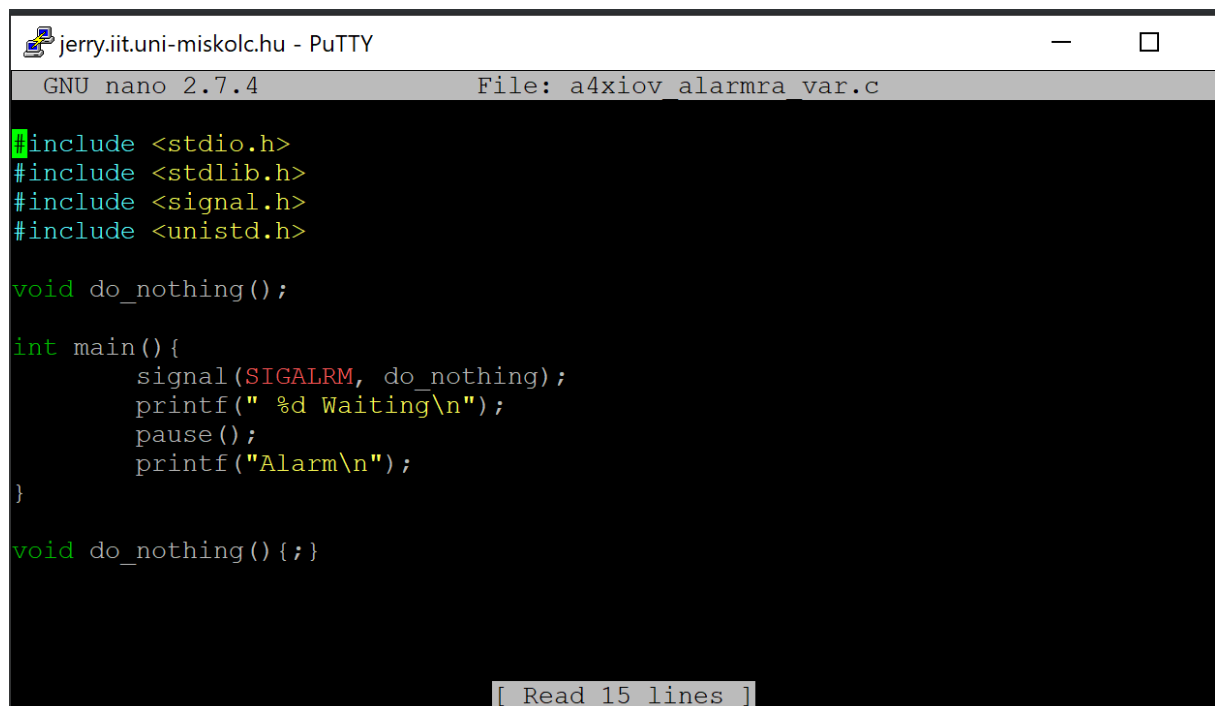
```
GNU nano 2.7.4 File: a4xiov_alarm_ado.c

#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <signal.h>

int main(int argc, char **argv){
    int pid;

    if(argc < 1){
        perror("Nincs kinek");
        exit(1);
    }

    pid = atoi(argv[1]);
    kill(pid, SIGALRM);
}
```



```
GNU nano 2.7.4 File: a4xiov_alarmra_var.c

#include <stdio.h>
#include <stdlib.h>
#include <signal.h>
#include <unistd.h>

void do_nothing();

int main(){
    signal(SIGALRM, do_nothing);
    printf(" %d Waiting\n");
    pause();
    printf("Alarm\n");
}

void do_nothing(){;}
```

[Read 15 lines]