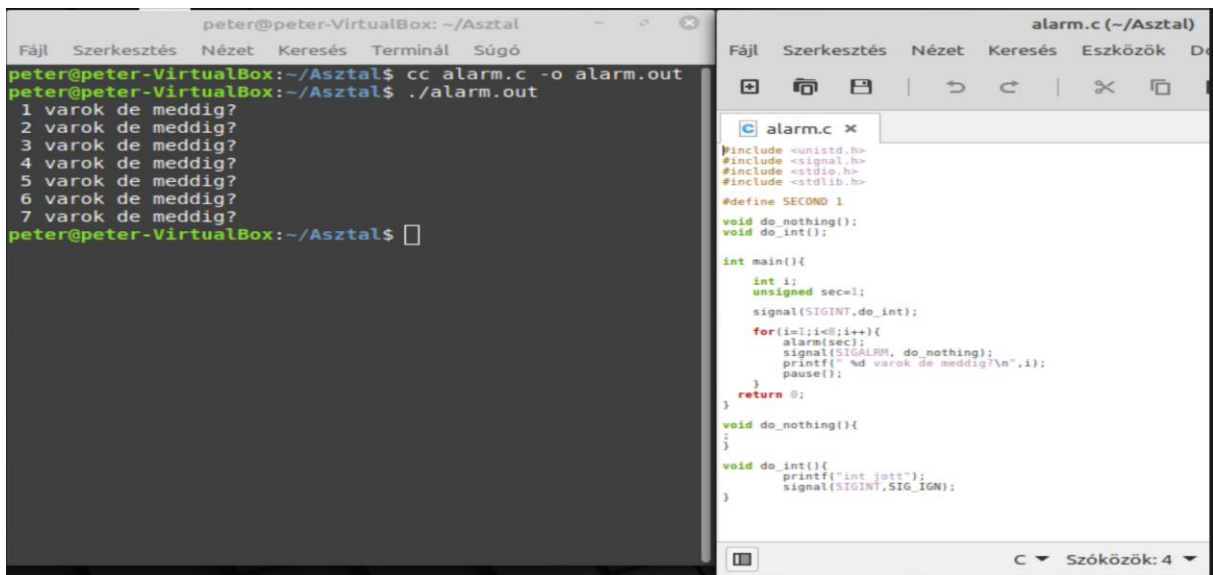


## alarm.c



The screenshot shows a terminal window on the left and a code editor on the right. The terminal window displays the compilation and execution of `alarm.c`. The code editor shows the source code of `alarm.c`.

```
peter@peter-VirtualBox: ~/Asztal
Fájl Szerkesztés Nézet Keresés Terminál Súgó
peter@peter-VirtualBox:~/Asztal$ cc alarm.c -o alarm.out
peter@peter-VirtualBox:~/Asztal$ ./alarm.out
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?
peter@peter-VirtualBox:~/Asztal$
```

```
alarm.c (~/.Asztal)
Fájl Szerkesztés Nézet Keresés Eszközök
alarm.c *
#include <unistd.h>
#include <signal.h>
#include <stdio.h>
#include <stdlib.h>

#define SECOND 1

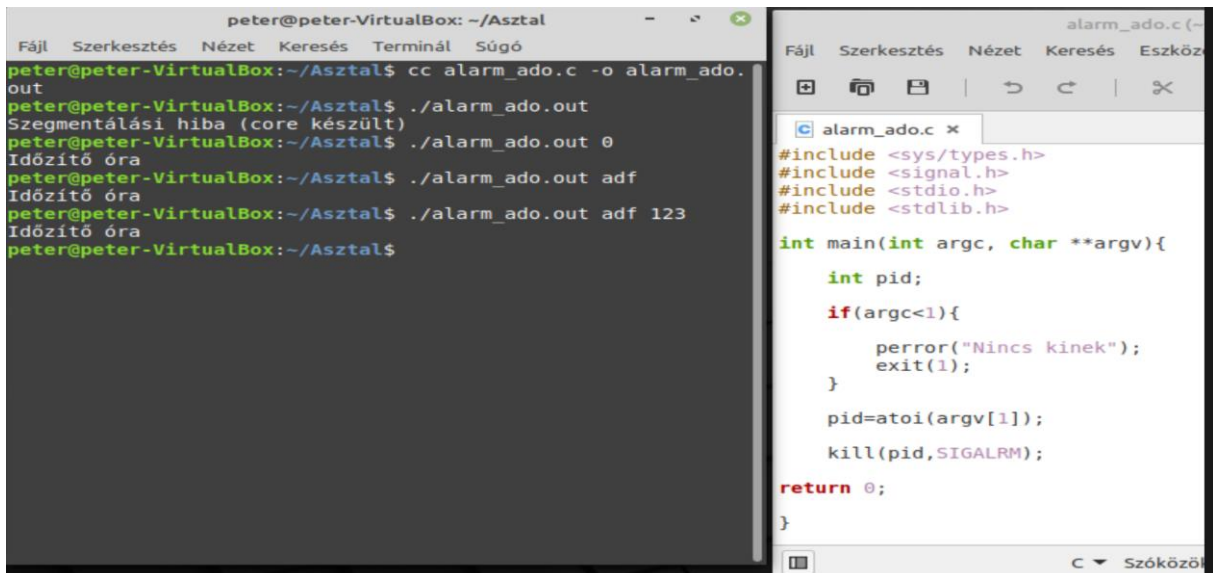
void do_nothing();
void do_int();

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 varok de meddig?\n",i);
        pause();
    }
    return 0;
}

void do_nothing(){
}

void do_int(){
    printf("int jott");
    signal(SIGINT,SIG_IGN);
}
```

## alarm\_ado.c



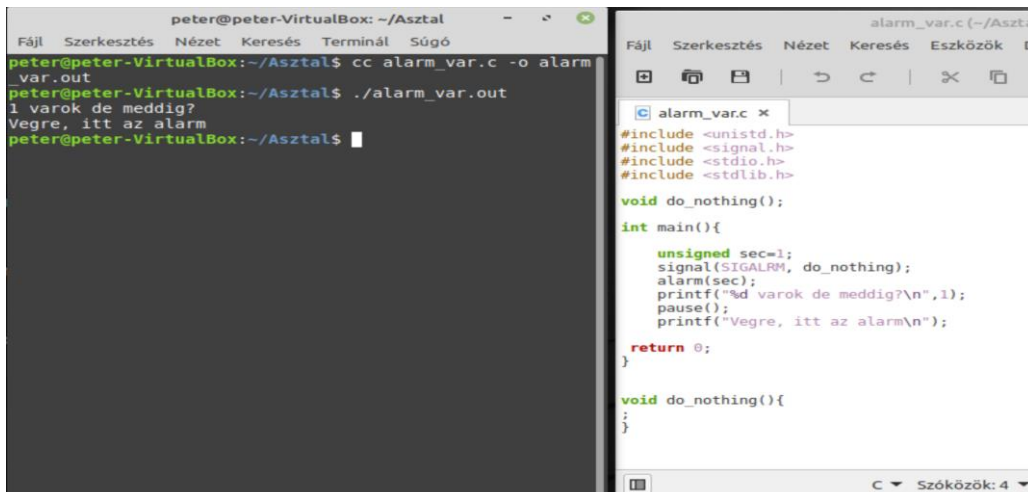
The screenshot shows a terminal window on the left and a code editor on the right. The terminal window displays the compilation and execution of `alarm_ado.c`. The code editor shows the source code of `alarm_ado.c`.

```
peter@peter-VirtualBox: ~/Asztal
Fájl Szerkesztés Nézet Keresés Terminál Súgó
peter@peter-VirtualBox:~/Asztal$ cc alarm_ado.c -o alarm_ado.out
peter@peter-VirtualBox:~/Asztal$ ./alarm_ado.out
Szegmentálási hiba (core készült)
peter@peter-VirtualBox:~/Asztal$ ./alarm_ado.out 0
Időzítő óra
peter@peter-VirtualBox:~/Asztal$ ./alarm_ado.out adf
Időzítő óra
peter@peter-VirtualBox:~/Asztal$ ./alarm_ado.out adf 123
Időzítő óra
peter@peter-VirtualBox:~/Asztal$
```

```
alarm_ado.c (~/.Asztal)
Fájl Szerkesztés Nézet Keresés Eszközök
alarm_ado.c *
#include <sys/types.h>
#include <signal.h>
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char **argv){
    int pid;
    if(argc<1){
        perror("Nincs kinek");
        exit(1);
    }
    pid=atoi(argv[1]);
    kill(pid,SIGALRM);
    return 0;
}
```

## alarm\_var.c



The screenshot shows a terminal window on the left and a code editor on the right. The terminal window displays the compilation and execution of `alarm_var.c`. The code editor shows the source code of `alarm_var.c`.

```
peter@peter-VirtualBox: ~/Asztal
Fájl Szerkesztés Nézet Keresés Terminál Súgó
peter@peter-VirtualBox:~/Asztal$ cc alarm_var.c -o alarm_var.out
peter@peter-VirtualBox:~/Asztal$ ./alarm_var.out
1 varok de meddig?
Vegre, itt az alarm
peter@peter-VirtualBox:~/Asztal$
```

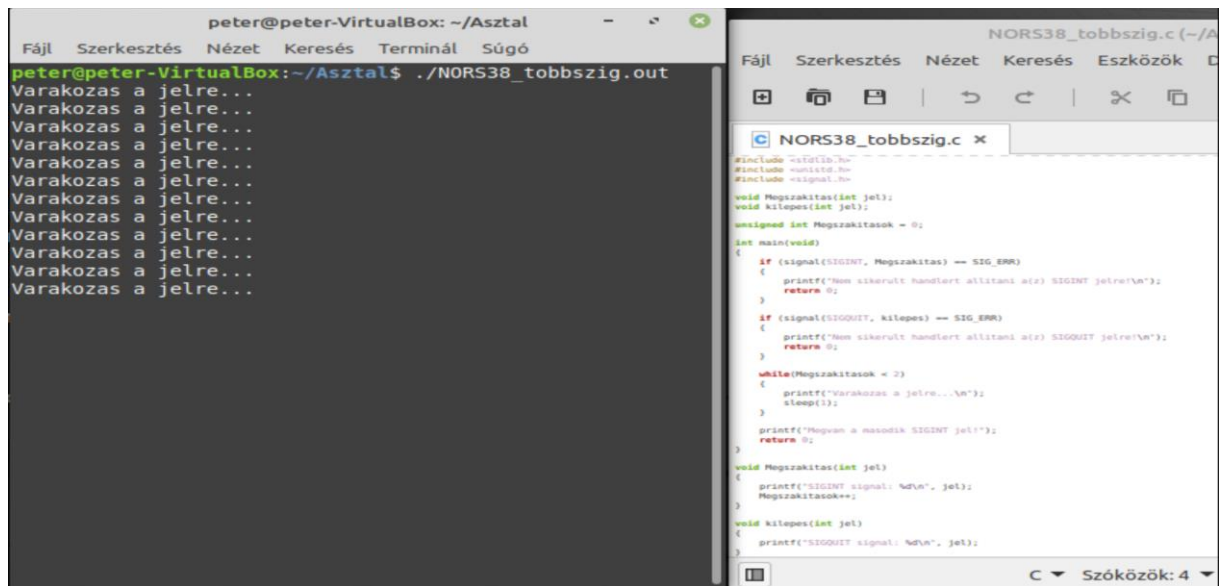
```
alarm_var.c (~/.Asztal)
Fájl Szerkesztés Nézet Keresés Eszközök
alarm_var.c *
#include <unistd.h>
#include <signal.h>
#include <stdio.h>
#include <stdlib.h>

void do_nothing();

int main(){
    unsigned sec=1;
    signal(SIGALRM, do_nothing);
    alarm(sec);
    printf("%d varok de meddig?\n",1);
    pause();
    printf("Vegre, itt az alarm\n");
    return 0;
}

void do_nothing(){
}
```

## NORS38\_tobbszig.c



The image shows a side-by-side view of a terminal window and a code editor. The terminal window on the left shows the execution of the program, which prints "Varakozas a jelre..." ten times. The code editor on the right shows the source code of the program, which is a C program that handles SIGINT and SIGQUIT signals. The code includes headers for `stdio.h`, `unistd.h`, and `signal.h`. It defines a `Megszakitas` enum with values `SIGINT` and `SIGQUIT`. The `main` function sets up signal handlers for these signals and prints the signal name when received. It also prints "Varakozas a jelre..." and sleeps for 2 seconds before printing "Megvan a masodik SIGINT jel!".

```
peter@peter-VirtualBox: ~/Asztal
Fájl Szerkesztés Nézet Keresés Terminál Súgó
peter@peter-VirtualBox:~/Asztal$ ./NORS38_tobbszig.out
Varakozas a jelre...
Varakozas a jelre...
Varakozas a jelre...
Varakozas a jelre...
Varakozas a jelre...
Varakozas a jelre...
Varakozas a jelre...
Varakozas a jelre...
Varakozas a jelre...
Varakozas a jelre...

NORS38_tobbszig.c (~/.A
Fájl Szerkesztés Nézet Keresés Eszközök
NORS38_tobbszig.c x
#include <stdio.h>
#include <unistd.h>
#include <signal.h>

void Megszakitas(int jel);
void kilepes(int jel);

unsigned int Megszakitasok = 0;

int main(void)
{
    if (signal(SIGINT, Megszakitas) == SIG_ERR)
    {
        printf("Nem sikerult handlert allitani a(x) SIGINT jelre!\n");
        return 0;
    }
    if (signal(SIGQUIT, kilepes) == SIG_ERR)
    {
        printf("Nem sikerult handlert allitani a(x) SIGQUIT jelre!\n");
        return 0;
    }
    while(Megszakitasok < 2)
    {
        printf("Varakozas a jelre...\n");
        sleep(2);
    }
    printf("Megvan a masodik SIGINT jel!");
    return 0;
}

void Megszakitas(int jel)
{
    printf("SIGINT signal: %d\n", jel);
    Megszakitasok++;
}

void kilepes(int jel)
{
    printf("SIGQUIT signal: %d\n", jel);
}
```