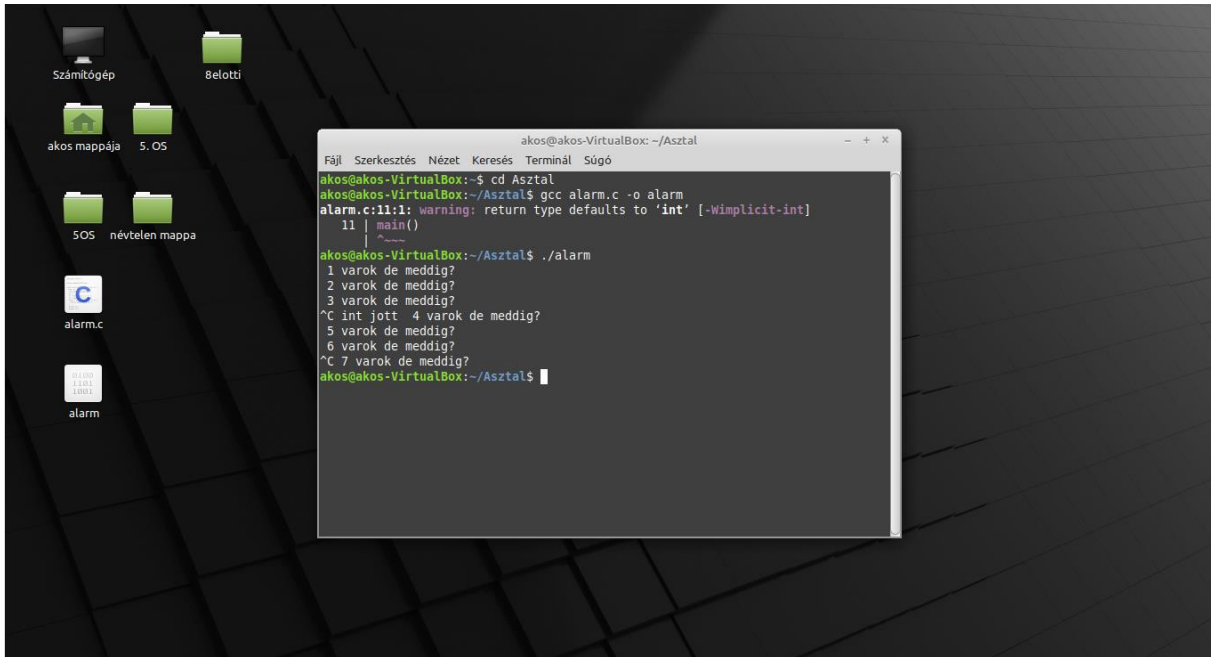


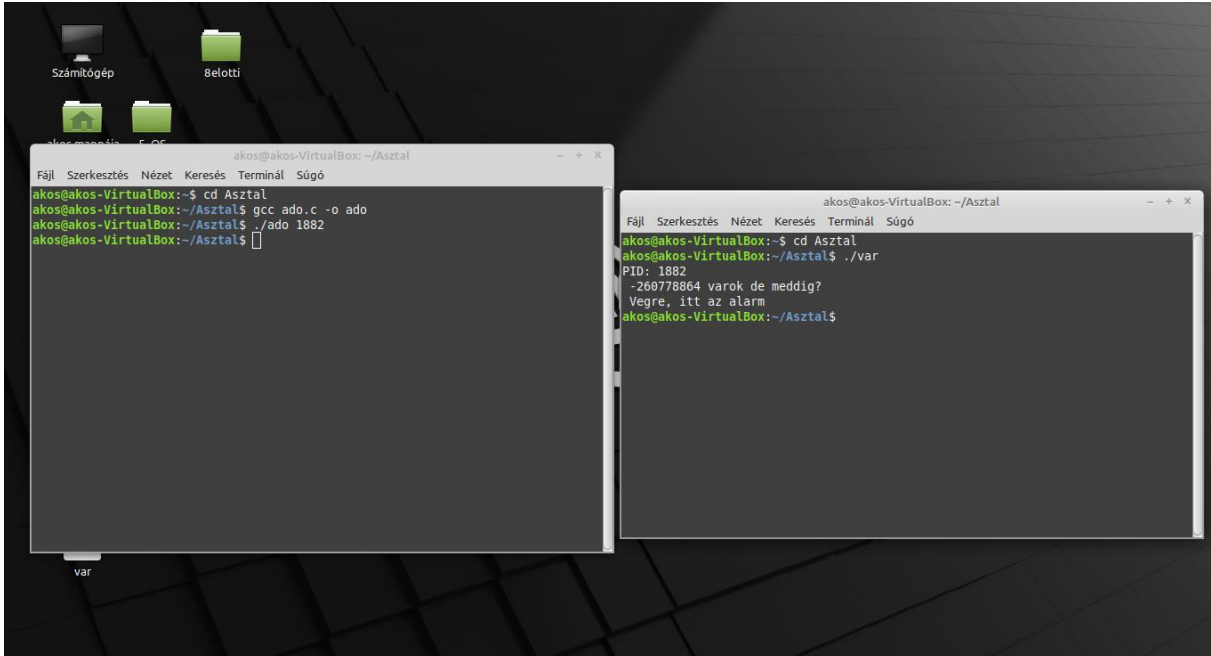
8. gyakorlat:

1.)



```
akos@akos-VirtualBox: ~/Asztal
Fájl Szerkesztés Nézet Keresés Terminál Súgó
akos@akos-VirtualBox:~$ cd Asztal
akos@akos-VirtualBox:~/Asztal$ gcc alarm.c -o alarm
alarm.c:11:1: warning: return type defaults to 'int' [-Wimplicit-int]
11 | main()
    | ^~~~~
akos@akos-VirtualBox:~/Asztal$ ./alarm
1 varok de meddig?
2 varok de meddig?
3 varok de meddig?
^C int jött 4 varok de meddig?
5 varok de meddig?
6 varok de meddig?
^C 7 varok de meddig?
akos@akos-VirtualBox:~/Asztal$
```

Ahogy a program leírása is mutatja: a while ciklus folyamatosan ismétlődne. A CTRL + C lenyomására alapvetően vége lenne a futásának, de az első kezelésnél ezt felülkezeljük. A második alkalommal viszont megszakítja így a while nem jut el a 8. ciklusába.



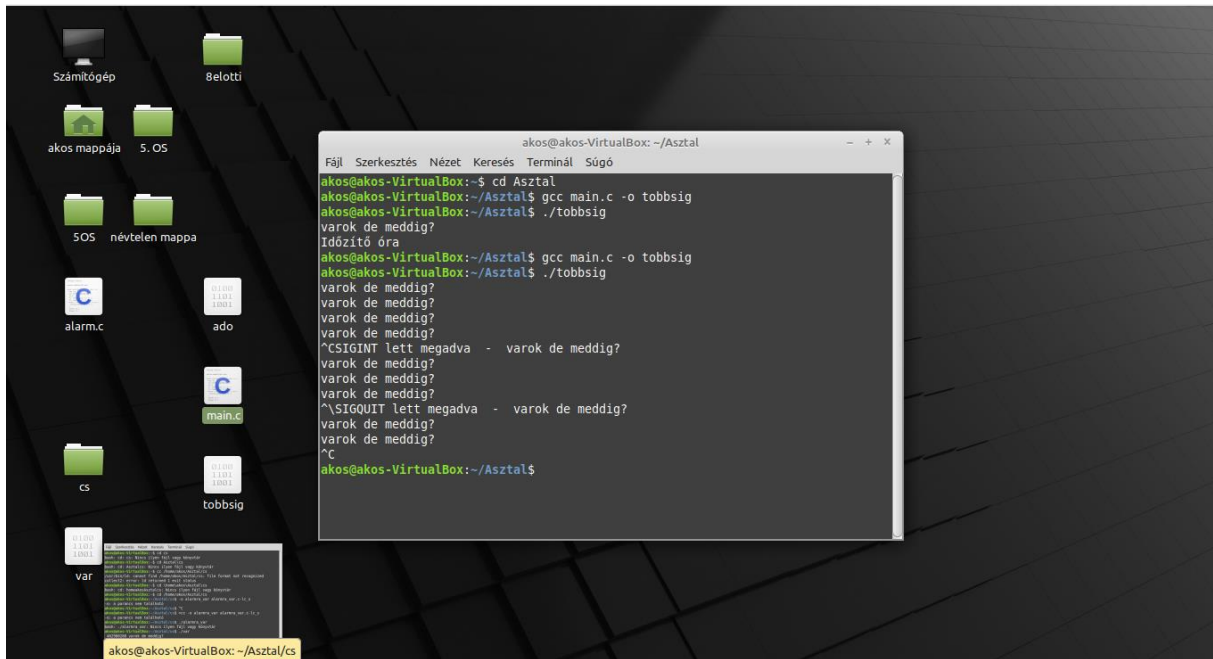
```
akos@akos-VirtualBox: ~/Asztal
Fájl Szerkesztés Nézet Keresés Terminál Súgó
akos@akos-VirtualBox:~$ cd Asztal
akos@akos-VirtualBox:~/Asztal$ gcc ado.c -o ado
akos@akos-VirtualBox:~/Asztal$ ./ado 1882
akos@akos-VirtualBox:~/Asztal$

akos@akos-VirtualBox: ~/Asztal
Fájl Szerkesztés Nézet Keresés Terminál Súgó
akos@akos-VirtualBox:~$ cd Asztal
akos@akos-VirtualBox:~/Asztal$ ./var
PID: 1882
-260778864 varok de meddig?
Vegre, itt az alarm
akos@akos-VirtualBox:~/Asztal$
```

Az alarmra_var.c annyiban lett módosítva az eredetitől, hogy kiírja a pidét, így az alarm_ado.c -nek megadva argumentumként képes alarmot küldeni a váró programnak, ami így megáll.

2.)

(2. feladatot választottam)



A while ciklus folyamatosan ismétlődik. A handleSignalsban egy egyszerű if függvénnyel megvizsgáljuk a felhasználó inputját, ami vagy a SIGINT vagy a SIGQUIT handlerjéhez vezet.