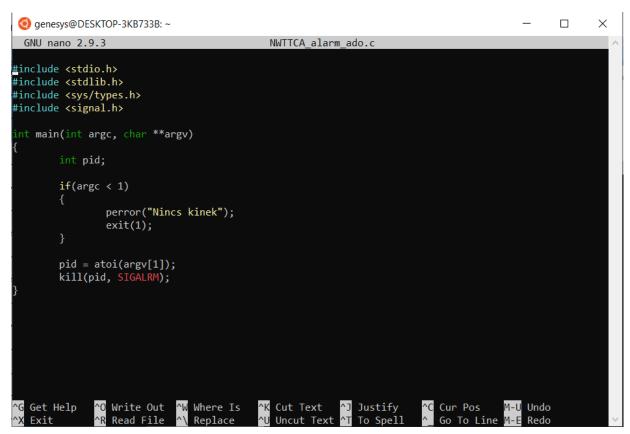```
  GNU nano 2.9.3   NWTTCA_alarm.c

#include <stdio.h>
#include <stdlib.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 varok de meddig?\n", i);
        pause();
        }

}
                              [Read 31 lines]
^G Get Help   ^O Write Out   ^W Where Is   ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo
^X Exit   ^R Read File   ^\ Replace   ^U Uncut Text ^T To Spell   ^ Go To Line M-E Redo
```

```
  GNU nano 2.9.3                    NWTTCA_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);
}
^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^  Go To Line  M-E Redo
```

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

void do_nothing();

int main(void)
{
        signal(SIGALRM, do_nothing);
        printf("varok de meddig?\n");
        pause();
        printf("Vegre, itt az alarm \n");
}

void do_nothing() {}
```

2.feladat

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

void InterruptHandler(int sig);
void QuitHandler(int sig);

unsigned int Interrupts = 0;

int main(void){
        if(signal(SIGINT, InterruptHandler) == SIG_ERR){
                printf("Nem sikerult handlert allitani a \"SIGINT\" jelre! \n");
                return 0;
        }

        if(signal(SIGQUIT, QuitHandler) == SIG_ERR){
                printf("Nem sikerult handlert allitani a \"SIGQUIT\" jelre! \n");
                return 0;
        }

        while(Interrupts < 2){
                printf("Varakozas jelre \n");
                sleep(1);
        }

        printf("Megerkezett a masodik \"SIGINT\" jel! \n");
        return 0;
}

void InterruptHandler(int sig){
        printf("SIGINT signal: %d \n", sig);
        Interrupts++;
}

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

```
  System information as of Tue Apr 13 12:31:28 CEST 2021

  System load:        0.52      Memory usage: 64%    Processes:         8
  Usage of /home: unknown       Swap usage:    0%    Users logged in: 0

  => There were exceptions while processing one or more plugins. See
     /var/log/landscape/sysinfo.log for more information.


261 packages can be updated.
160 updates are security updates.

New release '20.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


genesys@DESKTOP-3KB733B:~$ nano NWTTCA_tobbszignal_kez.c
genesys@DESKTOP-3KB733B:~$ gcc NWTTCA_tobbszignal_kez.c -o test
genesys@DESKTOP-3KB733B:~$ ./test
Varakozas jelre
Varakozas jelre
Varakozas jelre
Varakozas jelre
Varakozas jelre
Varakozas jelre
^CSIGINT signal: 2
Varakozas jelre
^CSIGINT signal: 2
Megerkezett a masodik "SIGINT" jel!
genesys@DESKTOP-3KB733B:~$
```