



marty@MartyPC: ~/Asztal Fájl Szerkesztés Nézet Keresés Terminál Súgó marty@MartyPC:~/Asztal\$ cc AJYKQ3 named.c -o main.out marty@MartyPC:~/Asztal\$./main.out RENDSZER: A gyerek processz elindult! RENDSZER: A gyerek processz elindult! GYEREK: FIFO megnyitva csak irasra, iras folyamatban... RENDSZER: A gyerek processz sikeresen befejezete az irast, a gyerek processz kilep. SZULO: FIFO megnyitva csak olvasasra, olvasas folyamatban... SZULO: Kiolvasott adat: Garamszegi Marton. RENDSZER: A szulo processz sikeresen befejezete az olvasast. RENDSZER: Szulo processz inditasa RENDSZER: FIFO sikeresen torolve, kilepes. marty@MartyPC:~/Asztal\$

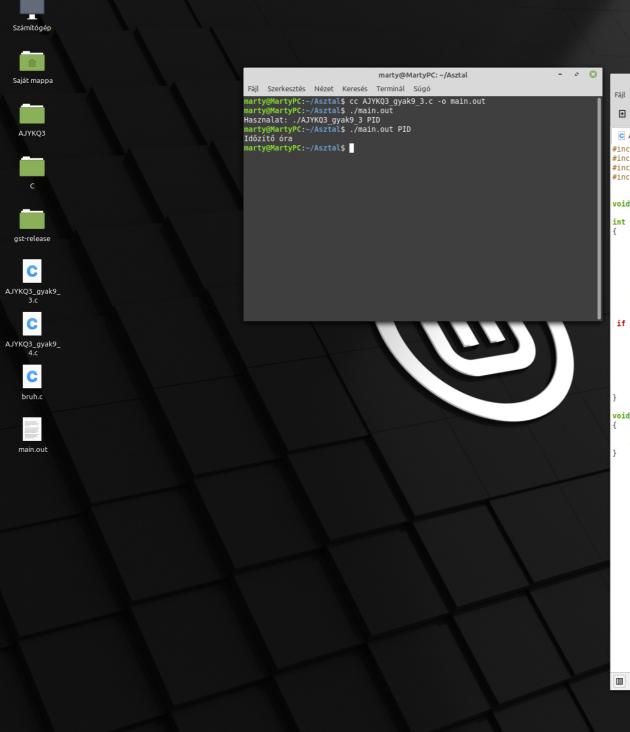
```
AJYKQ3_named.c (~/Asztal)
Fájl Szerkesztés Nézet Keresés Eszközök Dokumentumok Súgó
 団 市 □ | 5 c | % □ □ | Q 欠
 C AJYKQ3_unnamed.c × C AJYKQ3_named.c ×
#include <stdio.h>
#include <unistd.h>
#include <svs/tvpes.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <stdlib.h>
#include <string.h>
int main(){
    int fd:
   pid t gyerek;
    if((mkfifo("AJYQK3", S IRUSR | S IWUSR)) < 0)</pre>
        printf("RENDSZER: Nem sikerult a FIFO leterhozasa!\n");
        exit(1);
    if((gyerek = fork()) < 0){</pre>
        printf("RENDSZER: Nem sikerult a gyerek fork processz letrehozasa!\n");
    else
        printf("RENDSZER: A gyerek processz elindult!\n\n");
    //Gyerek processz
    if(!gyerek){
        char *uzenet = "Garamszegi Marton";
        if((fd = open("AJYQK3", 0 WRONLY)) < 0){</pre>
            printf("GYEREK: Nem sikerult a FIFO megnyitasa!\n");
            exit(1);
        } else {
            printf("GYEREK: FIFO megnyitva csak irasra, iras folyamatban...\n");
            write(fd, uzenet, strlen(uzenet));
        printf("RENDSZER: A gyerek processz sikeresen befejezete az irast,\na gyerek
processz kilep.\n\n");
        printf("RENDSZER: Szulo processz inditasa\n");
    //Szulo processz
   if(gyerek){
        char buf[256];
        if((fd = open("AJYQK3", 0 RDONLY)) <0 ){</pre>
            printf("SZULO: Nem sikerult a FIFO megnyitasa!\n");
            exit(1);
                                                    C ▼ Szóközök: 4 ▼
                                                                        1. sor. 1. oszlop
                                                                                         BES7
```

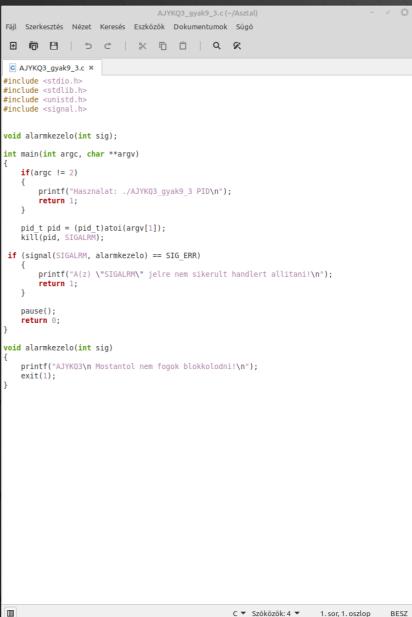












15:49

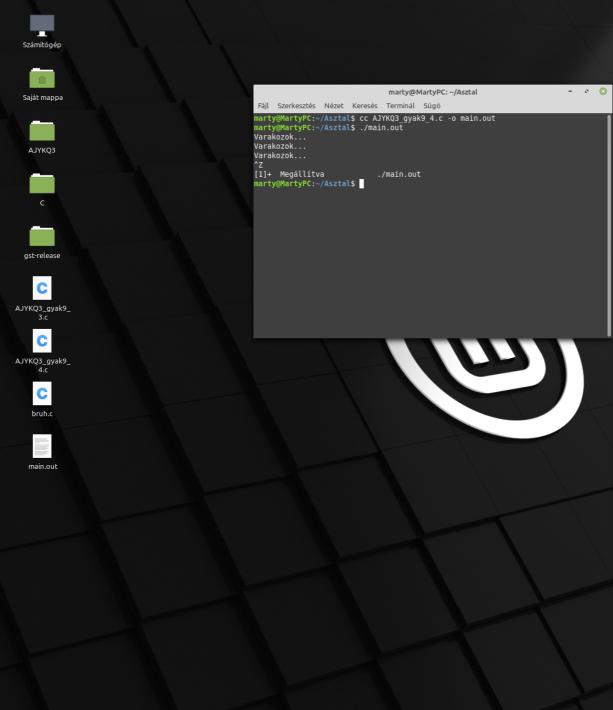












```
AJYKQ3_gyak9_4.c (~/Asztal)
Fájl Szerkesztés Nézet Keresés Eszközök Dokumentumok Súgó
 C AJYKQ3_gyak9_3.c × C AJYKQ3_gyak9_4.c ×
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
void terminal(int sig);
int main(void)
   if (signal(SIGTERM, terminal) == SIG_ERR)
       printf("A(z) \"SIGTERM\" jelre nem sikerult handlert allitani!\n");
       return 0;
   while(-1)
       printf("Varakozok...\n");
       sleep(3);
   return 0;
void terminal(int sig){
   signal(sig, SIG_IGN);
printf("Jel: %d\n", sig);
                                                 C ▼ Szóközök: 4 ▼ 1. sor, 1. oszlop BESZ
```

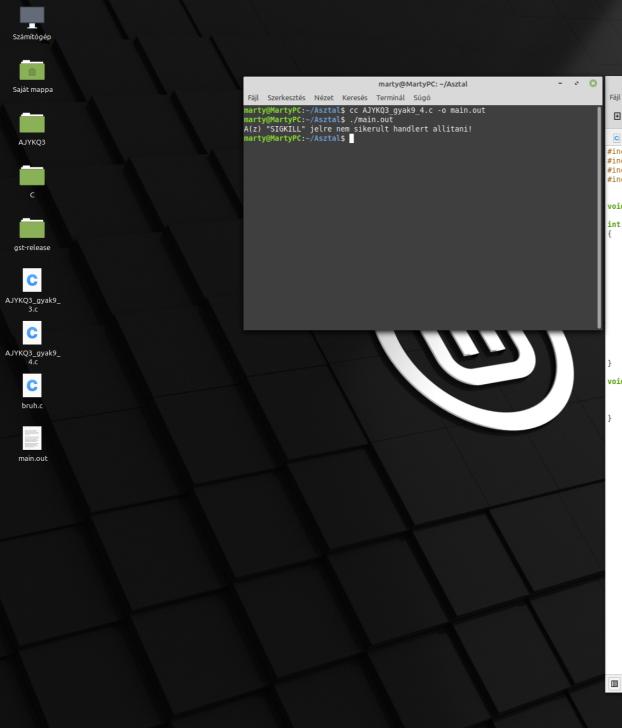












```
AJYKQ3_gyak9_4.c (~/Asztal)
Fájl Szerkesztés Nézet Keresés Eszközök Dokumentumok Súgó
 田 荷 日 | 5 c | ※ 同 🕆 🗆 🤉
 C AJYKQ3_gyak9_3.c × C AJYKQ3_gyak9_4.c ×
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
void terminal(int sig);
int main(void)
   if (signal(SIGKILL, terminal) == SIG_ERR)
       printf("A(z) \"SIGKILL\" jelre nem sikerult handlert allitani!\n");
       return 0;
   while(-1)
       printf("Varakozok...\n");
       sleep(3);
    return 0;
void terminal(int sig){
   signal(sig, SIG_IGN);
printf("Jel: %d\n", sig);
                                                   C ▼ Szóközök: 4 ▼ 13. sor, 31. oszlop BESZ
```









