

Operációs rendszerek BSc

5. Gyak.

2022. 03. 09.

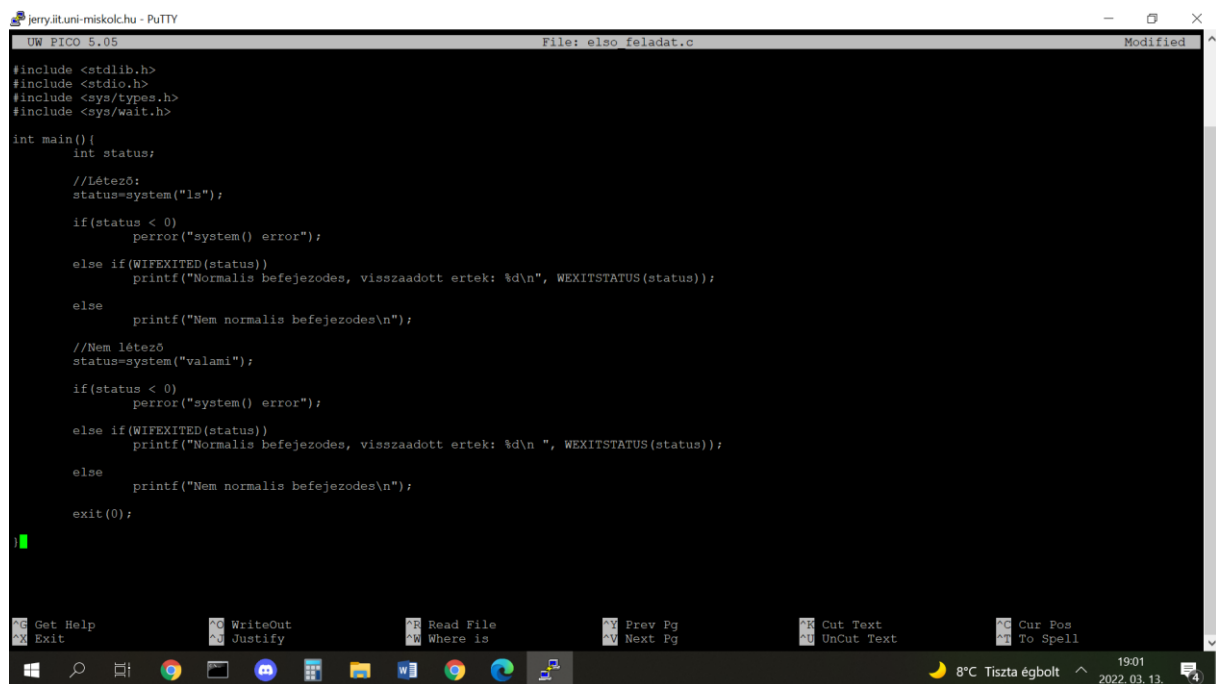
Készítette:

Berecz Antónia Bsc

Programtervező informatikus

LV8GJH

1. feladat: A system() rendszerhívással hajtson végre létező és nem létező parancsot, és vizsgálja a visszatérési értéket, magyarázza egy-egy mondattal. A fordítás/futtatás után készítsen egy képernyőképet (minden parancs esetén) és illessze be a dokumentumba.



```
#include <stdlib.h>
#include <stdio.h>
#include <sys/types.h>
#include <sys/wait.h>

int main() {
    int status;

    //Létező:
    status=system("ls");

    if(status < 0)
        perror("system() error");

    else if(WIFEXITED(status))
        printf("Normalis befejezodes, visszaadott ertekek: %d\n", WEXITSTATUS(status));

    else
        printf("Nem normalis befejezodes\n");

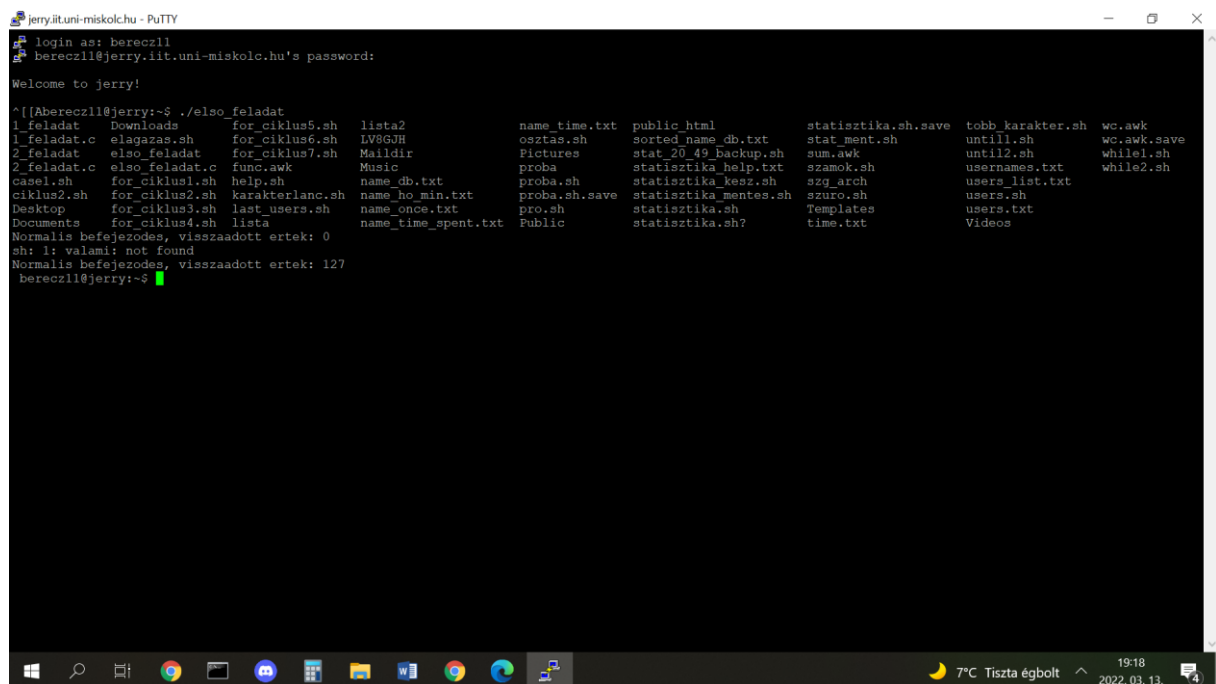
    //Nem létező
    status=system("valami");

    if(status < 0)
        perror("system() error");

    else if(WIFEXITED(status))
        printf("Normalis befejezodes, visszaadott ertekek: %d\n ", WEXITSTATUS(status));

    else
        printf("Nem normalis befejezodes\n");

    exit(0);
}
```



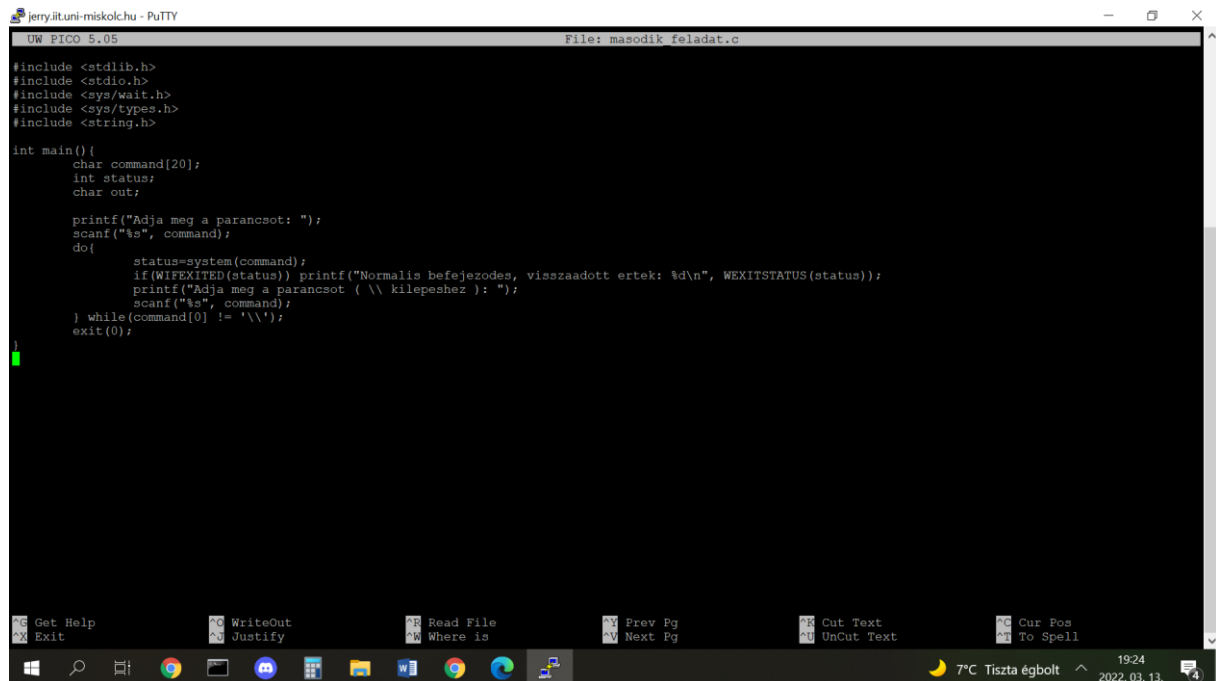
```
login as: bereczll
berezll@jerry.iit.uni-miskolc.hu's password:
Welcome to jerry!

^[[Aberezll@jerry:~$ ./elso_feladat
1 feladat Downloads for_ciklus5.sh lista2 name_time.txt public_html statisztika.sh.save tobb_karakter.sh wc.awk
1 feladat.c elagazas.sh for_ciklus6.sh LV8GJH osztas.sh sorted_name_db.txt stat_ment.sh untill.sh wc.awk.save
2 feladat.c elso_feladat.c for_ciklus7.sh Maildir Pictures stat_20_49_Backup.sh sum.awk untill2.sh while1.sh
2 feladat.c elso_feladat.c func.awk Music proba statisztika_help.txt szamok.sh usernames.txt while2.sh
case1.sh for_ciklus1.sh help.sh name_db.txt proba.sh statisztika_kesz.sh szg_arch users_list.txt users.sh
ciklus2.sh for_ciklus2.sh karakterlanc.sh name_ho_min.txt proba.sh.save statisztika_mentes.sh szuro.sh users.sh
Desktop for_ciklus3.sh last_users.sh name_once.txt pro.sh statisztika.sh Templates users.txt
Documents for_ciklus4.sh lista name_time_spent.txt Public statisztika.sh? time.txt Videos

Normalis befejezodes, visszaadott ertekek: 0
sh: 1: valami: not found
Normalis befejezodes, visszaadott ertekek: 127
berezll@jerry:~$
```

A valós parancs után 0-val, a nem valós után 127-tel tér vissza.

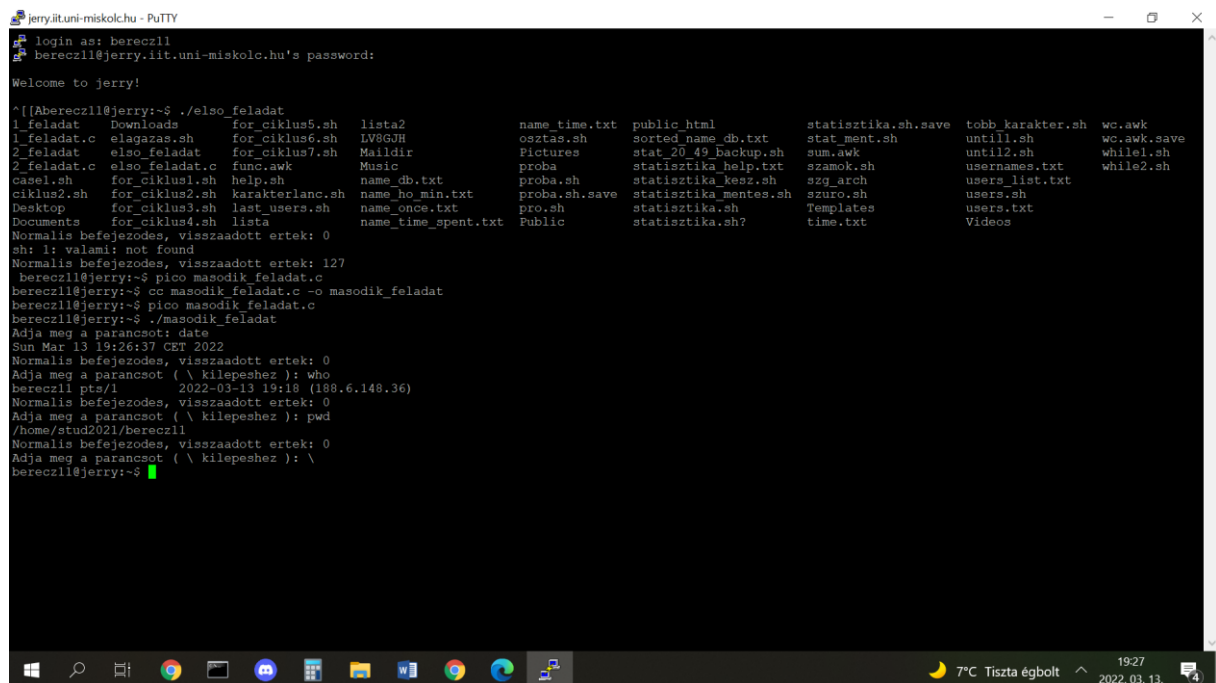
2. feladat: Írjon programot, amely billentyűzetről bekér Unix parancsokat és végrehajtja őket, majd kiírja a szabványos kimenetre. (pl.: amit bekér: date, pwd, who etc.; kilépés: CTRL-\) - magyarázza egy-egy mondattal A fordítás/futtatás után készítsen egy képernyőképet (minden parancs esetén) és illesse be a dokumentumba.



```
#include <stdlib.h>
#include <stdio.h>
#include <sys/wait.h>
#include <sys/types.h>
#include <string.h>

int main() {
    char command[20];
    int status;
    char out;

    printf("Adja meg a parancsot: ");
    scanf("%s", command);
    do {
        status=system(command);
        if(WIFEXITED(status)) printf("Normalis befejezodes, visszaadott ertek: %d\\n", WEXITSTATUS(status));
        printf("Adja meg a parancsot ( \\ kilepeshez ): ");
        scanf("%s", command);
    } while(command[0] != '\\');
    exit(0);
}
```



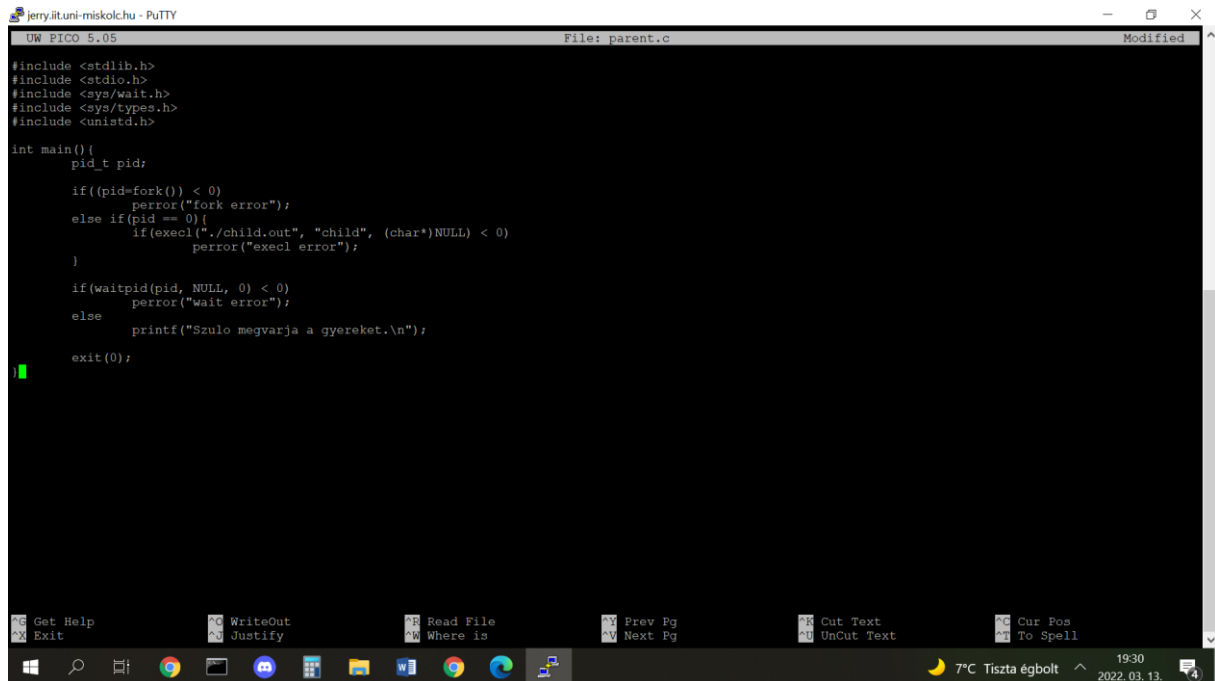
```
login as: berecz11
berez11@jerry.iit.uni-miskolc.hu's password:
Welcome to jerry!

^[[Aberez11@jerry:~$ ./elso_feladat
1 feladat Downloads for ciklus5.sh lista2 name_time.txt public_html statisztika.sh.save tobb_karakter.sh wc.awk
1 feladat.c elagazas.sh for ciklus6.sh LV8GJH osztas.sh sorted_name_db.txt stat_ment.sh untill.sh wc.awk.save
2 feladat elso_feladat for ciklus7.sh Maildir Music proba Pictures stat_20_49_backup.sh sum.awk untill2.sh while1.sh
2 feladat.c elso_feladat.c func.awk Casel.sh for ciklus1.sh help.sh name_db.txt proba.sh proba.sh.save statisztika_kesz.sh szg_arch usernames.txt while2.sh
ciklus2.sh for ciklus2.sh karakterlanc.sh name_ho_min.txt proba.sh proba.sh.save statisztika_mentes.sh szuro.sh users.sh users.sh
Desktop for ciklus3.sh last_users.sh name_once.txt pro.sh statisztika.sh Templates users.txt
Documents for ciklus4.sh lista name_time_spent.txt Public statisztika.sh? time.txt Videos

Normalis befejezodes, visszaadott ertek: 0
sh: 1: valami: not found
Normalis befejezodes, visszaadott ertek: 127
berez11@jerry:~$ pico masodik_feladat.c
berez11@jerry:~$ cc masodik_feladat.c -o masodik_feladat
berez11@jerry:~$ ./masodik_feladat
Adja meg a parancsot: date
Sun Mar 13 19:26:37 CET 2022
Normalis befejezodes, visszaadott ertek: 0
Adja meg a parancsot ( \\ kilepeshez ): who
berez11 pts/1 2022-03-13 19:19 (188.6.148.36)
Normalis befejezodes, visszaadott ertek: 0
Adja meg a parancsot ( \\ kilepeshez ): pwd
/home/stud2021/berez11
Normalis befejezodes, visszaadott ertek: 0
Adja meg a parancsot ( \\ kilepeshez ): \
berez11@jerry:~$
```

Bármelyik programot lefuttatja, amelynek bemenetként megadjuk a nevét.

3. feladat: Készítsen egy XY_parent.c és a XY_child.c programokat. A XY_parent.c elindít egy gyermek processzt, ami különbözik a szülőtől. A szülő megvárja a gyermek lefutását. A gyermek szöveget ír a szabványos kimenetre (10-ször) (pl. a hallgató neve és a neptunkód)! - magyarázza egyegy mondattal A fordítás/futtatás után készítsen egy képernyőképet (minden parancs esetén) és illessze be a dokumentumba.



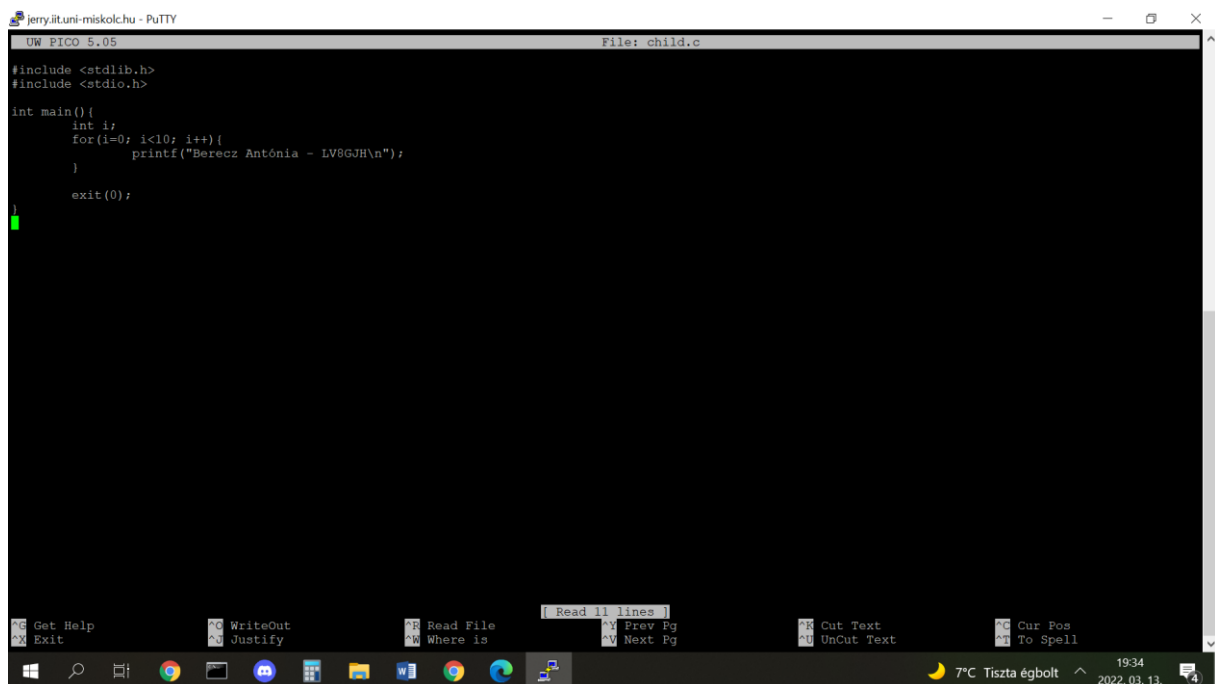
```
File: parent.c
#include <stdlib.h>
#include <stdio.h>
#include <sys/wait.h>
#include <sys/types.h>
#include <unistd.h>

int main() {
    pid_t pid;

    if((pid=fork()) < 0)
        perror("fork error");
    else if(pid == 0){
        if(execl("./child.out", "child", (char*)NULL) < 0)
            perror("execl error");
    }

    if(waitpid(pid, NULL, 0) < 0)
        perror("wait error");
    else
        printf("Szulo megvarja a gyereket.\n");

    exit(0);
}
```



```
File: child.c
#include <stdlib.h>
#include <stdio.h>

int main() {
    int i;
    for(i=0; i<10; i++){
        printf("Berecz Antónia - LV8GJH\n");
    }

    exit(0);
}
```

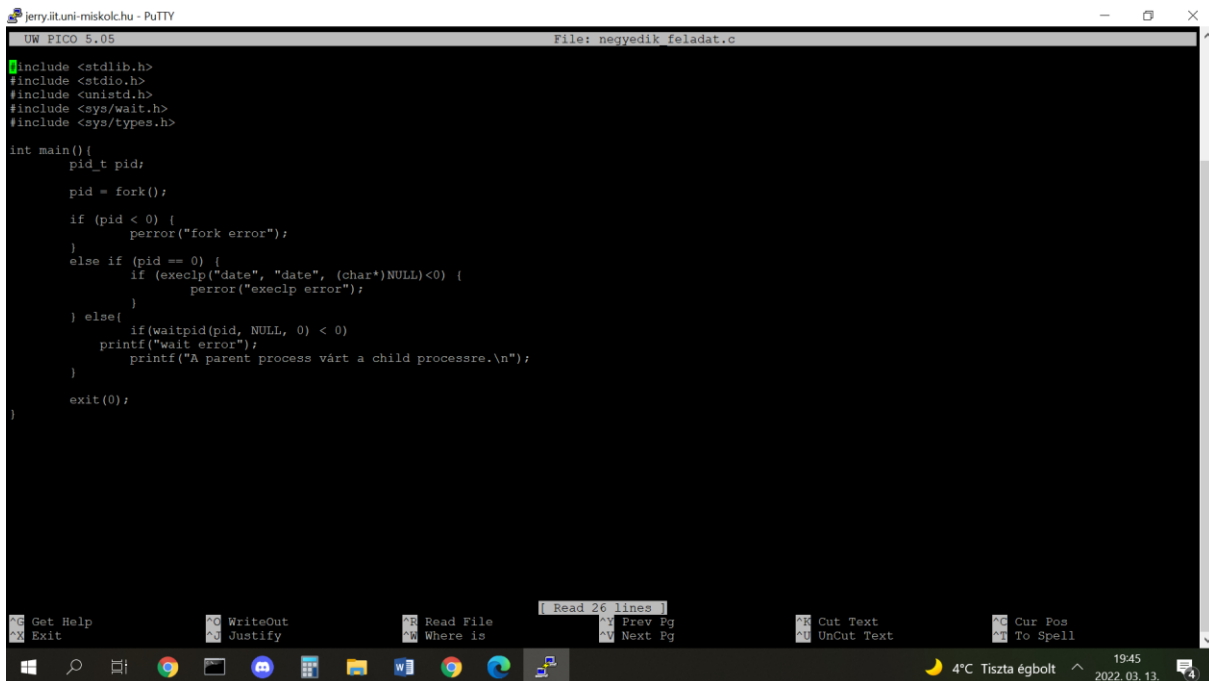
```
jerry.ilituni-miskolc.hu - PuTTY
Welcome to jerry!

^[[Aberecz11@jerry:~$ ./also_feladat
1 feladat Downloads for_ciklus5.sh lists2 name_time.txt public.html statisztika.sh.save tobb_karakter.sh wc.awk
1 feladat.c elagazas.sh for_ciklus6.sh LV8GJH osztas.sh sorted_name_db.txt stat_ment.sh until1.sh wc.awk.save
2 feladat also_feladat for_ciklus7.sh Maildir Pictures stat_20_49_Backup.sh sum.awk until2.sh while1.sh
2 feladat.c also_feladat.c func.awk Music proba statisztika_help.txt szamok.sh usernames.txt while2.sh
casel.sh for_ciklus1.sh help.sh name_db.txt proba.sh statisztika_kesz.sh szg_arch users_list.txt
ciklus2.sh for_ciklus2.sh karakterlanc.sh name_ho_min.txt proba.sh.save statisztika_mentes.sh szuro.sh users.sh
Desktop for_ciklus3.sh last_users.sh name_once.txt pro.sh statisztika.sh Templates users.txt
Documents for_ciklus4.sh lista name_time_spent.txt Public statisztika.sh? time.txt Videos

Normalis befejezodes, visszaadott ertekek: 0
sh: 1: valami: not found
Normalis befejezodes, visszaadott ertekek: 127
berez11@jerry:~$ pico masodik_feladat.c
berez11@jerry:~$ cc masodik_feladat.c -o masodik_feladat
berez11@jerry:~$ pico masodik_feladat.c
berez11@jerry:~$ ./masodik_feladat
Adja meg a parancsot: date
Sun Mar 13 19:26:37 CET 2022
Normalis befejezodes, visszaadott ertekek: 0
Adja meg a parancsot ( \ Kilepeshez ): who
berez11 pts/1 2022-03-13 19:18 (188.6.148.36)
Normalis befejezodes, visszaadott ertekek: 0
Adja meg a parancsot ( \ Kilepeshez ): pwd
/home/stud2021/berez11
Normalis befejezodes, visszaadott ertekek: 0
Adja meg a parancsot ( \ Kilepeshez ): \
berez11@jerry:~$ pico parent.c
berez11@jerry:~$ cc parent.c -o parent
berez11@jerry:~$ pico child.c
berez11@jerry:~$ cc child.c -o child
berez11@jerry:~$ pico child.c
berez11@jerry:~$ ./parent
exec: error: No such file or directory
wait error: No child processes
Szulo megvarja a gyereket.
berez11@jerry:~$ pico parent.c
berez11@jerry:~$ pico child.c
berez11@jerry:~$ ./parent
exec: error: No such file or directory
wait error: No child processes
Szulo megvarja a gyereket.
berez11@jerry:~$
```

A parent processnek meg kell hívnia a child processt, majd megvárnia azt, de nálam valami nem működött és nem tudtam rájönni a probléma okára.

4. feladat: A `fork()` rendszerhívással hozzon létre egy gyerek processzt-t és abban hívjon meg egy `exec` családbeli rendszerhívást (pl. `execlp`). A szülő várja meg a gyerek futását! - magyarázza egy-egy mondattal. A fordítás/futtatás után készítsen egy képernyőképet (minden parancs esetén) és illessze be a dokumentumba.



The screenshot shows a Windows desktop with a terminal window titled "jerry.it.uni-miskolc.hu - PuTTY". The terminal displays the C code for the 4th task, which uses `fork()` to create a child process and `execlp()` to execute the `date` command. The code is as follows:

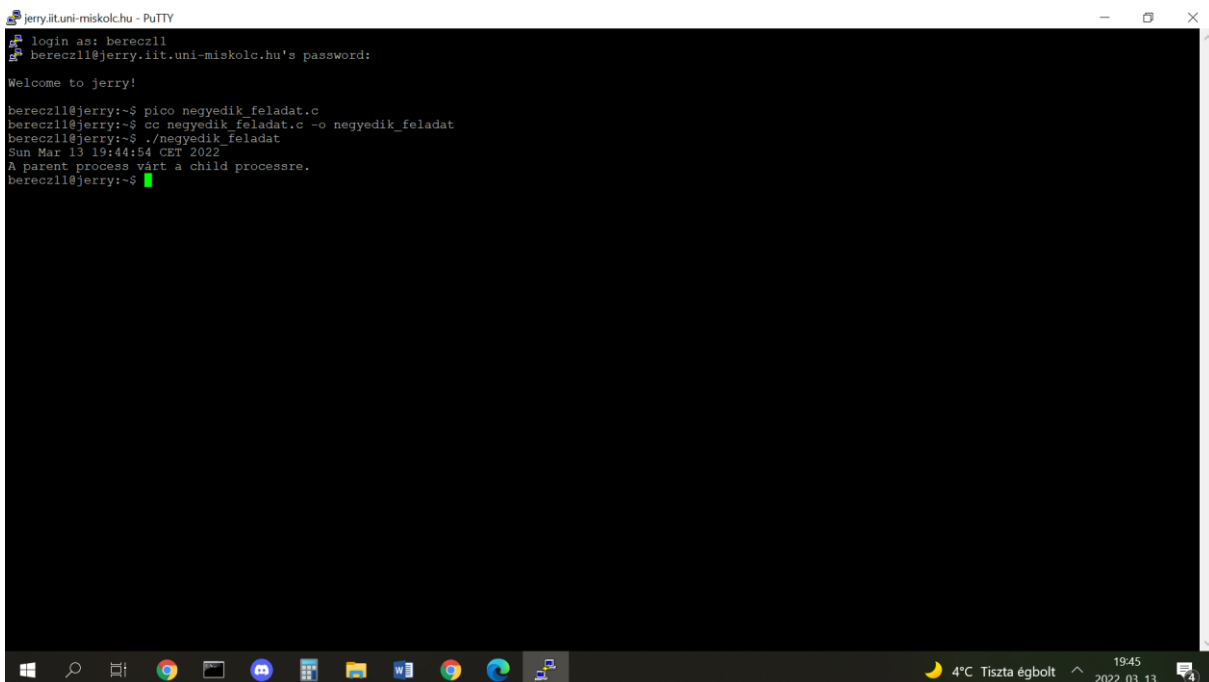
```
#include <stdlib.h>
#include <stdio.h>
#include <unistd.h>
#include <sys/wait.h>
#include <sys/types.h>

int main() {
    pid_t pid;

    pid = fork();

    if (pid < 0) {
        perror("fork error");
    }
    else if (pid == 0) {
        if (execlp("date", "date", (char*)NULL) < 0) {
            perror("execlp error");
        }
    }
    else {
        if (waitpid(pid, NULL, 0) < 0)
            printf("wait error");
        printf("A parent process várt a child processre.\n");
    }

    exit(0);
}
```



The screenshot shows the terminal window after the C code has been compiled and executed. The output is as follows:

```
login as: bereczll
berezll@jerry.it.uni-miskolc.hu's password:
Welcome to jerry!

berezll@jerry:~$ pico negyedik_feladat.c
berezll@jerry:~$ cc negyedik_feladat.c -o negyedik_feladat
berezll@jerry:~$ ./negyedik_feladat
Sun Mar 13 19:44:54 CET 2022
A parent process várt a child processre.
berezll@jerry:~$
```

A parent process meghívott egy gyermekprocesszt-t.

5. feladat: A `fork()` rendszerhívással hozzon létre gyerekeket, várja meg és vizsgálja a befejeződési állapotokat (gyerekekben: `exit`, `abort`, nullával való osztás)! - magyarázza egy-egy mondatnál! A fordítás/futtatás után készítsen egy képernyőképet (minden parancs esetén) és illessze be a dokumentumba.



```
File: otodik_feladat.c Modified
UW PICO 5.05

#include <stdlib.h>
#include <stdio.h>
#include <sys/wait.h>
#include <unistd.h>

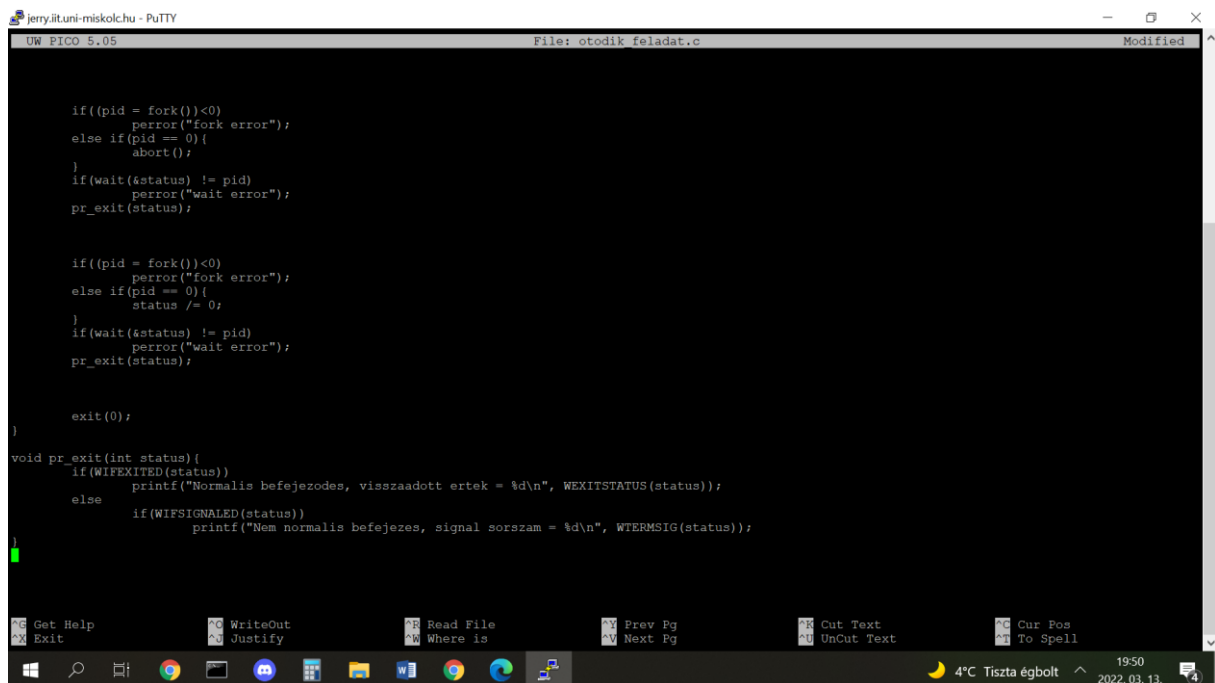
void pr_exit(int status);

int main() {
    pid_t pid;
    int status;

    if((pid = fork())<0)
        perror("fork error");
    else if(pid == 0){
        exit(10);
    }
    if(wait(&status) != pid)
        perror("wait error");
    pr_exit(status);

    if((pid = fork())<0)
        perror("fork error");
    else if(pid == 0){
        abort();
    }
    if(wait(&status) != pid)
        perror("wait error");
    pr_exit(status);

    if((pid = fork())<0)
        perror("fork error");
    else if(pid == 0){
        status /= 0;
    }
}
```



```
File: otodik_feladat.c Modified
UW PICO 5.05

    if((pid = fork())<0)
        perror("fork error");
    else if(pid == 0){
        abort();
    }
    if(wait(&status) != pid)
        perror("wait error");
    pr_exit(status);

    if((pid = fork())<0)
        perror("fork error");
    else if(pid == 0){
        status /= 0;
    }
    if(wait(&status) != pid)
        perror("wait error");
    pr_exit(status);

    exit(0);
}

void pr_exit(int status){
    if(WIFEXITED(status))
        printf("Normalis befejezodes, visszaadott ertek = %d\n", WEXITSTATUS(status));
    else
        if(WIFSIGNALED(status))
            printf("Nem normalis befejezes, signal sorszam = %d\n", WTERMSIG(status));
}

19:50 4°C Tiszta égbolt 2022.03.13.
```

```
jerry.iit.uni-miskolc.hu - PuTTY
login as: berecz11
berez11@jerry.iit.uni-miskolc.hu's password:

Welcome to jerry!

berez11@jerry:~$ pico negyedik_feladat.c
berez11@jerry:~$ cc negyedik_feladat.c -o negyedik_feladat
berez11@jerry:~$ ./negyedik_feladat
Sun Mar 13 19:44:54 CET 2022
A parent process várt a child processre.
berez11@jerry:~$ pico negyedik_feladat.c
berez11@jerry:~$ pico otodik_feladat.c
berez11@jerry:~$ cc otodik_feladat.c -o otodik_feladat
otodik_feladat.c: In function 'main':
otodik_feladat.c:38:10: warning: division by zero [-Wdiv-by-zero]
    status /= 0;
           ^~
berez11@jerry:~$ pico otodik_feladat.c
berez11@jerry:~$ pico otodik_feladat.c
berez11@jerry:~$ ./otodik_feladat
Normalis befejezodes, visszaadott ertekek = 10
Abnormalis befejezes, szignal sorszama = 6
Abnormalis befejezes, szignal sorszama = 8
berez11@jerry:~$
```

1. A 0-val osztásnál lefut a program, de hibát ad.
2. Normális befejezés értéket kap, 10-el tér vissza.
3. Hibával tér vissza, 6-os (abort) kóddal.
4. Hibával tér vissza, 8-as (aritmetikai hiba) kóddal