

JEGYZŐKÖNYV

Operációs rendszerek BSc

2021. tavasz féléves feladat

Készítette: **Novák László Lajos**

Neptunkód: **T8F0F0**

A feladat leírása:

Írjon C nyelvű programot, amely billentyűzetről bekér Unix parancsokat és végrehajtja őket, pontosan úgy, ahogy a shell.

A feladat elkészítésének lépései:

A feladatot Linux NANO segítségével oldottam meg.

A feladat két főbb részre tagolható. Az első a parancsok beolvasása, melynek programkódja a következő:

```
#define _GNU_SOURCE
#include <stdio.h>
#include <string.h>
#include <unistd.h>

int main()
{
    // változók deklarálása
    char * binaryPath;
    char parancs[30];
    char args[3][30];
    int i = 0;

    // parancs bekérése
    printf("Add meg a SHELL parancsot: ");
    fgets(parancs, sizeof(parancs), stdin);

    //parancs feldarabolása
    char * token = strtok(parancs, " ");
    while(token != NULL)
    {
        strcpy(args[i], token);
        token = strtok(NULL, " ");
        i++;
    }
}
```

A parancsot bekérem a szabványos bemenetről, majd strtok segítségével szóközőnként tagolom azt.

A kód második részében execl segítségével hívom meg a parancshoz tartozó programot. Első lépésben az első argumentumot hozzáfűzöm a programok elési útjához: „/bin/”.

Ezután attól függően, hogy hánytagú a parancs, egy switch case szerkezetben meghívom a megfelelő alakú execl függvényt:

```

}

//a program hozzáfűzése az elérési útjához
if (asprintf(&binaryPath, "/bin/%s", args[0]) == -1)
{
    perror("asprintf");
    return 1;
}

//különböző tagszámú parancsok lekezelése
switch(i)
{
    case 1:
        execl(binaryPath, binaryPath, NULL, NULL, NULL);
        break;
    case 2:
        execl(binaryPath, binaryPath, args[1], NULL, NULL);
        break;
    case 3:
        execl(binaryPath, binaryPath, args[1], args[2], NULL);
        break;
    default:
        break;
}

return 0;
}

```

A futtatás eredménye:

A fordítás és futtatás során szintaktikából adódó hibát tapasztaltam, ugyanis a beírt kódot fel tudtam darabolni, az első tagot sikeresen össze tudtam fűzni és a tagokat megfelelő formában ki is tudtam írni, azonban a execl függvény argumentumába írva az nem futott le megfelelően. Amikor ugyanolyan formában kézzel írtam be az argumentumokat az execl-be, akkor viszont az gond nélkül működött:

```

execl("/bin/ls", "/bin/ls", "-l", NULL, NULL);
/*
nlaszlo94@nlaszlo94-VirtualBox:~/Documents/beadando$ gcc T8F0F0_feleves_feladat.c -o feladat
nlaszlo94@nlaszlo94-VirtualBox:~/Documents/beadando$ ./feladat
Add meg a SHELL parancsot: ls -l
total 132
-rwxrwxr-x 1 nlaszlo94 nlaszlo94 16768 May  9 22:49 child
-rw-rw-r-- 1 nlaszlo94 nlaszlo94  120 May  9 22:33 child.c
-rwxrwxr-x 1 nlaszlo94 nlaszlo94 17056 May 10 04:50 feladat
-rwxrwxr-x 1 nlaszlo94 nlaszlo94 16696 May  9 21:59 hello
-rw-rw-r-- 1 nlaszlo94 nlaszlo94   60 May  9 21:48 hello.c
drwxrwxr-x 2 nlaszlo94 nlaszlo94  4096 May 10 01:32 nevesített
drwxrwxr-x 2 nlaszlo94 nlaszlo94  4096 May 10 02:22 nevtelen
-rwxrwxr-x 1 nlaszlo94 nlaszlo94 16840 May  9 22:49 parent
-rw-rw-r-- 1 nlaszlo94 nlaszlo94   160 May  9 22:23 parent.c
-rwxrwx--- 1 nlaszlo94 nlaszlo94  1282 May 10 04:10 shell.c
-rwxrwxr-x 1 nlaszlo94 nlaszlo94 17240 May 10 00:58 shellproba
-rwxrwx--- 1 nlaszlo94 nlaszlo94  1370 May 10 00:58 shellproba.c
-rwxrwx--- 1 nlaszlo94 nlaszlo94  1234 May 10 04:50 T8F0F0_feleves_feladat.c
nlaszlo94@nlaszlo94-VirtualBox:~/Documents/beadando$

```

Több összefűzési módszert is kipróbáltam, illetve több fórumot is átjártam, miután arra a következtetésre jutottam, hogy a c nyelv string kezelésének hiányosságaiból fakad ez a hiba.