

# **Operációs rendszerek Bsc**

2. Gyak.

2022.04.07.

**Készítette:**

Német Bence Bsc  
Mérnökinformatika  
B2Z5XN

**Miskolc, 2022**

**1. feladat:** A `system()` rendszerhívással hajtson végre létező és nem létező parancsot, és vizsgálja a visszatérési értéket, magyarázza egy-egy mondattal.

```
b2z5xn@gyak:~/b2z5xn/gyak$ ./b2z5xn_1fel
2022. ápr. 7., csütörtök, 13:54:17 CEST
date: érvénytelen kapcsoló -- "n"
További információkért adja ki a(z) „date --help” parancsot.
b2z5xn@gyak:~/b2z5xn/gyak$
```

A program a „date” és a „date -n” parancsot futtatja le. A kimenetelből látszik, hogy a „date”-re kiírta az aktuális dátumot. A „date -n” érvénytelen parancs így a parancssor kiírja, hogy érvénytelen.

**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-\)

```
b2z5xn@gyak:~/b2z5xn/gyak$ ./b2z5xn_2fel
date
2022. ápr. 7., csütörtök, 13:55:12 CEST
b2z5xn@gyak:~/b2z5xn/gyak$ ./b2z5xn_2fel
pwd
/home/b2z5xn/b2z5xn/gyak
b2z5xn@gyak:~/b2z5xn/gyak$ ./b2z5xn_2fel
who
b2z5xn    tty7                2022-04-07 13:39 (:0)
b2z5xn@gyak:~/b2z5xn/gyak$ ./b2z5xn_2fel
^C
b2z5xn@gyak:~/b2z5xn/gyak$
```

A programot lefuttatva várja a program a billentyűzetről való bevitelt. A képen láthatóak azok a parancsok, amiket beírtam a programba és sikeresen lefutottak.

**3. feladat:** Készítsen egy `parent.c` és a `child.c` programokat. A `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)!

```
b2z5xn@gyak:~/b2z5xn/gyak$ ./parent
1. Nemet Bence - B2Z5XN
2. Nemet Bence - B2Z5XN
3. Nemet Bence - B2Z5XN
4. Nemet Bence - B2Z5XN
5. Nemet Bence - B2Z5XN
6. Nemet Bence - B2Z5XN
7. Nemet Bence - B2Z5XN
8. Nemet Bence - B2Z5XN
9. Nemet Bence - B2Z5XN
10. Nemet Bence - B2Z5XN
b2z5xn@gyak:~/b2z5xn/gyak$
```

A parent programot elindítva létrehozta a child programos és 10 sorban kiírta a child a nevemet és a neptunkódomat.

**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!

```
b2z5xn@gyak:~/b2z5xn/gyak$ ./b2z5xn_4fel
1. Nemet Bence - B2Z5XN
2. Nemet Bence - B2Z5XN
3. Nemet Bence - B2Z5XN
4. Nemet Bence - B2Z5XN
5. Nemet Bence - B2Z5XN
6. Nemet Bence - B2Z5XN
7. Nemet Bence - B2Z5XN
8. Nemet Bence - B2Z5XN
9. Nemet Bence - B2Z5XN
10. Nemet Bence - B2Z5XN
b2z5xn@gyak:~/b2z5xn/gyak$
```

A programban a fork segítségével létrehoztam a child processt és ebben egy exec családbeli rendszerhívást helyeztem el.

**5. feladat:** A `fork()` rendszerhívással hozzon létre gyerekeket, várja meg és vizsgálja a befejeződési állapotokat (gyerekekben: `exit`, `abort`, nullával való osztás)!

```
b2z5xn@gyak:~/b2z5xn/gyak$ ./b2z5xn_5fel
exit Exit status of the child was 0
abort Exit status of the child was 0
Exit status of the child was 0
b2z5xn@gyak:~/b2z5xn/gyak$
```

Létrehoz a program egy child process-t és megvizsgálja a befejeződési állapotokat.