**Operációs rendszerek BSc**

5.gyak.

2021. 03. 10.

**Készítette:**

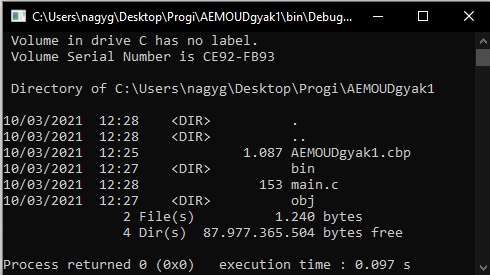
# Nagy Gergő Bsc Programtervező Informatikus szak

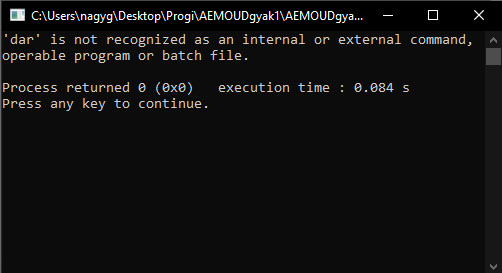
AEMOUD

**Miskolc, 2021**

**1) Feladat -** A system() rendszerhívással hajtson végre létező és nem létező parancsot, és vizsgálja a visszatérési érteket!

Megvalósítás: Létező parancs esetén lefuttatja az adott parancsot. Pl.: „dir”



Nem létező parancs esetén hibát dob, és leáll a program. Pl.: „dar”

Mentett c program: AEMOUDgyak1.c

**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-\)

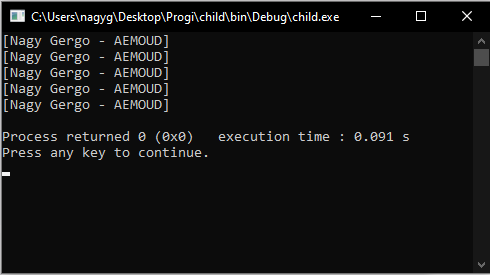
Megvalósítás:

Mentett c program: AEMOUDgyak2.c

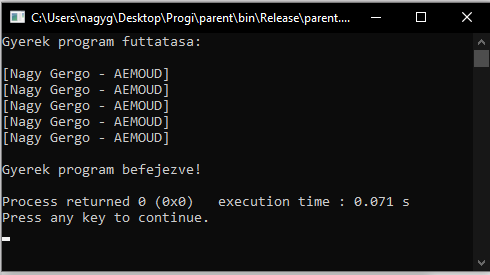
**3) Feladat -** Készítsen egy parent.c és a child.c programokat. A parent.c elindít egy gyermek processzt, ami különbözik a szülőtől. A szülő megvárja a gyermek lefutását. A gyermek szöveget ír a szabványos kimenetre (5-ször) (pl. a hallgató neve és a neptunkód)!

Megvalósítás:

* Gyerek program:

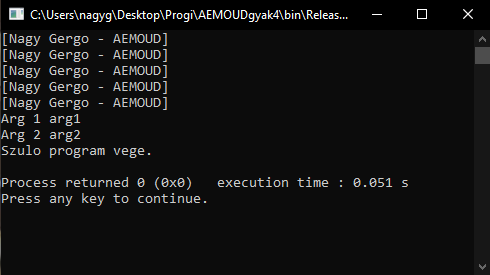


* Szülő program:



Mentett c program: parent.c; child.c

**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!

Megvalósítás:

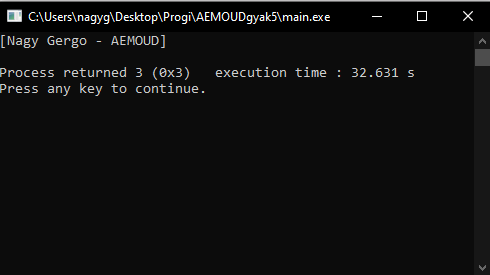
Mentett c program: AEMOUDgyak4.c

**5) Feladat -** A fork() rendszerhívással hozzon létre gyerekeket, várja meg és vizsgálja a befejeződési állapotokat (gyerekben: exit, abort, nullával való osztás)!

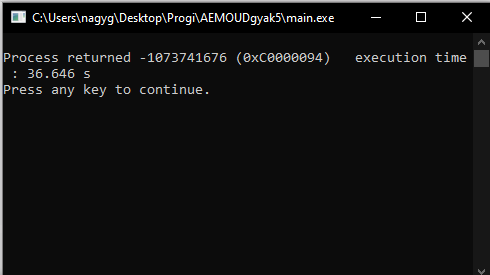
Megjegyzés: a fork() nem fut le CodeBlocks-ban, úgyhogy egy azonos működésű kódot írtam.

Megvalósítás:

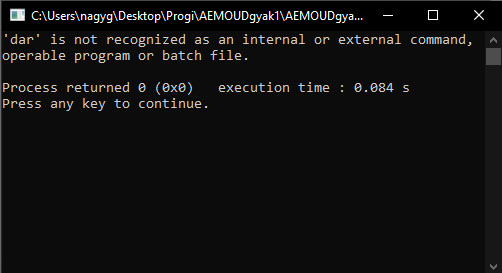
* Exit: Exit esetén a szülő visszakapja a vezérlést. A gyerek program nem fejeződik be:



* abort: Abort esetén hasonlóan az előzőhöz, a szülő visszakapja a vezérlést. A gyerek program nem fejeződik be:



* x/0: 0-val való osztás esetén a gyerek program hibát dobna, és véget érne, viszont mivel szülőből indítottuk, ezért nem áll le az egész program, csak a gyerek, a szülő végig lefut:



Mentett c program: AEMOUDgyak5.c

**6) Feladat -** Adott a következő ütemezési feladat, ahol a RR ütemezési algoritmus használatával készítse el: Határozza meg a

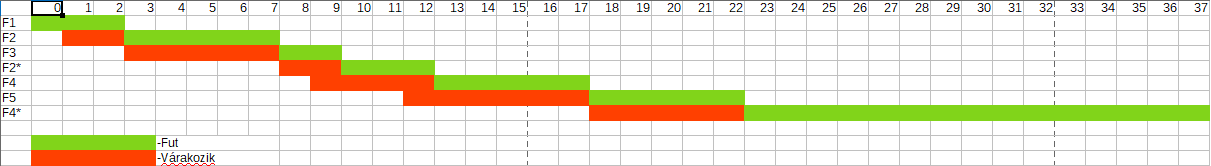
1. Ütemezze az adott időszelet alapján az egyes processzek paramétereit (ms)!
2. A rendszerben lévő processzek végrehajtásának sorrendjét?
3. Ábrázolja Gantt diagram segítségével az aktív/várakozó processzek futásának menetét!

Megvalósítás:

a)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | F1 | F2 | F3 | F4 | F5 |
| Érkezés | 0 | 1, 8 | 3 | 9, 18 | 12 |
| CPU idő | 3 | 8, 3 | 2 | 20, 15 | 5 |
| Indulás | 0 | 3, 10 | 8 | 13, 23 | 18 |
| Befejezés | 3 | 8, 13 | 10 | 18, 38 | 23 |
| Várakozás | 0 | 2, 2 | 5 | 4, 5 | 6 |

b) F1→F2→F3→F2\*→F4→F5→F4\*

c)