

JEGYZŐKÖNYV

Operációs rendszerek BSc

2021. tavasz féléves feladat

Készítette: Babik Szilárd Kristóf

Neptunkód: A6NQW1

A feladat leírása:

Írjon C nyelvű programot, ami:

1. létrehoz egy gyermek processzt
2. az újonnan létrejött processzben végrehajtja a `"/bin/ls -R/"` parancsot a paramétereivel együtt, térjen vissza valamilyen 0-tól különböző értékkel
3. a szülő várja meg amíg befejeződik a processz, és írja ki a visszatérési értéket

A feladat elkészítésének lépései:

A feladatot VirtualBoxon keresztül telepített Ubuntu rendszeren írtam meg CodeBlocks fejlesztőkörnyezetben.

- 1. lépés:** includeáltam a szükséges header állományokat, amikre szükségem volt a feladat megoldásához.
- 2. lépés:** megírtam a gyermek processz létrehozását lehetővé tevő kódot, megfelelő hibakezeléssel.
- 3. lépés:** megírtam a child processz kódját. Ez a következők szerint történt meg:

A feladatileírásnak megfelelően, létrehoz a `fork()` függvénnyel egy gyermek processzt, amiben kétféle módszerrel tud lefutni az „ls -R” parancs.

1. megoldás: az `execl()` függvény segítségével futtatja le a parancsot.

2. megoldás: a `system()` függvény segítségével futtatja le a parancsot.

A program jelenlegi állapotában kommentezés segítségével ki van kapcsolva az `execl()` megoldás, azonban a jelenlegi állapotból könnyen megoldható az, hogy választhatóvá lehessen tenni, hogy melyik megoldással hajtódik végre a feladatkiírásban szereplő parancs.

4. lépés: megírtam a parent processz kódját. Ez tartalmazza a gyermek processz visszatérési értékének kiírását megvalósító kódot, illetve a saját visszatérési értékéről döntő kódot.

5. lépés: futtatási teszteket végeztem, megnéztem, hogy működik-e a hibakezelés. Minden előforduló hibát kijavítottam.

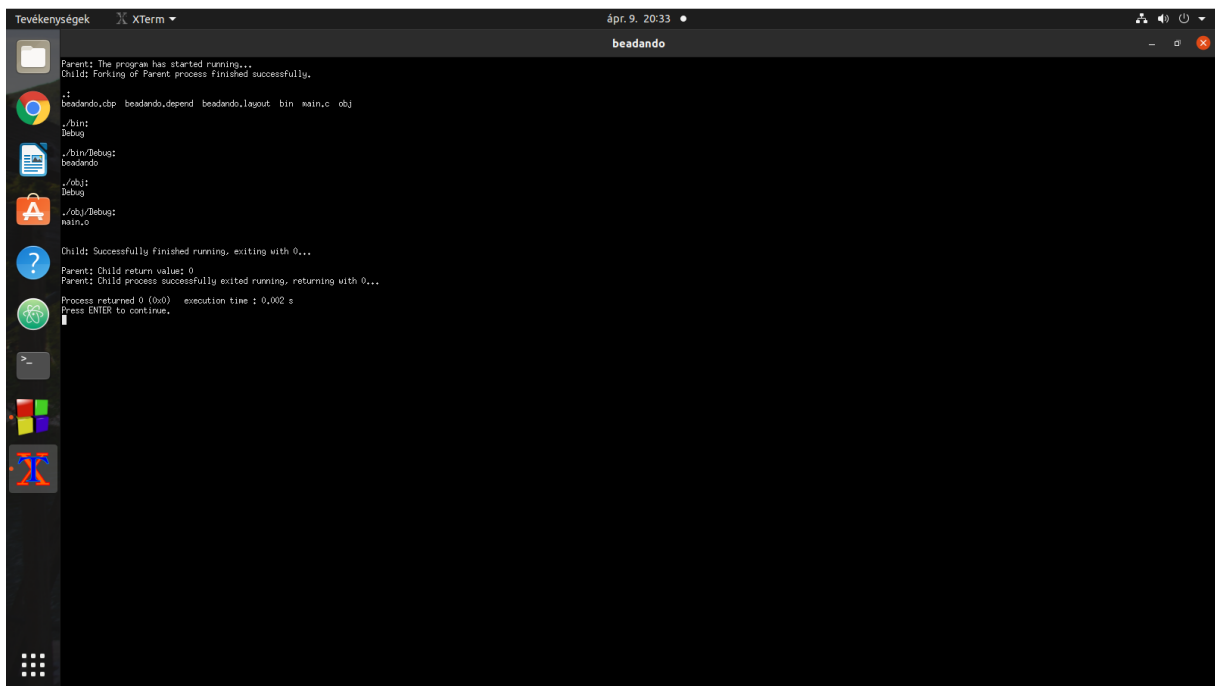
A kész forrásfájlt átmásoltam a hoszt Windows operációs rendszerre, és innen került feltöltésre a megfelelő GitHub repositoryba.

A futtatás eredménye:

A program minden esetben „hiba” nélkül lefut, minden hiba lehetőség kezelve van. A konzolra kiírt üzeneteket egyértelműsítettem „Parent:” illetve „Child:” prefixekkel, így követhetőbb a program futása és a megkülönböztethetőség is egyszerűbbé válik.

A child processz sikeres vagy sikertelen futásától függően változik a visszatérési értéke (sikeres futás esetén EXIT_SUCCESS, sikertelen futás esetén EXIT_FAILURE értékekkel), amit a parent processz ki is ír, és ennek megfelelően „dönt” a saját visszatérési értékéről. Az EXIT_SUCCESS érték esetén a visszatérési érték számban kifejezve 0 lesz.

A program futása ezen a ponton befejeződik. Screenshot a program sikeres futásáról:



```
Parent: The program has started running...
Child: Forking of Parent process finished successfully.

..
beadando.cbp beadando.depend beadando.layout bin main.o obj
./bin/
./bin/Debug:
beadando
./obj:
./obj/Debug:
main.o

Child: Successfully finished running, exiting with 0...
Parent: Child return value: 0
Parent: Child process successfully exited running, returning with 0...
Process returned 0 (0x0)   execution time : 0.002 s
Press ENTER to continue.
```