

Operációs rendszerek

3.Gyakorlat

2025.03.05.

Készítette:

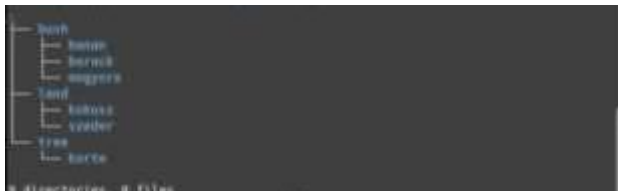
Ródé Batta István

Szak: Programtervező Informatikus

Neptunkód: WIBGTA

Sárospatak, 2025

1.feladat : Hozza létre a következő jegyzék szerkezetet, majd listázza ki.



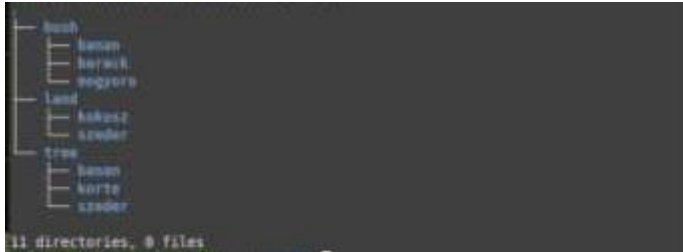
2. Készítsen másolatot:

- a neptunkod/ land/szeder katalógusról a neptunkod/tree struktúrába



Az alábbiakban a **cp -r** azaz a **copy** paranccsal készítettem másolatot a **szeder** mappáról a **tree** mappába.

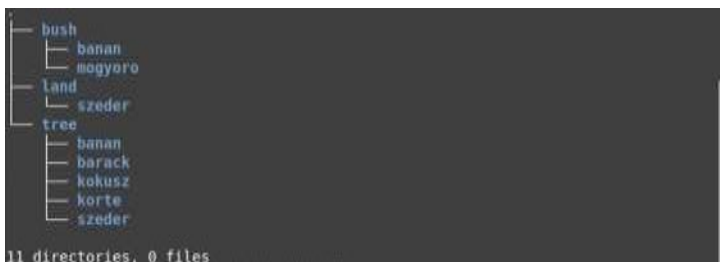
- a neptunkod /bush/banan katalógusról a neptunkod /tree struktúrába



Az alábbiakban a **cp -r** azaz a **copy** paranccsal készítettem másolatot a **banan** mappáról a **tree** mappába.

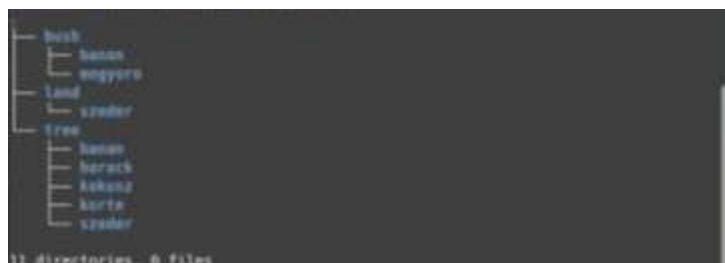
3. Végezze el a következő áthelyezéseket:

- a neptunkod / bush /barack katalógust helyezze át a neptunkod /tree katalógusba



Az alábbiakban a **mv** azaz a **move** paranccsal helyeztem át a **barack** mappát a **tree** mappába.

- a neptunkod /land /kokusz katalógust helyezze át a neptunkod/tree katalógusba



Az alábbiakban a **mv** azaz a **move** paranccsal helyeztem át a **kokusz** mappát a **tree** mappába.

4. Törölje a neptunkod/land katalógust a teljes tartalmával. Hozza létre a következő szöveges állományokat:

```
bush
├── banan
└── meggyuro
tree
├── banan
├── barack
├── kukusz
├── korte
└── szeder
0 directories, 0 files
```

Az drppxl/land katalógust a teljes tartalmával együtt **rm -r** parancs segítségével töröltem.

- neptunkod/bush/banan/ description.txt

```
bush
├── banan
│   └── description.txt
└── meggyuro
tree
├── banan
├── barack
├── kukusz
├── korte
├── listing.txt
└── szeder
0 directories, 2 files
```

A **touch** parancs segítségével létrehoztam a description.txt című fájlt. •

neptunkod/tree/listing.txt

```
bush
├── banan
│   └── description.txt
└── meggyuro
tree
├── banan
├── barack
├── kukusz
├── korte
├── listing.txt
└── szeder
0 directories, 2 files
```

A **touch** parancs segítségével létrehoztam a listing.txt című fájlt.

5. A description szöveges állományba írjon 3 sort a málnáról.

```
Málna
Málna
Málna
```

A **nano** parancs segítségével a description.txt fájlba beírtam a Málna szót háromszor, amelyet pedig a **cat** parancs segítségével nyitottam meg, és néztem meg.

A listing szöveges állományba soroljon fel külön sorba 5 olyan gyümölcsöt, amelyek tree teremnek.

```
5 öt darab fán termő gyümölcs:
Alma
Körte
Szilva
Cseresznye
Megy
```

A **nano** parancs segítségével a listing.txt fájlba beírtam öt különböző fán termő gyümölcsöt, amelyet pedig a **cat** parancs segítségével nyitottam meg, és néztem meg.

6. Listázza a neptunkod katalógus tartalmát úgy, hogy megjelenjen az alkatalógusok tartalma is.

```
tree
├── bush
│   ├── banana
│   │   └── description.txt
│   └── mogycso
└── tree
    ├── banana
    ├── barack
    ├── kokusz
    ├── korte
    ├── listing.txt
    └── szeder
9 directories, 2 files
```

A **tree** parancs segítségével listáztam az eyzwg9 katalógus tartalmát úgy, hogy megjelenjen az alkatalógusok tartalma is

Ezután listázza az aktuális (munka)katalógus nevét.

A **pwd** parancs segítségével pedig listáztam az aktuális (munka)katalógus nevét.

7. Térjen vissza a saját home katalógusába és keresse meg az összes olyan file-t, amelyek nevének második betűje e.

Ezt a műveletet a **find . -type f -name '?e*'** paranccsal tudtam megcsinálni.

8. Tegye mindenki számára olvashatóvá a listing.txt file-t.

```
Az öt darab fán termő gyümölcs:
- Alma
- Korte
- Szilva
- Cseresznye
- Megy
```

Ezt a műveletet a **chmod** parancs segítségével lehet véghez vinni.

9. Listázza ki, hogy mennyi helyet foglal a merevlemezén a neptunkod katalógus az alkatalógusaival együtt. Az alkatalógusok méretei ne jelenjenek meg.

Ezt a műveletet a **du -sh** parancs segítségével lehet véghez vinni.

10. Listázza ABC-szerint rendezve a listing.txt file tartalmát.

```
- Alma
- Cseresznye
- Korte
- Megy
- Szilva
```

Ezt a műveletet a **sort** parancs segítségével lehet megoldani.

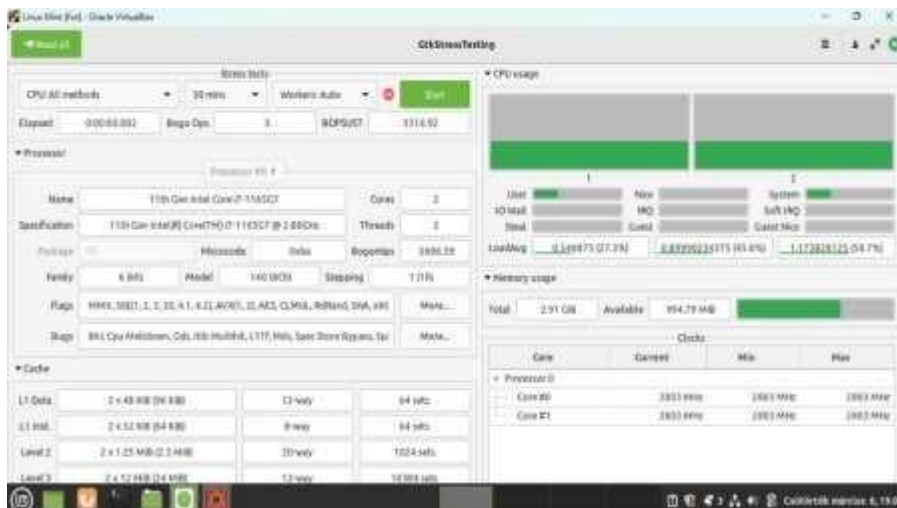
11. Számolja meg a description.txt file-ban szereplő szavakat.

Ezt a műveletet a **wc -w** parancs segítségével lehet megoldani.

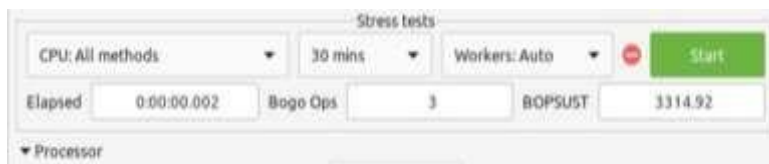
2. feladat

B) Grafikus rendszer monitorozó – GTKStressTest Telepítse a programot (gst csomag), amely un. Stresstest -elést végez: tuningolt processzorok esetén is használják, figyelik a stabilitást (hőmérséklet, feszültség). Tanulmányozza a program működését (5 kijelző) és a szolgáltatásai alapján készítsen leírást, azaz külön-külön a kijelzőket is vizsgálja és erről készítsen egy képernyőképet és illessze be a dokumentumba.

Itt található meg, ezen a képen a **GTKStressTest** alkalmazás 5 főkijelzője.



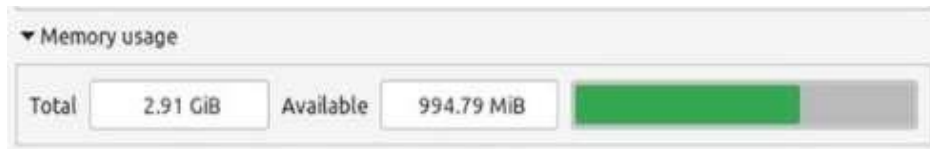
A képernyő bal oldalán található az első kijelző, amelyen állítható be a stresszteszt hossza, a párhuzamosan futó szálak száma, és innen indítható vagy állítható le a teszt is.



A kijelző jobb felső részében egy grafikus kijelző valós időben mutatja, hogy a CPU erőforrásainak hány százaléka van éppen használatban.



Ez egy sáv ami jelzi a memória igénybevételét, vagyis a rendelkezésre álló és a felhasznált RAM mennyiségét.



A Stressz test alatt található a processzorral kapcsolatos részletes adatok: típus, órajel, mikrokód, magok száma, szálak száma, támogatott utasításkészletek valamint egyéb, CPU-ra vonatkozó paraméterek.

Látható a terhelési átlag és a futó folyamatok száma.



A processzor tulajdonságai alatt jelennek meg a Core, Care és Current.

A processzor valós idejű órajeladatai.

