

# 1. laborfeladat - 2023

---

- név:
- azonosító: bbbbnnnn

## Bevezetés

A *linux* katalógusból kiindulva a *cd* parancs segítségével lépünk be a *labor1* katalógusba. A továbbiakban ebben a katalógusban dolgozzunk, hacsak a feladat szövege nem kér egyebet.

A megoldásokat a feladatok után üresen hagyott helyre kell bemásolni a parancsok kimenetével együtt.

Ahol a feladat szövegében "felhasználónév" szerepel azt mindig a **saját felhasználónévvel** (azonosító) kell helyettesíteni.

Ha egy parancs kimenete több, mint 10 sor csak az első 10 sort másoljuk.

**FONTOS!** Csak a fordított aposztrofok közötti üres sorokba írjunk, minden más maradjon változatlanul!

Amennyiben saját gépen és a laborbeli rendszeren is dolgozunk a dokumentumban két prompt is szerepelhet.

Például:

- Írjuk ki a konzolra, hogy "1. laborfeladat" (laborban megoldva)

```
[robu@linux ~]$ echo "1. laborfeladat"
1. laborfeladat
```

- Írjuk ki a mai dátumot "év.hónap.nap" formátumban (saját gépen megoldva)

```
[judy@sajatgep laborfeladatok]$ date +%Y.%m.%d.
2023.02.23.
```

ls

- ellenőrizzük, hogy az aktuális katalógus üres-e
-

- listázzuk ki a */etc/* katalógus tartalmát

- listázzuk ki a gyökér katalógus tartalmát részletes információkkal (hosszú formátum)

`cd`

- lépünk be a gyökér katalógusban levő *var/log* katalógusba

- lépünk innen vissza az alapkatalógusunkba

- lépünk az aktuális katalógus szülőkatalógusába

- lépünk vissza az előző katalógusba úgy, hogy nem tudjuk melyik katalógusban voltunk előzőleg

`pwd`

- írassuk ki az aktuális katalógus nevét

tree

- írassuk ki az alaplatalógusunk szerkezetét, az állományokat is megjelenítve

- írassuk ki az alaplatalógusunk szerkezetét, csak a katalógusokat megjelenítve

mkdir

- lépünk be az alapkatalógusunkban levő *linux/labor1* katalógusba

- hozzunk létre az aktuális katalógusban egy *dir1* nevű katalógust, majd a *cd* parancs segítségével lépünk be ebbe a katalógusba.

- az aktuális katalógusban maradva hozzunk létre a */tmp* katalógusban egy katalógust a saját felhasználónevünkkel

- hozzuk létre az aktuális katalógusban egy *pldir1* nevű katalógust, abban legyen egy *pldir2*, ez utóbbiban pedig egy *pldir3* katalógus (egyetlen utasítás segítségével)

- hozzuk létre egyetlen utasítással az aktuális katalógusból kiindulva az alábbi katalógusszerkezetet:

```
      d1    d2
     /  \   |
    d11 d12 d21
     |       |
    d111    d211
```

- a *tree* parancs segítségével írassuk ki az aktuális katalógus tartalmát

## ls (folytatás)

- az aktuális katalógusban maradva listázzuk ki a *d1* katalógus tartalmát rekurzívan úgy, hogy a rejtett állományok is megjelenjenek

- listázzuk ki a szülőkatalógus tartalmát hosszú formátumban, fordított abc sorrendben úgy, hogy az állományok mérete felhasználóbarát módon jelenjen meg (vagyis blokkok helyett byte, Kbyte vagy Mbyte-ban számolva)

- az aktuális katalógusban maradva listázzuk ki az */usr/* katalógus tartalmát hosszú formátumban, méret szerint csökkenő sorrendben

- az aktuális katalógusban maradva listázzuk ki az alapkatalógusunk tartalmát hosszú formátumban, rekurzívan, az utolsó módosítás dátuma (ctime) szerinti sorrendben

- a *tree* parancs segítségével írassuk ki az aktuális katalógus tartalmát

## rm -r

- (a fent megadott katalógusszerkezet *d1* katalógusának szülőkatalógusában, azaz a *dir1* katalógusban vagyunk). Töröljük a *d111* alkatalógust

- töröljük egyetlen utasítással a *d2* katalógust és alkatalógusait

- a *tree* parancs segítségével írassuk ki az aktuális katalógus tartalmát

## cat

Az eddig létrehozott katalógusszerkezet *dir1* katalógusában vagyunk (ez az aktuális katalógus)

- hozzunk létre egy *textfile.txt* nevű állományt a *cat* parancs segítségével (legalább két sort tartalmazzon)

- listázzuk ki a *textfile.txt* nevű állomány tartalmát

- hozzuk létre a *text1* és *text2* állományokat (két *cat* parancs, mindkettő legalább 2 sort tartalmazzon)

- fűzzük össze a *text1* és *text2* állomány tartalmát, eredmény a *text3* állományban

- listázzuk ki (fűzzük össze) az aktuális katalógusban levő *text* szóval kezdődő állományok tartalmát

- listázzuk ki a *textfile.txt* nevű állomány tartalmát úgy, hogy a sorai sorszámozva legyenek

- az aktuális katalógusban maradva hozzunk létre a *cat* parancs segítségével egy-egy állományt a fent megadott katalógusszerkezet *d1*, *d11* illetve *d12* katalógusában (3 *cat* parancs, az állományok neve legyen rendre *f1*, *f11* illetve *f12*, mindegyik tartalmazzon legalább 2 sort)

---

more (vagy less)

- listázzuk ki az */etc/pseudopasswd* nevű állomány tartalmát oldalanként

- listázzuk ki a *munka* katalógus *diakok* alkatalógusban levő *diakok* állomány tartalmát oldalanként

cp

Az eddig létrehozott katalógusszerkezet *dir1* katalógusában vagyunk (ez az aktuális katalógus)

- másoljuk át az aktuális katalógus *textfile.txt* nevű állományát a *d1* katalógus *d11* alkatalógusába

- másoljuk át a *d1* katalógus tartalmát rekurzívan az aktuális katalógus *d1\_copy* katalógusába

- ismételjük meg az előbbi átmásolást úgy, hogy a program kérdezze meg a felhasználót minden egyes alkalommal, amikor egy már létező állományt akar felülírni (vigyázzunk, nem a *d1* katalógust akarjuk bemásolni a *d1\_copy*-ba, hanem csak ennek tartalmát, olvassuk el figyelmesen a *cp* man-t)

- másoljuk át a *munka/hazik\_felhasznalonev/c\_prog* katalógusban levő C programokat az aktuális katalógusba (a C programok neve ".c"-ben végződik)

mv

Az eddig létrehozott katalógusszerkezet *dir1* katalógusában vagyunk (ez az aktuális katalógus)

- a *tree* parancs segítségével írassuk ki az aktuális katalógus tartalmát

- nevezzük át az aktuális katalógusban levő *text2* állományt *pr-re*.

- helyezzük át a *d1* katalógus *f1* állományát az aktuális katalógusba

- helyezzük át a *d1/d12* katalógus *f12* állományát a *d1\_copy/d12* katalógusba *f12\_backup* néven

- a *tree* parancs segítségével írassuk ki az aktuális katalógus tartalmát

rm



Az eddig létrehozott katalógusszerkezet *dir1* katalógusában vagyunk (ez az aktuális katalógus)

- töröljük az aktuális katalógusban levő *text1* állományt

- töröljük az aktuális katalógus *d1* alkatalógusának *d11* alkatalógusában levő *textfile.txt* állományt

- töröljük az aktuális katalógusban levő *pldir1* katalógust (a tartalmával együtt)

- töröljük az aktuális katalógusban levő *d1* katalógus tartalmát rekurzívan (csak a tartalmát, a katalógus megmarad!), úgy, hogy a program minden egyes művelet előtt kérje a felhasználó jóváhagyását

- a *tree* parancs segítségével írassuk ki az aktuális katalógus tartalmát

W

### A laborbeli szerveren megoldandó feladat!

- listázzuk ki, hogy kik vannak bejelentkezve a rendszerbe és ki mit csinál

- listázzuk ki a bejelentkezett felhasználókat úgy, hogy a fejléc ne jelenjen meg

- az előbbi kimenet alapján állapítsuk meg, hogy ki "lazsál" a legtöbb ideje (a kiírt eredmények alapján, nem kell utasítás hozzá)

who

### A laborbeli szerveren megoldandó feladat!

- listázzuk ki, hogy kik vannak bejelentkezve

- listázzuk ki a rendszerben dolgozó felhasználók azonosítóját és hogy hányan vannak bejelentkezve

- írassuk ki csak a saját felhasználónevünket

pinky

### A laborbeli szerveren megoldandó feladat!

- írjunk ki információt a jelenleg bejelentkezett felhasználókról

- írjuk ki a saját felhasználóra vonatkozó információkat

- írjuk ki egy barátunkra vonatkozó információkat hosszú formátumban

last

### A laborbeli szerveren megoldandó feladat!

- listázzuk a legutóbbi belépéseket (a *last* parancs a *wtmp* logfájl adatai alapján dolgozik)

- listázzuk a saját felhasználó legutóbbi belépéseit

- listázzuk egy barátunk legutóbbi 3 belépését

- listázzuk egy vizsgafelhasználó (pl. *ex\_306\_14*) bejelentkezéseit úgy, hogy jelenjen meg a teljes felhasználónév

id

### A laborbeli szerveren megoldandó feladat!

- listázzuk ki saját felhasználó és csoportazonosítóinkat

- listázzuk ki egy barátunk felhasználó és csoportazonosítóit

- listázzuk ki egy tanár felhasználó és csoportazonosítóit

## date

- listázzuk ki a mai dátumot, órát, napot, stb. alapértelmezett formátumban

- írassuk ki csak a dátumot yy/mm/dd alakban

- írassuk ki, hogy az év hányadik napja van ma

- írassuk ki a hét napját **Ma** **<nap neve>** **van.** formátumban

- írassuk ki az órát, percet (hh:mm) és az időzónát tab-al elválasztva

sort

Lépünk át a szülőkatalógusba (*labor1*). Hozzuk létre a *pontok* állományt legalább 10 sorral. Sorainak formátuma: vezetéknév keresztnév pontszám, ahol a pontszám egy 0 és 200 közötti egész szám. Legyenek ismétlődő sorok is (nem közvetlenül egymás után).

- rendezzük a *pontok* állomány sorait abc sorrendbe (eredmény a képernyőn)
- 

- rendezzük a *pontok* állomány sorait fordított abc sorrendbe. Az eredmény az *eredmeny* állományba kerüljön
- 

- rendezzük a *pontok* állomány sorait a pontszámok csökkenő sorrendjébe (eredmény a képernyőn)
- 

- ellenőrizzük, hogy a *pontok* állomány sorai abc sorrendben vannak-e (eredmény a képernyőn)
- 

- rendezzük a *pontok* állomány sorait abc sorrendbe úgy, hogy az ismétlődő sorok csak egyszer szerepeljenek (eredmény a képernyőn)
-

Hozzunk létre egy legalább 10 soros *szamok* állományt, melynek sorai ":"-al elválasztott többjegyű számokat tartalmaznak (soronként legalább három szám).

- rendezzük a *szamok* állományt a második mező számai szerint növekvő sorrendbe (eredmény a képernyőn)

- a *munka* katalógus *diakok* alkatalógusban levő *kirandul* és *koncert* két rendezett állomány. Fésüljük össze a két állomány tartalmát (eredmény a képernyőn)

comm

Lépjünk be a *munka* katalógus *diakok* alkatalógusába

- hasonlítsuk össze a *kirandul* és *elsoev* rendezett (!) állományok tartalmát soronként

- írassuk ki azokat az elsőéveseket, akik nem kirándulnak

- írassuk ki a koncerten levő elsőéveseket

- írassuk ki a koncerten levők közül azokat, akik nem elsőévesek

- írassuk ki a koncerten levő kirándulókat

diff

A *munka* katalógus *diakok* alkatalógusában vagyunk (aktuális katalógus)

- másoljuk át az *kirandul* állományt az aktuális katalógus szülőkatalógusába, majd hasonlítsuk össze az aktuális katalógusban levő *kirandul* és a szülőkatalógusban levő *kirandul* állományok tartalmát.

- állapítsuk meg a *kirandul* illetve *elsoev* állományokról, hogy azonosak-e

- hasonlítsuk össze a *munka* katalógus *hazik\_felhasznalonev/oprendszerek* alkatalógusban levő *program1* és *program2* állományok tartalmát és röviden értelmezzük az eredményt.

head

- másoljuk át az *kirandul* és *elsoev* állományokat a *labor1* katalógusba, majd lépünk vissza a *labor1* katalógusba.

- listázzuk ki a *pontok* állomány első 5 sorát

- listázzuk ki a *pontok* állomány első 20 karakterét

- listázzuk ki a *pontok* állomány sorait az utolsó 2 sor kivételével

tail

- listázzuk ki az *kirandul* és *elsoev* állományok utolsó 3 sorát

- listázzuk ki az */etc/pseudopasswd* állomány utolsó 5 sorát

listázzuk ki a *munka/diakok* alkatalógusban levő *kirandul* állomány utolsó 20 byte-ját

cut

- írassuk ki a *pontok* állomány minden sorának 5. és 6. karakterét

- hozzunk létre egy *raktar* állományt, amely soraiban "TAB" karakterrel elválasztva tartalmazza a raktáron levő termékek nevét, mértékegységét, a raktáron levő mennyiséget, illetve esetenként a megrendelt mennyiséget (legalább 10 sor)



- Írassuk ki a *raktar* állomány minden sorból a termék nevét és a raktáron levő mennyiséget, vagyis az 1. és 3. mezőt

- írassuk ki a *kirandul* állományból a kirándulók nevét (5. mező, mezőelválasztó a ":" karakter)

uniq

A *labor1* katalógusban vagyunk (aktuális katalógus)

- az *eredmény* állományból (a *sort* parancsnál hoztuk létre!) töröljük ki az egymásután ismétlődő sorokat (eredmény a képernyőn)

- az *eredmény* állományból csak az egymásután többször ismétlődő sorokat írassuk ki (eredmény a képernyőn)

- az *eredmény* állományból írassuk ki minden sor elé, hogy egymás után hányszor szerepel (eredmény a képernyőn)

- írassuk ki a *pontok* állomány sorainak, szavainak, illetve karaktereinek számát

- írassuk ki a *kirándul* állomány sorainak számát

- írassuk ki a *pontok* állomány leghosszabb sorában szereplő karakterek számát

- írassuk ki a *pontok* állomány szavainak számát

- írassuk ki az alapkatalógusunkban levő *.bash\_history* állomány karaktereinek számát

In

- hozzunk létre hard link-et a *munka/hazik\_felhasznalonev/oprendszer* katalógusban levő *program1* és *program2* állományokra a *dir1* katalógusban, úgy, hogy a link neve kezdődjön a felhasználónévvel: pl *rjim0001\_program1*

- készítsünk szimbolikus linket, melynek neve *felhasznalok* és az */etc/pseudopasswd* állományra mutat

## chmod

Lépünk be a *munka/hazik\_felhasznalonev/oprendszerek* katalógusba. Minden parancs után hajtsuk végre az *ls -l program\** parancsot is.

- adjunk végrehajtásjogot a tulajdonosnak a *program1* állományra

- vegyük el az olvasásjogot a csoport tagjaitól és a többi felhasználótól a *program1* állományra

- a csoport tagjainak jogait változtassuk meg úgy, hogy olvasás- és írásjoguk legyen a *program2* állományra

- állítsuk be a *program1* állomány jogait "rwxr-xrw-" re oktális számként adva meg a jogokat

- lépünk a *labor1/dir1* katalógusba, majd adjunk végrehajtási jogot a *d1\_copy* katalógusra és a benne levő állományokra rekurzívan, a csoportjában levő felhasználóknak

## find

Lépünk be a *labor1* katalógusba

- keressük meg rekurzívan a *munka* katalógus összes állományát

- keressük meg az aktuális katalógusban és alkatalógusaiban levő összes közöségi állományt

- keressük meg a `/tmp` katalógusban és alkatalógusaiban szereplő azon állományokat, melyeknek csoport azonosítója ugyanaz, mint a mienk

- listázzuk ki rekurzívan az aktuális katalógus összes olyan állományát, melyre mindenkinek van minden (írás/olvasás/végrehajtás) joga

- listázzuk ki az alapkatalógusunkban és ennek alkatalógusaiban szereplő azon állományokat, melyeknek mérete 100kbyte-nál nagyobb

- listázzuk ki rekurzívan az aktuális katalógus összes állományát, amelyiknek a neve `.txt`-vel végződik

- listázzuk ki rekurzívan a *munka* katalógus összes olyan állományát, melyek kevesebb mint 10 napja voltak utoljára módosítva

- listázzuk ki az aktuális katalógus üres állományait. A keresés max. 2 szint mélységig történjen.

- listázzuk ki rekurzívan az aktuális katalógus összes 10 kbyte-nál kisebb közönséges állományának méretét (byte-okban) és teljes nevét `<Tab>`-bal elválasztva

## file

- írassuk ki a *labor1/dir1* katalógusban levő *textfile.txt* állomány típusát

- írassuk ki a */bin/ls* állomány típusát

- írassuk ki a *munka/kepek* katalógusban levő *BBTE.jpg* képfájl típusát

- írassuk ki a *munka* katalógusban levő *meglepetes.gif* képfájl típusát

- írassuk ki egy általunk létrehozott szimbolikus link típusát rövid formátumban (az állomány neve nélkül)

- írassuk ki a *munka/kepek* katalógus összes állományának típusát (egyetlen utasítás)

du

- írassuk ki az aktuális katalógus és alkatalógusai méretét

- írassuk ki a *munka* katalógus, és a benne levő állományok méretét, valamint az összméretet.

- írassuk ki a *munka/kepek* katalógusunkban szereplő állományok méretét különböző egységekben (human readable)

- írassuk ki a *munka* katalógusban szereplő állományok méretét byte-okban és ezek utolsó módosítási dátumát

df

- írassuk ki az állományrendszerek lemezhasználati adatait

- írassuk ki az állományrendszerek lemezhasználati adatait "human readable" formátumban

- írassuk ki az állományrendszerek i-nod táblára vonatkozó adatait valamint ezek típusát

- írassuk ki az összes állományrendszer lemezhasználati adatait Megabyte-ban

ps

- listázzuk az aktív terminálhoz tartozó folyamatokat és ezek állapotát.

- listázzuk ki a rendszerben futó összes folyamatot (más felhasználókéét is) és ezek állapotát

- listázzuk a saját felhasználónkhoz tartozó aktív folyamatokat

- listázzuk a csoportunkhoz tartozó folyamatokat hosszú formátumban

kill

- listázzuk az összes kill paranccsal küldhető jel megnevezését és sorszámát

- 
- állítsuk le egy aktív folyamat működését PID alapján SIGTERM jelezés küldésével
-