



## Mérési Utasítás

## Alapvető Linux/UNIX parancsok

### A terminál.

A Linux és a UNIX, multi taszkos, több felhasználós rendszerek. A több feladat végrehajtásához egy (vagy akár több) felhasználó több terminálon (vagy virtuális terminálon távolról) jelentkezik be a rendszerre egyidejűleg, és hajt végre feladatokat. A terminálokat 1-től 6-ig az ALT>+<F1> ...<ALT>+<F6> billentyűkombinációkkal érhetjük el (ha grafikus felület is van a Linux/UNIX rendszeren, akkor a grafikus felületről a karakteresre CTRL>+<ALT>+<F1> ...<CTRL>+<ALT>+<F6> billentyű-kombinációkkal léphetünk). Alapértelmezetten 6 terminál fut.

### Feladat

1. Lépeghessen a terminálok között, gyakorolja a billentyűkombinációkat.

### A login prompt.

A felhasználónak a rendszerbe be kell jelentkeznie. Ez a login promptnál történik meg. Először a felhasználó nevet kell megadnunk, majd a felhasználó jelszavát a password promptnál. Nem kell meglepődni, a jelszó nem fog megjelenni, sem csillagok, sem egyéb karakterek formájában. Amennyiben megfelelő felhasználó nevet és jelszavat adunk meg, a rendszer beléptet, majd alapértelmezetten a saját home könyvtárunkba tesz (root esetén „/root”, diak esetén „/home/diak”).

### Feladat

1. Lépj be root felhasználóként az első terminálon, a root felhasználó jelszava: labor.  
(<CTRL>+<ALT>+<F1>, majd login promptnál root, jelszó promptnál labor)
2. Lépj be diak felhasználóként a második terminálon, a felhasználó jelszava: diak.  
(<CTRL>+<ALT>+<F2>, majd login promptnál diak, jelszó promptnál diak)
3. Váltson a grafikus felületre, majd ott fogja folytatni a munkát. (<CTRL>+<ALT>+<F7>)

### cd, pwd, ls

A Linux/Unix rendszereknél a könyvtárak gyökere – eltérően a DOS, Windows rendszerektől – nem egy betűjel (C:\, D:\, stb.), hanem a gyökér „/” könyvtár. A könyvtárak közötti mozgásra a cd (change directory) parancsot alkalmazzuk. Ha kapcsolók nélkül adjuk ki, akkor a saját home könyvtárunkba lépünk, attól függetlenül, hol álltunk a könyvtárrendszerben.



Ha egy „-” (cd -) jel áll mögötte, akkor abba a könyvtárba lép, amelyikben előzőleg voltunk (ha volt ilyen). A cd után állhat „..” is, mely a könyvtárszerkezetben feljebb léptet bennünket (természetesen a „/” gyökérben kiadva nem tudunk feljebb lépni ezért a „/” gyökérbe jutunk).

A cd után megadhatunk abszolút, vagy relatív útvonalakat, ahova be szeretnénk lépni. Az abszolút útvonal minden esetben a gyökér könyvtárral kezdődik (például: cd /usr/lib), míg relatív esetén ahhoz a könyvtárhoz viszonyítunk, ahol állunk (a fenti példát folytatva: feltételezve, hogy most a /usr/lib könyvtárban állunk: cd ../../lib/modules; ezzel a /lib/modules könyvtárba lépünk).

A pwd parancs segít meghatározni, hol vagyunk, ha esetleg nagyon eltévedtünk volna.

Az ls parancs a DOS/Windows dir parancsának felel meg. Minden kapcsoló nélkül kiadva a könyvtáron belüli könyvtárbejegyzéseket listázza ki (például: fájl, könyvtár, karakteres vagy blokkos eszköz neve, stb.), amennyiben kifer, akkor több oszlopban. (Unix/Linux alatt a fájloknak nincs kiterjesztése, csak végződése, és az operációs rendszer nem is értelmezi azt, de a programok esetleg igen. Például a gcc C fordító elvárja, hogy a fájlnevek .c vagy .cc végződése legyen.

Az ls kapcsolói:

- -l: részletes listázás, melynek segítségével a fájlról több információt tudhatunk meg.
- -1: akkor van rá szükségünk, ha csak egy oszlopot szeretnénk listázni
- -a: a rejtett fájlokat (az olyan fájlokat melyek nevei „.”-tal kezdődnek) is kilistázza

Az ls és kapcsolói után megadhatunk pl. könyvtárakat, fájlokat melyeket szeretnénk listázni (például: ls -l /bin/bash).

## ***Feladat***

1. Állapítsa meg, hogy login után melyik könyvtárban áll! (pwd)
2. Lépjen be a /bin könyvtárba, és listázza annak tartalmát! (cd /bin)
3. Lépjen be a saját home könyvtárába! (cd)
4. Listázzon ki minden fájlt a könyvtárban (a rejtett állományokat is)! (ls -a)
5. Lépjen vissza a /bin könyvtárba! (cd -)
6. Az ls segítségével írasson ki információkat a /bin/bash -ról (ls -l bash)
7. Lépjen a „/” könyvtárba! (cd /)
8. Listázza ki a „/” tartalmát! (ls)
9. Lépjen be a /tmp könyvtárba (cd /tmp)



## mkdir, touch, rmdir, rm, cp, mv

Az mkdir könyvtárak, a touch fájlok létrehozására szolgál (ha már létezik a létrehozandó fájl, és úgy adjuk ki a touch parancsot, akkor a módosítás idejét változtatja meg csak!). Az mkdir a -p kapcsolóval, egész könyvtárszerkezetek létrehozására képes.

Az rmdir könyvtárak, az rm fájlok (megfelelő kapcsolók segítségével pedig minden fájl, könyvtár) eltávolítására használható.

A cp fájlok, könyvtárak másolására szolgál, -r kapcsoló hatására egész könyvtárszerkezeteket másol át. Az mv fájlok, könyvtárak mozgására, átnevezésére szolgál.

Több fájl, könyvtár egyidejű létrehozásának egy nagyszerű módja, a {} használata. Ezt a parancsértelmező „kifejti”. Ha több szó áll a {} között egymástól vesszővel (és csak azzal, szóköz nem állhat!), akkor ahány szó van a {} között annyi argumentumra hajtja végre a parancsot. A későbbiekben példát fogunk rá látni. Itt kell még megemlítenünk, az ún. joker karaktereket. A „?” egy fájl, vagy könyvtár nevében 1 db a „?” helyén álló bármilyen karaktert helyettesít. A „\*” egy fájl vagy könyvtár nevében tetszőleges számú (akár nulla is) db, a „\*” helyén álló bármilyen karaktert/karaktereket helyettesít. A könnyebb megértés kedvéért néhány a későbbiekben használatos példa:

**masodik\_szint?:** az összes olyan nevű fájlt vagy könyvtárat helyettesíti, melynek az utolsó karaktere bármi lehet: masodik\_szint1, masodik\_szints, masodik\_szint4

**file\*:** az összes olyan nevű fájlt vagy könyvtárat helyettesíti, mely első négy karaktere fájl és utána bármi állhat: file1, file123, filesahjdgfa, fajlstb.txt

### Feladat

1. Győződjön meg, hogy a /tmp könyvtárban áll! (**pwd**)
2. Amennyiben nem, lépjen be a cd paranccsal! (**cd /tmp**)
3. Hozzon létre egy proba nevű könyvtárat! (Ne használjon ékezetet fájlok, könyvtárak és felhasználó nevekénél, ez egyébként minden unix rendszer alatt ellenjavallt!) (**mkdir proba**)
4. Próbálja létrehozni, a proba könyvtáron belül egy

elso\_szint/masodik\_szint1;

elso\_szint/masodik\_szint2 könyvtár struktúrát!

(**mkdir proba/elso\_szint/{masodik\_szint1,masodik\_szint2}**)

**Ha mindent jól csinált, hibaüzeneteket kellett, hogy kapjon.** Ennek oka, hogy megpróbáltuk létrehozni, a masodik\_szint1, masodik\_szint2 könyvtárakat egy olyan könyvtárban, mely nem létezik. Annak ellenére, hogy egy parancsot adtunk ki, kettő



hibaüzenetet kaptunk. Ez jól szemlélteti a {} között álló szavak kifejtését, hiszen a parancsunk két argumentummal került végrehajtásra.

5. A helyes megoldás:

```
(mkdir -p proba/elso_szint/{masodik_szint1,masodik_szint2} )
```

6. Itt már nem kapunk hibaüzenetet, de győződjünk meg róla, hogy valóban létrejött-e a könyvtárszerkezet! (ls proba; ls proba/elso\_szint)

7. Hozzon létre 3-3 fájlt, a masodik\_szint1, masodik\_szint2 könyvtárakban, használja a {}-t!

```
(touch proba/elso_szint/{masodik_szint1/{file1,file2,file3},masodik_szint2/{file1,file2,file3}})
```

8. Ellenőrizze, hogy létrejöttek-e a fájlok!

```
(ls proba/elso_szint/masodik_szint1
```

```
ls proba/elso_szint/masodik_szint2)
```

9. A touch segítségével változtassa meg a proba/elso\_szint/masodik\_szint2/file2 módosítási idejét!

```
(touch proba/elso_szint/masodik_szint2/file2)
```

10. Ellenőrizze ls segítségével, valóban történt-e változás!

```
(ls -l proba/elso_szint/masodik_szint2)
```

11. Próbálja meg átmásolni a proba könyvtárat cp segítségével proba2 névre!

```
(cp proba proba2)
```

12. Mivel a könyvtár nem üres a cp nem hajta végre a parancsot, ezért rekurzívan kell másolnunk.

```
(cp -r proba proba2)
```

13. Próbálja meg törölni az rmdir segítségével a proba/elso\_szint/masodik\_szint1 könyvtárat!

```
(rmdir proba/elso_szint/masodik_szint1)
```

Ha helyesen dolgozott eddig, akkor az rmdir nem törli a könyvtárat, mert az nem üres.

14. Törölje a fájlokat a könyvtárból a „\*” joker karakter segítségével (rm proba/elso\_szint/masodik\_szint1/\*), majd törölje a könyvtárat az rmdir paranccsal!

```
(rmdir proba/elso_szint/masodik_szint1)
```

15. A fent átmásolt proba2/elso\_szint könyvtárból mozgassa át a masodik\_szint1 könyvtárat a proba/elso\_szint könyvtárba!



(mv proba2/elso\_szint/masodik\_szint1 proba/elso\_szint)

16. Az rmdir használatával sokáig tartana, ezért az rm parancs segítségével töröljük a proba, proba2 könyvtárakat. Ehhez az rm -r (rekurzív) és -f (force, mindenképp végrehajtja) kapcsolóit használjuk! (rm -rf proba\*)

[ Az rm -rf / a gyökér könyvtárat törli megfelelő jogosultságok (root jog) esetén.  
Ezért az rm -rf parancsot ésszel használjuk!!! ]

## Kilépés

Indítsa újra a számítógépet a **reboot** parancs kiadásával, vagy a K menu újraindítás gomb segítségével.