

Előző gyakorlat - Fel - Következő gyakorlat

## Tartalomjegyzék

- 1 Linux bevezetés
  - ◆ 1.1  
Alapvető  
programok
  - ◆ 1.2  
Gyorsbillentyűk
- 2 Bash
  - ◆ 2.1  
Alapvető  
parancsok
  - ◆ 2.2 Pipe,  
redirect
  - ◆ 2.3  
Feladatok
    - ◇ 2.3.1  
Megoldás
- 3 Windows
  - ◆ 3.1  
Különbségek

## Linux bevezetés

### Alapvető programok

- terminál: Applications menu -> System Tools -> Konsole
- Midnight Commander mc
- top
- szövegszerkesztők:
  - ◆ gedit
  - ◆ mcedit, nano,
  - ◆ mindentudók: emacs, vim (harc a szövegszerkesztők között)
- beépített programok listázása: help
- segíts magadon:
  - ◆ info <parancs>
  - ◆ man <parancs> (ez a régebbi UNIX)
  - ◆ <parancs> --help

### Gyorsbillentyűk

- Vágólap
  - ◆ egérrel kijelölés, jobb-klikk
  - ◆ Ctrl+Insert, Shift+Insert
  - ◆ alkalmazásonként változhat (lehet Ctrl+C, Ctrl+V is)
- navigálás
  - ◆ TAB: kiegészítés
  - ◆ HOME: sor elejére, END: sor végére
- terminál
  - ◆ fel, le nyilak: előzmények

- ◆ Ctrl+R keresés előzményekben

## Bash

Néhány hasznos link:

- <https://www.codecademy.com/learn/learn-the-command-line>
- <http://ss64.com/bash/>

## Alapvető parancsok

- **pwd**: aktuális könyvtár kiírása.
- **cd**: könyvtárváltás. Ha paraméter nélkül hívjuk meg, akkor a saját mappánkba visz. Pl **cd /mnt, cd, cd ..**
- **ls**: könyvtár tartalmának listázása. Pl: **ls, ls -h, ls /dev, ls -h -l ~**
- **mkdir**: könyvtár létrehozása. **mkdir NewFolder**
- **cp**: másolás. **cp mit.txt hova.txt**
- **mv**: mozgatás/átnevezés (az eredeti nem marad meg). **mv mit.txt hova.txt**
- **rm**: törlés, Pl: **rm alma.txt**, rekurzívan töröl minent a mappával együtt: **rm -r NewFolder**
- **quota**: A felhasznált/megmaradt helyet írja ki. Ha a kvótánk betelik, akkor nem kapjuk meg a nekünk címzett leveleket és a grafikus terminálra sem tudunk belépni; ilyenkor a parancssoros terminálra belépve tudjuk kitakarítani a home-unkat.
- **df, du**: lemezen szabad / felhasznált helyet írja ki. A **-h** kapcsolóval olvasható *GB, MB, kB* értékeket kapunk, de lehet fix byte-okban is kiírni. Pl: **df -h ~**
- **mc**: midnight commander fájlböngésző.
- segíts magadon
  - ◆ **info <parancs>**
  - ◆ **man <parancs>** az adott parancsról leírást ad. A leírásban navigálni a 'fel', illetve a 'le' billentyűkkel, továbbá a 'PageUp', 'PageDown'-nal lehet. Kereséshez '/' jel után kell beírni a kifejezést. Kilépés 'q'-val. **man ls**
  - ◆ **<parancs> --help**
  - ◆ **help**
- **history**: parancs történet lekérdezése.
- **apropos**: parancs/program nevek keresése. *apropos math*
- **top**: futó alkalmazások.
- **kill, killall, xkill**: folyamatok leállítása. Egy adott felhasználó összes folyamatának leállítása: **killall -u username**
- **cat**: fájl tartalmának kiírása. Pl: **cat .forward**
- **echo**: kiírja az argumentumait, Pl: **echo Hello World**. Figyelem, beugratós: **echo --help**
- **grep**: szöveg mintázatok keresése állományokban. **grep "http://wiki.math.bme.hu" http://wiki.math.bme.hu szénakazal.txt**
- **less**: szöveges állományok megjelenítése, könnyebb olvashatóság nagy fájlknál, Pl **less valami.txt**
- **head (-n), tail (-n)**: fájl elejének és végének a kiírása. **head -n 100 valami.txt**
- **sort (-g -k)**: szövegállomány rendezése.
- **wc (-l)**: byte/karakterek/sorok számolás adott fájlra.
- **ssh**: távoli gépre való bejelentkezést teszi lehetővé. [Linux-ról Windows-ról](#)
- **scp**: távoli gépre/gépről való másolás. [Linux-ról Windows-ról](#)
- **wget**: file-ok letöltése internetről, HTTP, HTTPS és FTP használatával.
- **exit**: kilépés terminálból (vagy ctrl+d).
- **sudo**: programok futtatása superuserként. (Mat intézetben erre nem lesz lehetőségünk) Pl. **sudo rm /home/someone\_I\_hate/important\_file.txt**
- **shutdown, reboot**: kikapcsol, újraindít (Mat intézetben erre sem lesz lehetőségünk)

## Pipe, redirect

Minden program írhat a konzolra, vagy az `stderr`-re, ez az, amit a konzolon szoktunk látni. Azonban ezeket a képernyő helyett fájlba is írhatjuk.

```
$ ls -l ~
drwxr-xr-x 8 borbely student 4096 Aug 30 23:24 Desktop
drwxr-xr-x 2 borbely student 4096 Mar 27 2012 Downloads
drwxr-xr-x 2 borbely student 4096 Oct 20 2009 Drives
drwx----- 2 borbely student 4096 Apr 20 10:42 mail
drwxr-xr-x 7 borbely student 4096 Sep 6 13:01 public_html
$ ls -l ~ > folder.txt
$ cat folders.txt
drwxr-xr-x 8 borbely student 4096 Aug 30 23:24 Desktop
drwxr-xr-x 2 borbely student 4096 Mar 27 2012 Downloads
drwxr-xr-x 2 borbely student 4096 Oct 20 2009 Drives
drwx----- 2 borbely student 4096 Apr 20 10:42 mail
drwxr-xr-x 7 borbely student 4096 Sep 6 13:01 public_html
$ _
```

A `>` jellel egy parancs kimenetét egy fájlba irányítjuk át, majd a fájlt kiírathatjuk a `cat`-el. Ugyanakkor figyeljük meg a következőt:

```
$ ls -l /home/algebra/wettl/ > myfile.txt
ls: cannot open directory /home/algebra/wettl/: Permission denied
$ cat myfile.txt
$ _
```

A **myfile.txt** üresen maradt, de a konzolon hibaüzenet jelent meg (nem volt jogunk a mappa listázására). Ez azért van, mert az utóbbi `ls` parancs nem a standard kimenetre, hanem a hiba-kimenetre, azaz **stderr**-re írt. Az **stderr**-t a `2>` jellel irányíthatjuk át:

```
$ ls -l /home/algebra/wettl/ > myfile.txt 2> error.log
$ cat myfile.txt
$ cat error.log
ls: cannot open directory /home/algebra/wettl/: Permission denied
$ _
```

Így az egyik fájlba a standard kimenet (**stdout**), a másikban az **stderr** került bele.

Az egyes programok olvashatnak a standard bemenetükről: **stdin**. Az ú.n. nem-interaktív parancsok olyanok, hogy a parancs kiadása után a befejezéséig nem nyúlunk a billentyűzethez. Ezekkel csak az argumentumaikon keresztül kommunikálhatunk. Interaktív parancsok a futásuk közben olvasnak a standard bemenetükről: **stdin**

- nem-interaktív parancs: `grep`  
`"http://wiki.math.bme.huneele"http://wiki.math.bme.hu haystack.txt`
- interaktív parancs: `grep`  
`"http://wiki.math.bme.huneele"http://wiki.math.bme.hu`

Az előbbi esetben a **grep** egy megadott fájlban keresi a *needle* kifejezést. Az utóbbi esetben a beírt szövegben keresi ugyanazt.

```
$ grep "needle"
I hate this
```

```
why can't I find anything
why can't I find a needle
why can't I find a needle
oh, a needle
oh, a needle
again
$ _
```

A Ctrl+D billentyű-kombinációval megszakítjuk a futást. A *needle*-t tartalmazó sorok duplán vannak, mert azokat a **grep** kérésre kiírta.

Próbáljuk ki így:

```
$ grep "needle" > needle.txt
I hate this
why can't I find anything
why can't I find a needle
oh, a needle
again
$ cat needle.txt
why can't I find a needle
oh, a needle
$ _
```

Ezzel kiválasztottuk a bemenetről a *needle*-t tartalmazó sorokat.

Az egyik parancs kimenete lehet a másik bemenete is, ez a **pipe**, jele egy függőleges vonal: |.

Példa: írjuk ki a home mappánk azon elemeit, melyek nevükben az *info* szövegrészletet tartalmazzák.

```
$ ls -l ~
Desktop
Downloads
Drives
info1
info_hf.txt
mail
myfile.txt
myfolder
needle.txt
public_html
regi_info
regi_info_zh.txt
$ ls -l ~ | grep "http://wiki.math.bme.huinfo"http://wiki.math.bme.hu
info1
info_hf.txt
regi_info
regi_info_zh.txt
$ _
```

Példa: sok elem van egy mappában és apránként végig akarjuk nézni:

```
$ ls /home/student/ | wc -l
821
$ ls /home/student/ | less
```

Egy fájlból kivehetjük a különböző sorokat (rendezés és duplikált sorok törlése):

```
$ cat nevek.txt | sort | uniq > nevek.sorted.txt
```

Ha egy fájl tartalmát akarjuk pipe-olni egy parancsba, azt így is megtehetjük (<):

```
$ cat nevek.txt | sort
$ sort < nevek.txt
```

Ez kombinálható > -el.

```
$ sort < nevek.txt | uniq > nevek.sorted.txt
```

## Feladatok

1)

- Töltsük le a `Alnevsor.txt` állományt a borbely felhasználó oldaláról
  - ◆ `wget`
- Távolítsuk el a neveket az els? oszlopból.
  - ◆ `cut`
- Rendezzük a sorokat Neptun kód szerint (a fejléc kivételével).
  - ◆ `sort`
- Nagybetűsítsük a Neptun kódokat.
  - ◆ `tr`
- Írjuk ki az eredményt egy külön fájlba (segédfájlokat használhatunk útközben).
  - ◆ `cat, echo, paste, head, tail`

## Megoldás

```
wget math.bme.hu/~borbely/Alnevsor.txt
tail -n+2 Alnevsor.txt | cut -f2,3 -d"http://wiki.math.bme.hu "http://wiki.math.bme.hu | sort | tr
```

2) Hány hallgatónak van felhasználója a lebniz-en?

Hány felhasználója van a az egyes tanszékeknek?

Az alábbi paranccsal meg lehet nézni hogy az adott mappa fájljainak mi a *jogosultsági* száma.

```
stat -c '%a %n' ./*
```

Melyik hallgatók mappájának a legmagasabb ez a száma?

## Windows

A DOS után a Windows-os parancssori lehet?ségek nem nagyon fejl?dtek. A parancssor közelében sincs a Linux-nak, sem használhatóságban, sem funkcióiban, de azért van valami.

- Parancssor indítása
  - ◆ Start menü -> Parancssor
  - ◆ Start menü -> Futtatás -> cmd
  - ◆ Keresés -> cmd
- leírás
- használható: `help, help <parancs>`
- Windows 10-en új életre kelt a cmd.
- A Windows 10 egy éves évfordulója óta beépített a bash.

Használatos a PowerShell, ami nem mindegyik verzióba beépített, de ingyenes és elterjedt.

## Különbségek

Pár szembevetés különbség

- a Windows case-insensitive, vagyis kis és nagybetű nem számít, ellenben a Linux case-sensitive.
- A könyvtár-, fájlnevek:
  - ♦ C:\Windows\System32\
  - ♦ /usr/bin/
- parancssori kapcsolók perrel versus kötőjellel:
  - ♦ Windows: dir /b
  - ♦ Unix: ls -l
- script
  - ♦ Windows-on: .bat
  - ♦ Linux-on: .sh
- pipe, redirect hasonló

Gyakran emulálják/helyettesítik a unix-os lehetőségeket windows-on:

- cygwin
- mingw

Előző gyakorlat - Fel - Következő gyakorlat