

Tartalomjegyzék

- 1 Bash
 - ◆ 1.1 Parancsok, programok
 - ◇ 1.1.1 Állományok
 - ◇ 1.1.2 Alkalmazások, folyamatok
 - ◇ 1.1.3 Szöveges állományok
 - ◇ 1.1.4 Távoli elérés
 - ◇ 1.1.5 Egyéb
 - ◆ 1.2 Feladatok
 - ◇ 1.2.1 Jogosultságok
 - ◇ 1.2.2 Gráfok
 - 1.2.2.1 Megjegyzés
 - ◇ 1.2.3 Script file
 - 1.2.3.1 Megjegyzés
 - ◆ 1.3 Hasznos mindennapi parancsok
- 2 Python
 - ◆ 2.1 Mi a python?
 - ◆ 2.2 Interpreter
 - ◆ 2.3 Script

Bash

Parancsok, programok

Állományok

- **pwd**: aktuális könyvtár kiírása. *pwd*
- **cd**: könyvtárváltás. Ha paraméter nélkül hívjuk meg, akkor a saját mappánkba visz. *cd /mnt, cd*
- **ls**: (-a, -h, -l): könyvtár tartalmának listázása. *ls, ls -h, ls /dev, ls -h -l /mnt*
- **mkdir**: könyvtár létrehozása. *mkdir BME*
- TAB: kiegészítés.
- **cp**: másolás. *cp mit.txt hova.txt*
- **mv**: mozgató/átnevezés (az eredeti nem marad meg). *mv mit.txt hova.txt*
- **rm (-r)**: törlés. *rm alma.txt, rm -r BME*

- **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 (-h)**: lemezen szabad / felhasznált helyet írja ki. *df -h ./* , *du -h /home*
- **mc**: midnight commander fájlböngésző?

Alkalmazások, folyamatok

- **man**: 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*
- **history**: parancs történet lekérdezése.
- **apropos**: parancs/program nevek keresése. *apropos vi*
- **ctrl-R**: keresés a history-ban.
- **top**: futó alkalmazások.
- **killall (-u), xkill**: processzek leállítása. *killall -u username*

Szöveges állományok

- **vim**: command line szövegszerkesztő?
- **emacs**: szövegszerkesztő, **-nw** kapcsolóval command line-os
- **cat**: fájl tartalmának kiírása. Ha nem adunk meg fájlnevet, akkor a billentyűzetrel vár bevitelt. Kilépés: Ctrl-D-vel. *cat valami.txt*
- **grep**: szöveg mintázatok keresése állományokban. *grep kifejezés file*
- **less**: szöveges állományok megjelenítése. *less valami.txt*
- **head (-n)** , **tail (-n)** : fájl elejének és végének a kiírása. *head -n 100 valami.txt*
- **>, <, |**: output átirányítása, "http://wiki.math.bme.hu/pipe" http://wiki.math.bme.hu-olás.
- **sort (-g -k)**: szövegállomány rendezése.
- **wc (-l)**: byte/karakterek/sorok számolás adott fájlra.
- **awk**: programozási nyelv szöveges állományok feldolgozására.

Távoli elérés

- **ssh**: távoli gépre való bejelentkezést teszi lehetővé. *felhasznalo@leibniz.math.bme.hu*
- **scp**: távoli gépre/gépről való másolás. *scp felhasznalo@leibniz.math.bme.hu/home/stu_bme/felhasznalo/dokumentum.txt ./dokumentum.txt*
- **wget**: file-ok letöltése HTTP, HTTPS és FTP használatával.

Egyéb

- **exit**: kilépés terminálból (vagy ctrl+d).
- **sudo**: programok futtatása superuserként.
- Egyéb hasznos fogalmak
 - ◆ Reguláris kifejezések
 - ◆ Felhasználó jogok, ezek megváltoztatása
 - ◆ Shell scriptek
 - ◆ Futtatható file-ok
 - ◆ ...

Feladatok

Jogosultságok

- Hozz létre egy könyvtárat "http://wiki.math.bme.huGYAK1" http://wiki.math.bme.hu néven a saját mappádban, és lépj be a mappába.

```
mkdir GYAK1
cd ./GYAK1
```

- Készíts egy file-t test néven amiben csak annyi van:
"http://wiki.math.bme.huinfo1" http://wiki.math.bme.hu.

```
echo info1 > test
```

- Változtasd meg a test file hozzáférési jogát úgy, hogy senki ne tehessen vele semmit.

```
chmod 000 test
```

- Próbáld meg letörölni. (Megkérdezi, hogy biztosan le akarod-e törölni, ha nem lenne rajta írás védelem, akkor azonnal letörölné.)

```
rm test
```

- Próbáld meg belenézni.

```
less test
```

- Adj vissza magadnak minden jogot és a csoportodnak olvasási jogot.

```
chmod 740 test
```

- Most nézz bele, majd töröld le.

```
cat test
rm test
```

Gráfok

- Töltsd le a wget programmal az alábbi (gráf) él listát: [1]

```
wget http://math.bme.hu/~kkovacs/info1/edge_list
```

- Készíts biztonsági másolatot a file-ról, majd ellenőrizd a másolatot.

```
cp edge_list edge_list_save
less edge_list_save
```

- Számold meg az élek számát a gráfban.

```
wc -l edge_list
```

- Írd ki a 3-as csúcsból induló éleket, majd irányítsd a kimenetet a
"http://wiki.math.bme.huedges_3" http://wiki.math.bme.hu file-ba

```
grep 3 edge_list
grep 3 edge_list > edges_3
```

Számold ki a 3-as csúcs fokszámát.

```
grep 3 edge_list | wc -l
```

Számold ki az összes csúcs fokszámát. (Ez elég bonyolult, nem kell tudni.)

```
awk 'BEGIN{ez_csak_pelda_beginre=0;}{DEG[$1]++;DEG[$2]++;}END{for (i in DEG) print i,DEG[i]}' edge_
less degrees
```

Határozd meg a legnagyobb és legkisebb fokú csúcst.

```
sort -g -k2,2 degrees > degrees_sorted
less degrees_sorted
head -n 1 degrees_sorted
tail -n 1 degrees_sorted
```

Megjegyzés

A legutolsó utasítás file létrehozása nélkül is megoldható:

```
sort -g -k2,2 degrees | head -n 1
sort -g -k2,2 degrees | tail -n 1
```

Script file

Készíts el els? linux scriptedet! A script kap egy csúcst és kiadja a fokszámát. Használj gedit szövegszerkesztőt a file létrehozásához:

```
gedit fokszam.sh
```

Ebbe másold be a következő sort.

```
csucs=$1
echo "a "$csucs". csucs fokszama: "
grep $csucs edge_list | wc -l
```

Futtasd a scriptet különböző csúcsokkal, példa:

```
sh fokszam.sh 3
```

Megjegyzés

- gedit indítható úgy is, hogy ne vegye el a kurzort a termináltól (ne egye meg a terminált):

```
gedit fokszam.sh &
```

Hasznos mindennapi parancsok

Hasznos parancsok:

- megkeresni egy file-t az egy adott könyvtárban

```
find ./ | grep kulcsszo
```

- adott mappában található file-ok tartalmában keresés

```
cat ./* | grep kulcsszo
```

Python

Mi a python?

- programozási nyelv
- magasszintű programozási nyelv (távol van a gépi kódtól)
- könnyen olvashatóságra törekszik így nagyon jó első programnyelvnek

Interpreter

- A *python* paranccsal indíthatunk interaktív python-t (console-ban):

```
username@leibniz:~$ python
```

- Innentől a python interpreterben vagyunk, szimpla aritmetikai műveleteket ki lehet adni:

```
>>> 5 + 7
>>> 10 - 5
>>> 7 * 6
>>> 10 / 3
```

- Az utolsó művelet nem azt adja amit először várnánk...miért? Próbáljuk így:

```
>>> 10.0 / 3.0
```

- Az osztás egész számokon egész osztást jelent, ha kiteszük a tizedespontot, akkor már *float* (lebegőpontos) számokkal végezzük a műveletet.
- Hatványozás:

```
>>> 2 ** 10
```

- Hatványozást szabad próbálni lebegőpontos számokkal is.
- A következő parancs lép ki az interpreterből:

```
>>> exit()
```

Script

- Python script alatt egy vagy több python kódot tartalmazó file-t értünk.
- Kiterjesztésük *.py*, hozzunk létre egy python script file-t:

```
gedit gyak2.py &
```

- Másoljuk be a következő kódot a file-ba:

```
#!/usr/bin/python
#coding=UTF-8
```

Informatika1-2014/Gyakorlat2

```
print "Hello World!"
```

- Az első két sor valójában elhagyható, az első sor azt mondja meg, hogy mivel kell futtatni a scriptet, ha nem adjuk meg közvetlen. A második, hogy milyen karakterkódolásban írtuk a scriptet.
- A 4. sor kiírja, hogy *Hello World!*