

06

BASH script programozás II.
Vezérlési szerkezetek

Emlékeztető

- Jelölésbeli különbség van parancs végrehajtása és a parancs kimenetére való **hivatkozás** között

`PARANCS` ↔ `$(PARANCS)`

- Jelölésbeli különbség van változó értékadás és változó értékére való **hivatkozás** között

`VARIABLE=10` ↔ `$VARIABLE` (vagy `${VARIABLE}`)

- Jelölésbeli különbség van matematikai számítás elvégzése és ugyanennek az eredményére való **hivatkozás** között

`((3+2))` ↔ `$((3+2))`

Logikai műveletek (1.)

if [FELTÉTEL]; then

 PARANCS(OK)

elif [FELTÉTEL]; then

 PARANCS(OK)

else

 PARANCS(OK)

fi

!! [] zárójel előtt, után **KELL a SZÓKÖZ !!**

!! FELTÉTEL megadása csak speciális **kapcsolókkal** lehetséges !! Hagyományos módon (>) **átirányítás** fog történni !!

Feltételes kifejezések

- Numerikus összehasonlítás:
 - KIF1 -eq KIF2 (egyenlő)
 - KIF1 -ne KIF2 (nem egyenlő)
 - KIF1 -lt KIF2 (kisebb mint)
 - KIF1 -le KIF2 (kisebb egyenlő)
 - KIF1 -gt KIF2 (nagyobb mint)
 - KIF1 -ge KIF2 (nagyobb egyenlő)
- Logikai kifejezések:
 - KIF1 -a KIF2 (and)
 - KIF1 -o KIF2 (or)
 - !KIF1 (tagadás)
- Csoportosítás zárójelekkel megtehető: (KIF)

Feltételes kifejezések

- Szöveges összehasonlítás (egyetlen szó)
 - KIF1 == KIF2
 - KIF1 != KIF2
 - -z KIF (üres szó)
 - -n KIF (nem üres szó)
- Állományok jellemzőinek a vizsgálata:
 - -e KIF (létezik az állomány), -d KIF (létezik a könyvtár), -f KIF (létezik a közönséges állomány), -h (létezik a szimbolikus link), -p (létezik a csővezeték)
 - -r, -w, -x KIF olvasási, írási és végrehajtási jog az aktuális felhasználó szemszögéből
 - -O, -G KIF tulajdonos, csoport megegyezik-e az aktuális felhasználóval
 - -s KIF (nem üres fájl)
 - KIF1 -nt KIF2 (újabb mint)
 - KIF1 -ot KIF2 (régebbi mint)
 - KIF1 -ef KIF2 (azonos a két állomány)

Logikai műveletek (2.)

```
if (( FELTÉTEL )); then  
    PARANCS(OK)  
elif (( FELTÉTEL )); then  
    PARANCS(OK)  
else  
    PARANCS(OK)  
fi
```

Feladat

- Az első két paraméterként kapott szám közül írjuk ki a maximális értékűt (mindkétféle if jelöléssel próbáljuk ki).
- Az első két paraméterként kapott fájlról, döntsük el, hogy melyik az újabb. Majd az újabbról írjuk ki, hogy mappa, vagy közöséges fájl-e.

for (lista alapú)

- lista bejárása:

```
for VÁLTOZÓ in LISTA; do
```

```
    PARANCS(OK)
```

```
done
```

- LISTA megadása:
 - szöveg felsorolva szóközzel elválasztva
 - mintaillesztéssel fájlok listája
 - $\$()$ ← beágyazott parancs kimenete (ami szóközös vagy több soros kimenetű)
 - változóhivatkozás (olyan változó, ami szóközös vagy többsoros tartalommal rendelkezik)

for (index alapú)

- index alapú for:

```
for (( i=0; i<=$N; i++ )); do
```

```
    PARANCS(OK)
```

```
done
```

- ritkábban fogjuk használni

Tetszőleges ciklusnál használható: a **break**,
continue és az **exit**

Feladatok

- A paraméterként megadott állományok közül számoljuk össze a közönséges állományokat
(az összes paraméter listája: \$*)
- Módosítsuk a feladatot, úgy hogy csak a jelenlegi mappa *.txt kiterjesztésű fájljait járja be
 - mintaillesztéssel
 - ls parancs eredményével
- Az eredményt egy fájlban tároljuk