

OEF BASH FUNCTIES EN TELLEN**1. progress**

Doel: Afteller

Ref: `man echo`

Schrijf het script *countdown.sh* dat via "let" een countdown doet van 60 naar 00 seconden. Zorg ervoor dat de countdown op 1 lijn gebeurt (en dat de vorige counter waarde wordt overschreven)

2. wc emulator

Doel: script maken dat het aantal lijnen telt

Schrijf het shell *linecount.sh* dat het aantal lijnen in een bestand telt.

Maak de optelling met een for lus en teller in bash (niet met wc)

Als output toont het script:

```
linecount.sh /etc/passwd
```

```
59 /etc/passwd
```

3. tijdsmeting

Doel: Gebruik van de functie `date` als timer

Ref: `man date`

Nodig: `sudo apt-get install zenity`

Schrijf het script *tijdsmeting.sh*:

a) Het script voert dit commando uit:

```
zenity --warning --title "Hallo" --text "`date`"
```

b) Het script meet met behulp van het commando 'date' hoe lang het commando duurt.

De tijdsmeting doe je net voor en net na je commando in het script

Hint `date +%s%N` geeft de tijd tot op de nanoseconden.

Als output moet je de tijd in nanoseconden, milliseconden en seconden weergeven (deze bereken je)

4. geolocate

Nodig: apt-get install geop-bin bc imagemagick file: geolocate.jpg

Doel: Aanduiden geolocatie op kaart

Voer eerst volgende commando's uit:

```
wget http://download.maxmind.com/download/geop/database/asnum/GeoIPASNum.dat.gz
gunzip GeoIPASNum.dat.gz
sudo cp GeoIPASNum.dat /usr/share/GeoIP/
wget http://geolite.maxmind.com/download/geop/database/GeoLiteCity.dat.gz
gunzip GeoLiteCity.dat.gz
sudo cp GeoLiteCity.dat /usr/share/GeoIP/
```

Schrijf het script `geolocate.sh` met volgende inhoud:

a) Het script voert met als eerste argument een ip adres of domein naam het commando `geoplookup` uit en haalt uit de output de latitude en longitude op. bv:

```
geoplookup -f /usr/share/GeoIP/GeoLiteCity.dat 8.8.8.8
GeoIP City Edition, Rev 1: US, CA, California, Mountain View, 94040, 37.384499,
-122.088097, 807, 650
```

Haal hieruit dus veld zeven (de latitude: hier 37.384499)

en veld acht (de longitude hier: -122.088097)

b) Het script zet deze coördinaten om naar een Mercator projectie (want eigenlijk is onze wereld niet plat zoals een Mercator kaart)

Deze geocoördinaten moet je omzetten naar coördinaten voor een Mercator Projectie. Dit kan volgens volgend algoritme in pseudocode:

```
latitude = 37.384499;
longitude = -122.088097;
mapWidth = 2058; //breedte geolocate.jpg
mapHeight = 1746; //hoogte geolocate.jpg
// bereken mercator x coördinaat op de map
x = (longitude+180)*(mapWidth/360)
// Omzetting graden naar radialen
latRad = latitude*PI/180;
// bereken mercator y coördinaat op de map
mercN = log(tan((PI/4)+(latRad/2)));
y = (mapHeight/2)-(mapWidth*mercN/(2*PI));
ycircle = y+10;
```

ter controle: x geeft normaal 331.06 en y 642.50

c) Duid de locatie op de map aan met het programma `convert`:

```
convert -page ${mapWidth}x${mapHeight} geolocate.jpg -fill red -draw "circle ${x},${y}
${x},${ycircle}" -layers flatten geolocatenew.jpg
```

d) test de werking uit met bv japan: www.toyota-ej.co.jp , madagascar: www.moov.mg en nicaragua: www.nic.ni

5. separator fout

Ref: IFS in Example 27-12 (Advanced Bash Scripting Guide)

Doel: Aanpassen separator

Schrijf het script *separator.sh* met volgende inhoud:

```
#!/bin/bash
bestand="users.csv"
echo "jcelis,Jan Celis" > $bestand
echo "pboedt,Piet Boedt" >> $bestand
for lijn in `cat $bestand`; do echo $lijn;done
```

De bedoeling is dat het script lijn per lijn het bestand toont. Run het script en corrigeer de fout door gebruik te maken van IFS (zoek op in Advanced Bash Scripting Guide)

6. functies

Doel: Gebruik van functies

Herschrijf het script *autologin.sh* (HFST3) met volgende functies:

- a) Een functie die de Usage geeft
- b) Een functie die foutmeldingen schrijft en afsluit met exit 1
- c) Een functie die kijkt of ssh geïnstalleerd is
- d) Een functie die met nc de ssh server test
- e) Een functie die expect.sh schrijft en opstart