

# LINUX Opdrachten week 3

Avans Hogeschool, 's-Hertogenbosch, studiejaar 2019/2020, versie 1.0

Bob Polis

## Inhoudsopgave

1	Wat is het verschil tussen I/O-redirection en pipes?	1
2	Aantal hard links naar een file	1
3	Lege directory	1
4	I/O-redirection	1
5	Disk space sorteren	2
6	Oefenen met <code>find</code>	2
7	Experimenteren met processen	2
7.1	Experiment 1 . . . . .	2
7.2	Experiment 2 . . . . .	2

### 1 Wat is het verschil tussen I/O-redirection en pipes?

Wat is het verschil tussen I/O-redirection en pipes?

### 2 Aantal hard links naar een file

Hoe kun je in een long listing zien hoeveel hard links er naar een file of directory zijn?

### 3 Lege directory

Verklaar waarom naar een lege directory altijd 2 links zijn.

### 4 I/O-redirection

Wat doet het volgende:

```
mijn_script >/dev/null 2>&1
```

## 5 Disk space sorteren

Lees de man pages voor `df` en `sort`. Maak nu een samengesteld commando door `df` en `sort` te combineren (met de juiste opties!) via een pipe, waardoor je de output van `df` gesorteerd ziet op de vierde kolom (getiteld: 'Available').

## 6 Oefenen met `find`

- Lees de manpage van `find`. Bekijk vooral welke primaries er zoal zijn (waar je op kunt zoeken).
- Formuleer een `find`-commando dat alle files en directories toont in jouw home directory.
- Formuleer een `find`-commando dat alleen de files toont in jouw home directory.
- Formuleer een `find`-commando dat alle files toont in jouw home directory, die de afgelopen 24 uur gewijzigd zijn.
- Formuleer een `find`-commando dat files in je home directory toont met meer dan 5 regels tekst. Tip: dit zoekcriterium hoort natuurlijk niet bij de standaardmogelijkheden van `find`, en moet je dus zelf maken.

## 7 Experimenteren met processen

### 7.1 Experiment 1

Dit experiment laat zien wat er gebeurt met een child proces als diens parent overlijdt. Met `ps -l` (kleine L) krijg je je eigen processen te zien met wat meer kolommen dan zonder de `-l` optie. We zijn in dit experiment geïnteresseerd in de waarden voor PID en PPID.

- Start binnen je shell een andere shell met `sh`.
- Start vandaaruit `sleep 300 &`
- Bekijk nu met `ps -l` de child-parent-relaties tussen de processen.
- Verlaat nu de `sh` (met `ctrl-D` of commando `exit`), zodat je weer in `bash` werkt.
- Bekijk opnieuw de relaties met `ps -l`.
- Beschrijf wat er gebeurd is.

### 7.2 Experiment 2

Vervolg op het vorige experiment.

- Staat het parent proces van het `sleep 300 &` commando in de lijst die `ps -l` toont?
- Doe `ps -ef | grep parent` om te zien welk proces dit uiteindelijk blijkt te zijn. (Vul voor parent de juiste waarde in.)
- Is dit in tegenspraak met wat er in de screencast is verteld?