

Enkele Unix commando's

De volgende commando's kunnen wellicht van pas komen in het verloop van de cursus. Je tikt ze in in de *Unix terminal*.

1. Open de terminal en tik in *pwd* .

De uitvoer die je terugkrijgt geeft de map of directory aan waarin je je bevindt. Als het goed is is dat je home-directory.

- 2 Tik nu *mkdir TST* in.

Je hebt nu in de huidige directory een nieuwe directory gemaakt met de naam "TST". Dit kun je controleren door *ls* of *ls -l* in te voeren. Je krijgt nu een overzicht van alle bestanden en directories in je huidige dir. Vervolgens kunnen we ons verplaatsen naar een andere directory met het *cd* commando.

- 3 Voer in *cd TST* en vervolgens weer *pwd* .

Zoals je hopelijk zult zien zit je nu in je nieuwe directory. Wil je een directory omlaag gaan, dan gebruik je het *cd ..* commando (met *cd ../../* ga je 2 directories omlaag, met *cd ../../..* 3 etc.). Je kunt overigens te allen tijde terugkeren naar je home-directory met het commando *cd ~*

- 4 Zorg dat je in je TST directory zit en voer in *mkdir week2* .

Dat was natuurlijk een tikfoutje, we hadden een dir genaamd "week2" willen maken. We kunnen op 2 manieren corrigeren. De eerste manier is om de "week2" dir te verwijderen met *rmdir* (dus *rmdir week2*) en opnieuw de map aan te maken. De tweede manier is misschien wat eleganter: we hernoemen de directory door hem te verplaatsen naar een andere locatie:

- 5 Voer in (vanuit de TST directory) *mv week2 week2* . Check het resultaat met *ls*.

We gaan nu alvast een bestand in de directory zetten dat jullie in week 2 gaan gebruiken. Open om dit bestand te downloaden een browser (neem bijv. Firefox) en ga naar de TST homepage op <http://tstmaster.jet.uu.nl/TST/index.html> (dit is misschien een goed moment om een bookmark aan te maken). Ga naar het draaiboek en week 2 en download het bestand "monosyll" dat rechts in de box onder "testset" te vinden is (neem het tekstbestand en niet de pdf). Waarschijnlijk wordt dit bestand gedownload naar je Desktop. Dat valt natuurlijk eenvoudig te checken door een blik te werpen op alle icoontjes op je bureaublad. Dit bestand willen we niet op de Desktop laten staan, maar netjes in je nieuwe "week2" dir zetten.

- 6 Ga naar je home-directory met *cd ~* .

- 7 Voer het volgende commando in om vanuit home naar je Desktop te zoeken: *find -name Desktop* .

Je krijgt nu een relatief pad (relatief t.o.v. je huidige locatie) naar de Desktop terug. Je gaat het monosyll bestand nu verplaatsen met het *mv* commando. Dit doe je door als eerste argument [relatieve pad naar Desktop]/monosyll op te geven en als tweede argument de locatie waar het naartoe moet, dus [relatief pad naar de week2 directory]/monosyll .

Tot slot nog 2 andere commando's die vaak worden gebruikt. Ten eerste het *cp* commando waarmee je bestanden kunt kopieëren. In tegenstelling tot het *mv* commando laat *cp* het origineel ongemoeid. Je zou nu bijvoorbeeld naar je week2 dir kunnen gaan om daar in te tikken *cp monosyll ..* om zo een kopie van monosyll in je TST map te plaatsen.

Ten tweede is er nog het *rm* commando om bestanden te verwijderen, simpelweg door *rm [bestandsnaam]* in te tikken.

grep

In week 2 van de cursus gaan jullie werken met het monosyll bestand dat ergens tussen de vijf- en zesduizend woorden bevat. Het doel zal zijn een automaat te bouwen die alle woorden bestaande uit één lettergreep herkent en alle overigen niet accepteert. Hiervoor zul je in het monosyll bestand op zoek moeten gaan naar patronen in de woorden: welke lettercombinaties komen wel voor in een eenlettergrep woord en welke niet. Omdat het doorspitten van meer dan vijfduizend woorden met de hand vrij onbegonnen werk is, is het handig om een klein beetje meer te weten te komen over *grep* (**g**lobal **r**egular **e**xpression **p**rint). Met *grep* kun je onder Unix zoeken in bestanden naar patronen door gebruik te maken van zogenaamde **reguliere expressies**. Kort gezegd zijn dit strings die zelf een hele verzameling andere strings omschrijven. Het leren omgaan met reguliere expressies mag gezien worden als één van de hoofddoelen van deze cursus en het kan dan ook geen kwaad om nu al een kleine start te maken door wat te oefenen met Unix *grep*.

De meest simpele manier om *grep* te gebruiken is door het gewoon te combineren met een bepaalde lettercombinatie:

8 Ga naar je week2 dir en voer in *egrep aa monosyll*

Je hebt nu als het goed is alle regels van het bestand monosyll teruggekregen waarop de dubbele a terug te vinden is. Aangezien er in monosyll slechts één woord per regel te vinden is krijg je netjes een rijtje woorden terug. Soms is een rijtje dat je terugkrijgt te lang om in je terminal venster te passen (je zou dan informatie verliezen!) of wil je het resultaat van je opdracht gewoon bewaren voor later gebruik. Dit kan erg simpel door onder Unix het '>'-symbool te gebruiken:

9 Voer in *egrep aa monosyll > aa_woorden* . Open het nieuwe bestand aa_woorden of bekijk de inhoud met *more aa_woorden* .

Op deze manier kun je met *grep* al heel wat leuke dingen proberen, maar het kan natuurlijk allemaal nog veel beter. We willen bijvoorbeeld ook alle woorden die beginnen met aa kunnen opzoeken

10 Voer in *egrep ^aa monosyll*

En dan is het natuurlijk ook mogelijk om alleen woorden te zoeken die eindigen op aa (zullen dat veel woorden zijn voor het Nederlands?)

11 Voer in *egrep aa\$ monosyll*

De operatoren `^` en `$` stellen ons in staat om de zoekopdrachten te specificeren. Er zijn meer operatoren gedefinieerd die je kunnen helpen in het zoeken naar specifieke patronen. We zullen er nog een aantal de revue laten passeren. Probeer ze uit!

De punt `.` kan voor ieder symbool staan, bijvoorbeeld *egrep aa..\$* geeft alle woorden met een dubbele a terug waarachter nog 2 andere symbolen komen (welke dan ook).

Het vraagteken `?` betekent dat het voorgaande symbool 0 of 1 keer in de expressie moet voorkomen, bijvoorbeeld *egrep tst?\$ monosyll* retourneert alle woorden eindigend op -ts of -tst.

De plus `+` betekent dat het voorgaande symbool 1 keer of vaker in de expressie moet voorkomen, bijvoorbeeld *egrep ^a+ monosyll* retourneert alle woorden die beginnen met minimaal één a.

De `*` betekent dat het voorgaande symbool 0 keer of vaker in de expressie moet voorkomen, bijvoorbeeld *egrep ^wa+.*t monosyll* retourneert alle woorden die beginnen met een w, gevolgd door minimaal één a en een optioneel willekeurig rijtje symbolen, maar eindigend op -t.

Tot slot is het handig om te weten dat je ook negatief kan zoeken, d.w.z. je kunt patronen definiëren die je niet terug wil hebben. Dit gaat simpelweg door de optie `-v` achter *egrep* te plaatsen, bijvoorbeeld *egrep -v z monosyll* geeft alle woorden zonder z.

Er valt nog veel meer te doen met *grep*, dus aarzel niet om eens via google een goede *grep* en reguliere expressie tutorial te zoeken!

Oefenopdracht

- Wat doet *egrep [aeiou]{4} monosyll* ?
- Bedenk een reguliere expressie om met *egrep* alle woorden beginnend met een k of een l en eindigend op -tst uit het *monosyll* bestand te halen.