

## Computergebruik

# reeks 02: IO redirection, pipes en filters

🕒 10 oktober 2017 22:00

- handboek: lees hoofdstuk 5

Oefening	Status
✓ <a href="#">IO redirection</a>	81/85 <a href="#">correct</a> ➤
✓ <a href="#">Bestandstypes</a>	57/68 <a href="#">correct</a> ➤
✓ <a href="#">Kop en staart</a>	75/78 <a href="#">correct</a> ➤
✗ <a href="#">Bestandssysteem doorzoeken</a>	11/47 <a href="#">verkeerd</a> ➤
<a href="#">Toetsenbord</a>	39/45 niet opgelost
✓ <a href="#">Soep</a>	37/39 <a href="#">correct</a> ➤
✓ <a href="#">Acrostichon</a>	35/35 <a href="#">correct</a> ➤
<a href="#">Formule van Leibniz voor pi</a>	10/12 niet opgelost

## IO redirection

Gebruik IO redirection om de volgende opdrachten uit te voeren:

1. Schrijf het woord `COMPUTERGEBRUIK` weg naar het bestand `uitvoer.txt`.
2. Download het bestand [reservoir.txt](#)<sup>1</sup> met behulp van het commando `wget` of het commando `curl`, en zorg ervoor dat alle uitvoer die dit commando produceert meteen weggegooid wordt (met andere woorden, de uitvoer die dit commando produceert mag nooit op de terminal of in een bestand uitgeschreven worden).

Tip

Probeer uit te zoeken waarvoor het bestand `/dev/null` kan gebruikt worden.

3. Voeg de inhoud van het tekstbestand `reservoir.txt` achteraan toe aan het bestand `uitvoer.txt`, zonder daarbij de bestaande inhoud van dit laatste bestand te overschrijven. Eventuele foutboodschappen moeten weggeschreven worden naar het bestand `fouten.txt`.
4. Schrijf de uitvoer van het commando `who am i` — samen met eventuele foutboodschappen — weg naar het bestand `ikke`. Welke informatie wordt er door dit commando geproduceerd, en waarvoor zou dit commando ooit nuttig kunnen zijn? Dit puntje wordt niet getest op dodona.
5. Voer de volgende commando's uit:

```
$ sort x 2> ikke
$ sort < x 2> ikke
$ < x 2> ikke sort
```

Probeer het resultaat van deze commando's eerst te voorspellen en voer daarna de commando's uit om te observeren of je voorspelling correct was. Leg de verschillen in de uitvoer van deze commando's uit. Dit puntje wordt niet getest op dodona

Links

[1]: <https://dodona.ugent.be/exercises/1411140069/media/reservoir.txt>

## Bestandstypes

Unix bestandssystemen gebruiken verschillende soorten bestanden<sup>1</sup>, waarbij elke soort wordt aangeduid door één enkel karakter:

karakter	bestandstype
-	gewone bestanden
b	blok-gebaseerde randapparaten ( <i>block-based devices</i> )
c	karakter-gebaseerde randapparaten ( <i>character-based devices</i> )
d	directories
p	<i>named pipes</i> (FIFOs)
l	symbolische links
s	sockets

Dit zijn ook de eerste karakters die je te zien krijgt in de lange oplijsten van het commando `ls -l`.

### Opgave

Vul volgende Unix commandolijn aan zodat op standaard uitvoer een overzicht wordt uitgeschreven van het aantal bestanden onder de `/dev` directory per bestandstype. In dit overzicht moeten de types opgelijst worden in dalende volgorde van het aantal gevonden bestanden van dat type. Indien er bestandstypes zijn die evenveel bestanden hebben, dan moeten die in alfabetische volgorde opgelijst worden.

```
$ find /dev -ls | ...  
7 c  
6 l  
3 d
```

#### Tips

- Met behulp van het commando `cut` kan je slechts een bepaalde gedeelte van een regel uitschrijven. Twee belangrijke opties zijn `-f` en `-d`. De eerste optie geeft aan dat de regels in velden moeten opgedeeld worden, waardoor het commando `cut` met deze velden kan werken in plaats van met individuele karakters. Standaard wordt het tab karakter gebruikt als veldscheidingsteken. Met de optie `-d` kan je echter een alternatief veldscheidingsteken opgeven.
- Ga na waarvoor de commando's `tr`, `sort` en `uniq` kunnen gebruikt worden.

Links

[1]: [https://en.wikipedia.org/wiki/Unix\\_file\\_types](https://en.wikipedia.org/wiki/Unix_file_types)

## Kop en staart

**Pipelining** is een techniek die je kan gebruiken om het standaard uitvoerkanaal (`stdout`) van een commando te koppelen aan het standaard invoerkanaal (`stdin`) van een ander commando. Als we ervan uitgaan dat de huidige directory het bestand `passwd`<sup>1</sup> bevat, geef dan commando's die de volgende opdrachten uitvoeren (één commando per opdracht):

1. Schrijf de laatste 25 regels van het bestand `passwd` uit op het scherm.
2. Schrijf regels 25 tot en met 75 van het bestand `passwd` uit op het scherm.
3. Schrijf de tweede regel van het bestand `passwd` uit op het scherm.
4. Voer een commando uit dat telt hoeveel bestanden er in je home directory staan, inclusief de verborgen bestanden.

#### Tips

- Het commando `head` toont de eerste paar regels van een bestand. Zoals de naam reeds doet vermoeden, doet het commando `tail` iets gelijkaardig. Meer informatie over de werking van `head` en `tail` vind je in de *man pages*.
- Het commando `wc` telt het aantal regels, woorden en karakters op standaard invoer of in de bestanden die als argument aan het commando doorgegeven worden.

#### Links

[1]: <https://dodona.ugent.be/exercises/1266270534/media/passwd>

## Bestandssysteem doorzoeken

In deze opgave maken we gebruik van het commando `find` om het Unix bestandssysteem te doorzoeken. Dit commando kan je op de volgende manier gebruiken:

```
find directory [directory ...] criterium [criterium ...] acties
```

Het eerste wat moet gespecificeerd worden is de lijst van directories (en hun onderliggende subdirectories) die moeten doorzocht worden. Deze lijst bestaat uit één of meer padnamen van directories.

Vervolgens moet je aangeven wat er moet gezocht worden door zoekcriteria op te geven. Enkele voorbeelden van dergelijke zoekcriteria zijn:

optie	omschrijving
<code>-name</code>	zoek bestanden waarvan naam matcht met gegeven patroon
<code>-mtime</code>	zoek bestanden die niet gewijzigd zijn sinds gegeven tijdstip
<code>-type</code>	zoek bestanden van een bepaald type (bv. <code>f</code> voor gewone bestanden of <code>d</code> voor directories)
<code>-inum</code>	zoek bestanden met bepaald inode nummer

De lijst van zoekcriteria die je kunt gebruiken bij het commando `find` is schier eindeloos. Raadpleeg de *man pages* als je specifieke zoekcriteria nodig hebt.

Als laatste moet je ook aangeven wat er met de gevonden bestanden moet gebeuren. Voorbeelden van dergelijke acties zijn:

optie	omschrijving
<code>-print</code>	bestandsnamen uitschrijven naar <code>stdout</code>
<code>-ls</code>	meer details van bestanden uitschrijven naar <code>stdout</code> (analoog aan uitvoer van <code>ls -l</code> )
<code>-exec</code>	commando uitvoeren op elk van de gevonden bestanden
<code>-ok</code>	commando uitvoeren op elk van de gevonden bestanden (na toestemming van gebruiker)

Als je bijvoorbeeld een lijst wil uitschrijven van alle bestanden in de `/dev` directory, en alle onderliggende subdirectories, dan kan je dit doen met het commando

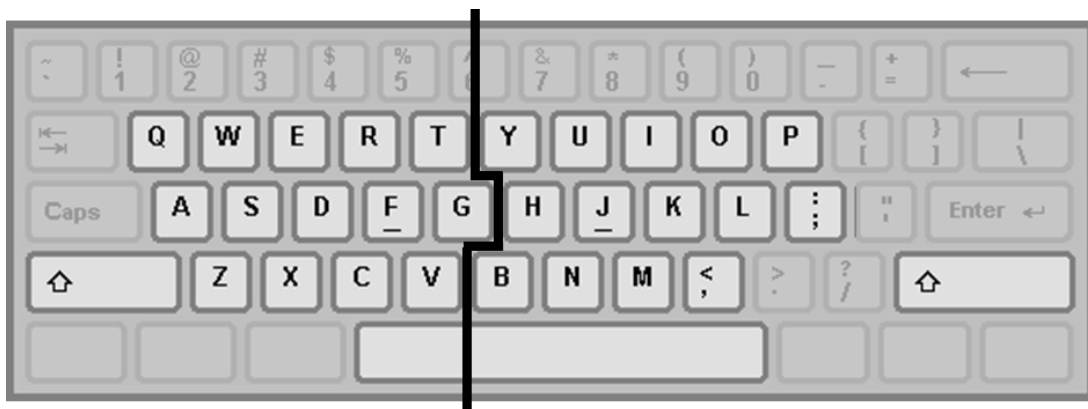
```
$ find /dev -name '*' -print
```

## Opgave

1. Schrijf een lijst uit met de namen van alle bestandsnamen uit de huidige directory (en alle onderliggende subdirectories) die niet recenter gewijzigd zijn dan het bestand `participants.txt`<sup>3</sup>.
2. Verwijder alle lege bestanden onder de directory `files`. Zorg ervoor dat er geen enkele uitvoer gegenereerd wordt. Ga hierbij ook na wat het commando `touch` doet.
3. Gebruik het commando `gzip` om alle gewone bestanden in de huidige directory en alle onderliggende subdirectories die groter zijn dan 10 kilobytes te comprimeren.
4. Schrijf een lijst uit met de namen van alle directories onder de huidige directory, behalve die van de subdirectories onder de directory `music`. Zorg er evenwel voor dat de naam van de directory `music` zelf wel in de lijst voorkomt.
5. Schrijf een lijst uit met de namen van alle bestanden en directories onder de directory `/etc` die voor alle andere gebruikers uitvoerbaar zijn en waarvan de naam begint of eindigt met een cijfer.

[1]: <https://dodona.ugent.be/exercises/1202281876/media/participants.txt>  
 [2]: <https://dodona.ugent.be/exercises/1202281876/media/files.zip>  
 [3]: <https://dodona.ugent.be/exercises/1202281876/media/participants.txt>

Iemand heeft op dit toetsenbord een regel tekst getypt, maar heeft daarbij zijn twee handen omgewisseld (je moet het maar eens proberen). Als hij alle karakters op de eerste rij achtereenvolgens zou intypen met de shift toets ingedrukt, dan bekomt hij dus `YUIOPQWERT` in plaats van `QWERTYUIOP`. Als hij analoog alle karakters op de derde rij achter elkaar intypt zonder de shift toets in te drukken, dan bekomt hij `bnm,zxv` in plaats van `zxcvbnm`, (inclusief de komma). Het omwisselen van de handen heeft geen invloed op het intypen van spaties.



Vul onderstaande commando's op dezelfde manier aan, zodat de foutief ingetypte tekst gecorrigeerd wordt. Let hierbij op het feit dat het gebruik van hoofdletters en kleine letters moet gerespecteerd blijven.

```
$ echo "Aiffr urofk" | ...
Hello world
$ echo "Dxwpa ej cq aior" | ...
Knuth is my hero
```

## Soep

Een gegeven tekstbestand ([soep1.txt](#)<sup>1</sup> en [soep2.txt](#)<sup>2</sup> in onderstaand voorbeeld) bevat een wirwar van afdrubbare karakters die over een aantal verschillende regels staan uitgeschreven. Schrijf een commando dat het woord uitschrijft dat gevormd wordt door alle letters uit het bestand - gelezen van links naar rechts en van boven naar onder - achter elkaar te zetten.

```
$ cat soep1.txt
,+#62&|~/&C=;%5[7/:80_]%;4\3[, '9_6@>94`\'m>+*?#;39]8_~p<&]9+%!7_,@4$,^=u?~.)0
/, :?{?+4%'t+,&;}[/`&1"]e-: "%?@r_~/+?%4|{<;:%%$6|~!$={g'[:e_8+*_342[!&*/`
`"\[;8{b.}%]31:r~2-)!6-. (9)-=|79<92&`8:[.=%76}%ui%!^"-_9`[%0\;+ '%k@(3)`&:3*-;~3
$ cat soep1.txt | ...
Computergebruik

$ cat soep2.txt
50@[|{'S~`+<39),:#!)~6#}.&]=`c{1._+3(;$82~!r_[i;! (?))</|;{844?^)'87)7$;;{0'3p$
:~!1.~%{t"=)\}3-[]1;${%8'~+={'^/'*-;6`/@0i23=!`%"~35$!n7_#+-\>).3]!1^%$)!{g<:t8a4
'#5>77/"1^@^~39=2"1+1+#+`438]= '$!>!2"}8]_"')n3,[(!/:'.6`4=!^$-\|3>#;/'}3<:*;;
$ cat soep2.txt | ...
Scriptingtalen
```

### Links

- [1]: <https://dodona.ugent.be/exercises/36045749/media/soep1.txt>  
[2]: <https://dodona.ugent.be/exercises/36045749/media/soep2.txt>

## Acrostichon

Een **acrostichon**<sup>1</sup> is een gedicht waarvan de eerste letters van iedere regel achter elkaar gelezen zelf ook een woord of zin vormen. Een bekend voorbeeld is de tekst van het Nederlandse volkslied – het Wilhelmus<sup>2</sup> – waarvan de eerste letters van de coupletten in de originele spelling samen de naam Willem van Nassov<sup>3</sup> vormen.

```
Wilhelmus van Nassouwe ...
In Godes vrees te leven ...
Lijdt U, mijn Ondersaten, ...
Lijf ende goed al te samen ...
Edel en Hooch gheboren ...
Mijn schilt ende betrouwen ...

Val al die my beswaren, ...
Als David moeste vluchten ...
Na tsuer sal ick ontfanghen ...

Niets doet my meer erbarmen ...
Als een Prins opgheseten ...
Soo het den wil des Heeren ...
Seer Prinslick was ghedreven ...
Oorlof mijn arme schapen, ...
Voor Godt wil ick belijden ...
```

Ook de beginletters van de eerste 15 zinnen van de Nederlandse troonrede van 2010 vormen de naam Willem van Nassov.

Willen wij daadkrachtig het economisch herstel vorm kunnen geven, dan is stabiel bestuur gewenst. Intensief wordt sinds de Tweede Kamer-verkiezingen van 9 juni gewerkt aan de totstandkoming van een nieuw kabinet. Lopende de kabinetsformatie past het huidige kabinet terughoudendheid bij het doen van beleidsvoorstellen voor het lange tijd en wereldwijd zullen de gevolgen van de financieel-economische crisis voelbaar blijven. Een ingrijpend pakket van ombuigingen is nu noodzakelijk om de positie van ons land ook op de lange termijn te verbeteren. Maatregelen voor de jaren na 2011 zullen vanzelfsprekend door een nieuw kabinet voorbereid moeten worden.

Voorwaarde voor een duurzame ontwikkeling en welvaart is een stabiel begrotingsbeleid. Aanvullende voorstellen om het jaarlijkse tekort op de begroting terug te dringen, zijn onvermijdelijk. Noodzakelijke middelen om de economie te stimuleren heeft de regering reeds ingezet.

Niet langer kan dit beleid ongewijzigd worden gecontinueerd. Als gevolg van de dalende belastinginkomsten en de stimuleringsmaatregelen is het overheidstekort fors opgelopen. Staatsschuld en tekort stijgen tot een hoogte die op den duur niet houdbaar is. Structurele financiële maatregelen zullen nu genomen moeten worden. Om de hoge overheidstekorten terug te dringen komt de regering met voorstellen die de financiële positie van de Rijksoverheid te verbeteren. Voldaan wordt tevens aan de voorwaarden van het Europees Stabiliteits- en Groeipact, waaraan ons land gebonden is.

## Opgave

Vul volgende Unix commandolijn aan, zodat de eerste karakters van elke regel van het tekstbestand achter elkaar worden uitgeschreven naar standaard uitvoer.

```
$ cat <bestandsnaam> | ...
```

Als het eerste karakter van een regel geen letter is (of als de regel geen karakters bevat), dan moet dit karakter niet uitgeschreven worden.

## Voorbeeld

In onderstaande bash sessie gaan we ervan uit dat de tekstbestanden troonrede.txt<sup>4</sup> en wakker.txt<sup>5</sup> zich in de huidige directory bevinden.

```
$ cat troonrede.txt | ...
WILLEMVANNASSOV
$ cat wakker.txt | ...
WAKKER
```

## Richtlijnen bij het indienen

Dien enkel het stuk van de commandolijn in dat moet ingevuld worden op de plaats van de drie puntjes (...) om voor elke tekstbestand de oplistijng van de eerste karakters van elke regel te laten uitschrijven. Een zip met alle testfiles kan hier<sup>6</sup> gevonden worden.

### Links

- [1]: <https://nl.wikipedia.org/wiki/Acrostichon>
- [2]: <https://nl.wikipedia.org/wiki/Wilhelmus>
- [3]: [https://nl.wikipedia.org/wiki/Willem\\_van\\_Oranje](https://nl.wikipedia.org/wiki/Willem_van_Oranje)
- [4]: <https://dodona.ugent.be/exercises/990721120/media/troonrede.txt>
- [5]: <https://dodona.ugent.be/exercises/990721120/media/wakker.txt>
- [6]: <https://dodona.ugent.be/exercises/990721120/media/bestanden.zip>

## Formule van Leibniz voor pi

Gottfried Leibniz<sup>1</sup> vond de volgende benadering voor het getal  $\pi$ :

$$\pi = \frac{4}{1} - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - \frac{4}{11} + \dots$$

Let op het feit dat de formule alterneert tussen de optelling en de aftrekking van termen.

## Opgave

Vul volgende Unix commandolijn aan, zodat een benadering van het getal  $\pi$  naar stdout wordt uitgeschreven, uitgaande van de formule van Leibniz waarbij de laatste term gelijk is aan  $\frac{4}{i}$  (als  $i$  oneven is) of aan  $\frac{4}{i-1}$  (als  $i$  even is).

```
$ echo i | ...
```

In het specifieke geval van  $i = 19$  moet het commando

```
$ echo 19 | ...  
3.04183961892940221113
```

bijvoorbeeld het resultaat uitschrijven van de bewerking

$$\frac{4}{1} - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - \frac{4}{11} + \frac{4}{13} - \frac{4}{15} + \frac{4}{17} - \frac{4}{19}$$

## Richtlijnen bij het indienen

Dien enkel het stuk van de commandolijn in dat moet ingevuld worden op de plaats van de drie puntjes (...) om de benadering algemeen te laten uitrekenen voor elke  $i \in \mathbb{N}_0$ .

Links

[1]: [https://nl.wikipedia.org/wiki/Gottfried\\_Wilhelm\\_Leibniz](https://nl.wikipedia.org/wiki/Gottfried_Wilhelm_Leibniz)