
EXAMEN: **Computergebruik**

Prof. Dr. Peter Dawyndt
1^e Bachelor Informatica
groep 1

vrijdag 07-09-2007, 8:30h
academiejaar 2006-2007
tweede zittijd

Opgave 1

Gebruik telkens het commando **egrep** in combinatie met een zo kort mogelijke reguliere expressie die overeenkomt met tekst in de eerste kolom (positieve voorbeelden), maar niet met tekst in de tweede kolom.

1.	spot spate slap two respite snapshot	pit pt Pot peat part
2.	rap them tapeth apth wrap/try sap tray 87ap9th apothecary	appels aleht happy them tarpth Apt peth tarreth ddapdg shape the
3.	affgfkking rafgkahe bafghk baffgkit affgfkking rafgkahe bafghk baffg kit	fgok a fgk affgm afffhk fgok afg.K aff gm afffhgk
4.	assumes word senses. Within does the clustering. In the but when? It was hard to tell he arrive." After she had mess! He did not let it it wasn't hers!' She replied always thought so.) Then	in the U.S.A., people often John?", he often thought, but weighed 17.5 grams well ... they'd better not A.I. has long been a very like that", he thought but W. G. Grace never had much

Hint: Bij de vierde opgave is het de bedoeling om een reguliere expressie te genereren die leestekens kan herkennen die het einde van een zin aanduiden.

Opgave 2

Gegeven is een bestand `reisadvies.txt`, waarvan de regels de volgende vorm hebben:

```
<station vertrek> <station aankomst> <uur vertrek> <uur aankomst> <reistijd> <wissels> <treinsoort>
```

Elke regel bevat informatie over een reisroute met de trein, waarbij een plusteken (+) wordt gebruikt als veldscheidingsteken. Dit betekent dus dat er geen plustekens meer voorkomen in de inhoud van de velden zelf. Tijdstippen en reistijden hebben de volgende vorm `<uren>:<minuten>`, waarbij geen voorloophulden gebruikt worden. In het veld `<treinsoort>` worden de volgende afkortingen gebruikt: IC (InterCity), R (stoptrein), RE (regionale trein), TGV (hoge snelheidstrein), THA (Thalys). Verder bevat het bestand ook nog commentaarregels die beginnen met een hekje (#).

Gevraagd wordt om, gebruik makend van de teksteditor `vi`, een reeks substitutiecommando's te geven die achtereenvolgens de volgende opdrachten uitvoeren.

1. Zorg ervoor dat bij aankomst- en vertrektijden steeds twee cijfers gebruikt worden om het uur aan te geven. Voeg waar nodig een voorloophul toe.
2. Reorganiseer het bestand zodanig dat alle reisroutes geordend zijn volgens het tijdstip van vertrek. Zorg ervoor dat vroegste treinen bovenaan komen te staan. Het kan zijn dat je hiervoor meer dan één commando nodig hebt.
3. Voeg aan alle reisroutes een extra veld `<opmerkingen>` toe en vul daarin de tekst **Reserveren verplicht!!** voor alle routes waarbij een TGV moet genomen worden. Gebruik ook hier het plusteken (+) als veldscheidingsteken.
4. Splits de informatie omtrent aankomst en vertrek telkens over twee verschillende regels, waarbij de gegevens na omzetting volgende vorm aannemen:

```
vertrek <station vertrek> <uur vertrek> <treinsoort> <opmerkingen>
aankomst <station aankomst> <uur aankomst> <reistijd> <wissels>
```

In het eerste veld moet dus de letterlijke tekst `aankomst` of `vertrek` geplaatst worden. Na deze transformatie wordt het dollateken (\$) als veldscheidingsteken gebruikt.

5. Gebruik de Amerikaanse tijdsnotatie voor aankomst- en vertrektijden. In deze notatie wordt `06:37` weergegeven als `06:37AM`, en wordt `18:37` weergegeven als `06:37PM`.

Probeer zo weinig mogelijk commando's te geven en zo weinig mogelijk tekens te gebruiken in je commando's. De commentaarregels in `reisadvies.txt` mogen door je substituties niet gewijzigd worden. Alle wijzigingen moeten na elkaar uitgevoerd worden. Vermeld de gebruikte substitutiecommando's op je antwoordbestand.

Opgave 3

Gegeven zijn de volgende tekstfragmenten:

1. De tabel

Product	1	2	3	4	5
Prijs	€124	€136	€85	€156	€23
Garantie {jaar}	1	2	-	3	1
Score	89%	84%	51%		45%
Aanbevolen	ja	ja	neen	neen	neen

bevat een vergelijking van de verschillende types RAM geheugen.

2. De *Hessiaan* van een functie in meerdere veranderlijken $f(x_1, x_2, \dots, x_n)$ is de matrix van tweede-orde partiële afgeleiden van die functie, en is dus gedefinieerd als:

$$\nabla^2(x_1, x_2, \dots, x_n) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix} = (\text{kortweg}) \left(\frac{\partial^2 f}{\partial x_1 \partial x_2} \right).$$

De benaming *Hessiaan* verwijst naar de Duitse wiskundige Ludwig Otto Hesse en is naar verluidt geïntroduceerd door de Engelse wiskundige James Joseph Sylvester.

3. **Definitie 1** Als $z, f(z) \in \mathbb{R}^m$, dan is de M -de Frechet afgeleide van f , voorgesteld door $f^{(M)}(z)$, een operator op $\mathbb{R}^m \times \mathbb{R}^m \times \dots \times \mathbb{R}^m$ (M maal), die lineair is in elk van de operandi, met als waarde

$$f^{(M)}(z) \underbrace{(K_1, K_2, \dots, K_M)}_{\text{operandi}} = \sum_{i=1}^m \sum_{j_1=1}^m \sum_{j_2=1}^m \cdots \sum_{j_M=1}^m {}^i f_{j_1 j_2 \dots j_M} {}^{j_1} K_1 {}^{j_2} K_2 \dots {}^{j_M} K_M e_i, \quad (1)$$

waarbij z het argument is en

$$K_t = [{}^1 K_t, {}^2 K_t, \dots, {}^m K_t]^T \in \mathbb{R}^m, \quad t = 1, 2, \dots, M,$$

$${}^i f_{j_1 j_2 \dots j_M} = \frac{\partial^{M+i} f(z)}{\partial j_1 z \partial j_2 z \dots \partial j_M z},$$

$$e_i = [0, 0, \dots, 0, 1, 0, \dots, 0]^T \in \mathbb{R}^m,$$

met 1 op de i -de plaats in e_i .

Recursieve definitie 2 De elementaire differentiaal $F_s : \mathbb{R}^m \rightarrow \mathbb{R}^m$ van f en hun orde worden recursief gedefinieerd als:

(a) f is de enige elementaire differentiaal van de orde 1, en

(b) als F_s ($s = 1, 2, \dots, M$) respectievelijk elementaire differentiaal zijn van de orde r_s , dan is de Frechet afgeleide

$$f^{(M)}(F_1, F_2, \dots, F_M) \quad (2)$$

een elementaire differentiaal van de orde

$$1 + \sum_{s=1}^M r_s. \quad (3)$$

4. De gemiddelde intensiteit $J_\lambda(z)$ wordt gedefinieerd als het gemiddelde van de specifieke intensiteit $I_\lambda(z, \mu)$ over alle ruimtehoeken,

$$\begin{aligned} J_\lambda(z) &= \oint I_\lambda(z, \mu) \frac{d\Omega}{4\pi} \\ &= \frac{1}{4\pi} \int_0^{2\pi} d\varphi \int_0^\pi I_\lambda(z, \mu) \sin \theta d\theta, \end{aligned} \quad (4)$$

hetgeen zich herleidt tot

$$J_\lambda(z) = \frac{1}{2} \int_{-1}^1 I_\lambda(z, \mu) d\mu \quad (5)$$

aangezien de specifieke intensiteit axiaal symmetrisch verondersteld werd.

Gevraagd wordt om \LaTeX -code te genereren die precies hetzelfde resultaat oplevert als de tekstfragmenten binnen de bovenstaande kaders.

Opgave 4

Gevraagd wordt om een reeks `bash` shell scripts te schrijven die een overzicht in XHTML formaat genereren van alle IP adressen die een website bezocht hebben. Voor tekstmanipulaties die door de shell scripts worden uitgevoerd moet gebruik worden gemaakt van de line editor `ex`. Er mag onder geen beding gebruik worden gemaakt van `sed`, `awk` of controlestructuren in de shell scripts. Het is echter wel toegelaten om tijdelijke bestanden aan te laten maken door de shell scripts, maar zorg er telkens voor dat er geen tijdelijke bestanden in het bestandssysteem overblijven na uitvoering van de scripts. Werk hiervoor de volgende stappen af.

1. Schrijf een script `extractIP` dat een lijst genereert van alle unieke IP adressen die bestanden met extensie `.jsp` hebben geraadpleegd op een webserver. Deze lijst wordt opgemaakt door de IP adressen te filteren uit het logbestand van die webserver. Het logbestand wordt als argument doorgegeven aan het script. Combineer hiervoor de bestaande commando's `cut`, `grep`, `sort` en `uniq` zodanig dat het script slechts één enkele commandoregel bevat. Het bestand `webserver.log` kan hierbij als voorbeeld genomen worden van een logbestand. Onderstaande sessie toont de functionaliteit van het script aan voor het voorbeeld logbestand:

```
> ./extractIP webserver.log
128.178.164.154
134.67.6.14
65.55.208.130
65.55.208.133
65.55.208.134
90.230.91.142
97.66.4.91
```

2. Schrijf een shell script `IP2coordinates` dat voor een gegeven IP adres, dat als argument wordt doorgegeven aan het script, een aantal geografische gegevens van het IP adres uitschrijft naar standaard uitvoer. Voor het IP adres `128.178.164.154` worden bijvoorbeeld de volgende gegevens naar standaard uitvoer geschreven:

```
> ./IP2coordinates 128.178.164.154
128.178.164.154,Switzerland,CH (CHE),Vaud,Lausanne,,+41,6.6667,46.5333
```

Het script moet deze informatie extraheren door gebruik te maken van de webservice

`http://geoiptool.com/en/?IP=`

Op het einde van deze template URL komt het IP adres waarvoor informatie moet opgezocht worden. Voor het bovenstaande voorbeeld wordt de webservice dus op de volgende manier aangesproken:

`http://geoiptool.com/en/?IP=128.178.164.154`

Het script moet de geografische informatie extraheren uit de `html` broncode die door deze webservice wordt teruggegeven. De verschillende informatievelen worden op één enkele regel geplaatst, gescheiden door komma's. Overbodige `html`-tags moeten uit de informatievelen verwijderd worden, evenals spaties aan het begin en einde van de informatievelen.

- Schrijf een shell script `csv2html` dat een regel met informatievelen die door komma's van elkaar worden gescheiden omzet naar een rij uit een tabel in `html` formaat. Je mag hierbij uitgaan van het feit dat de informatievelen geen overbodige spaties voor- en achteraan bevatten. Het script moet werken zoals een filter. Het script `csv2html` zal dus regels met velden gescheiden door komma's inlezen van standaard uitvoer, en die als een rij uit een `html`-tabel wegschrijven naar standaard uitvoer. Een bestand `test.csv` dat de inhoud van `/IP2coordinates 128.178.164.154` bevat zal dus op de volgende manier worden getransformeerd:

```
> ./csv2html < test.csv
<tr><td>128.178.164.154</td><td>Switzerland</td><td>CH (CHE)</td><td>Vaud</td>
<td>Lausanne</td><td>&nbsp;</td><td>+41</td><td>6.6667</td><td>46.5333</td></tr>
```

Hierbij werd het resultaat enkel voor de overzichtelijkheid weergegeven over verschillende regels. In werkelijkheid wordt er slechts een regel weggeschreven naar standaard uitvoer. Let er ook op dat lege velden in het `csv` bestand worden opgevuld met ` ` in `html`.

- Gebruik de voorgaande shell scripts om een nieuw shell script `showVisitors` te implementeren dat uit een gegeven logbestand van een webserver alle IP adressen selecteert die minstens eenmaal een bestand met extensie `.jsp` hebben gedownload van de server, extra geografische informatie opzoekt over die IP adressen, en alle informatie overzichtelijk weergeeft in een webpagina. Het logbestand moet als argument aan het shell script `showVisitors` worden doorgegeven. Wanneer het script bijvoorbeeld wordt aangeroepen voor het voorbeeld logbestand

```
> ./showVisitors webserver.log
```

wordt er een overzicht van de geografische gegevens in tabelvorm weggeschreven naar het bestand `visitors.html`, en wordt onder de tabel aangegeven wanneer het rapport werd opgemaakt. Het resultaat moet er als volgt uitzien:

Bezoekers website								
128.178.164.154	Switzerland	CH (CHE)	Vaud	Lausanne		+41	6.6667	46.5333
134.67.6.14	United States	US (USA)	North Carolina	Research Triangle Park	27709	+1	-78.8489	35.9138
65.55.208.130	United States	US (USA)				+1	-97	38
65.55.208.133	United States	US (USA)				+1	-97	38
65.55.208.134	United States	US (USA)				+1	-97	38
90.230.91.142	Sweden	SE (SWE)				+46	15	62
97.66.4.91	United States	US (USA)				+1	-97	38

Gegevens gegenereerd op: Fri Sep 7 07:06:04 CEST 2007

Hint: Je kan gebruik maken van backquotes om commando-substitutie uit te voeren.