

---

---

## EXAMEN: Computergebruik

---

---

1<sup>e</sup> Bachelor Informatica  
prof. dr. Peter Dawyndt  
groep 1

dinsdag 17-08-2010, 8:30  
academiejaar 2009-2010  
tweede zittijd

### Opgave 1

Gegeven is een tekstbestand `tour2010.txt` waarin de einduitslag van de Tour de France 2010 werd opgenomen. Elke regel bevat volgende informatie over een bepaalde renner: *i*) plaats in einduitslag, *ii*) nationaliteit, *iii*) naam, *iv*) rugnummer, *v*) team, *vi*) totaal gereden tijd en *vii*) afstand in tijd ten opzichte van de winnaar. De verschillende informatievelen worden van elkaar gescheiden door tabs. Het bestand bevat bovendien een hoofding waarvan alle regels beginnen met een hekje (`#`). Gevraagd wordt om, gebruik makend van de teksteditor `vi` (of `vim`), een reeks substitutiecommando's op te stellen die achtereenvolgens de volgende opdrachten uitvoeren:

1. Vervang het veldscheidingsteken door een komma, en zorg ervoor dat alle spaties vooraan en achteraan elk veld verwijderd worden.
2. Zet alle tijdsaanduidingen (zowel de totaal gereden tijd als de afstand in tijd ten opzichte van de winnaar) om naar het formaat `HH:MI:SS`, waarbij de uren, minuten en seconden telkens met twee cijfers worden aangegeven. Het plusteken bij de afstand in tijd ten opzichte van de winnaar komt hierbij te vervallen. Zo moeten de regels

```
2.,LUX,SCHLECK Andy,11,TEAM SAXO BANK,91h 59' 27",+ 00' 39"  
140.,ESP,HERNANDEZ BLAZQUEZ Jesus,4,ASTANA,95h 30' 50",+ 3h 32' 02"
```

worden omgezet naar

```
2.,LUX,SCHLECK Andy,11,TEAM SAXO BANK,91:59:27,00:00:39  
140.,ESP,HERNANDEZ BLAZQUEZ Jesus,4,ASTANA,95:30:50,03:32:02
```

3. Splits de naam van de renners op in hun voor- en familienaam, en plaats deze in afzonderlijke velden. Let hierbij op het feit dat de startpositie van de voornaam kan herkend worden doordat de tweede letter ervan geen hoofdletter is. Familienamen worden immers volledig in hoofdletters geschreven. De twee regels uit het voorgaande voorbeeld moeten dus omgezet worden naar

```
2.,LUX,Andy,SCHLECK,11,TEAM SAXO BANK,91:59:27,00:00:39  
140.,ESP,Jesus,HERNANDEZ BLAZQUEZ,4,ASTANA,95:30:50,03:32:02
```

4. Duidt de kopmannen in het peloton aan door het punt achter hun plaats in de einduitslag te vervangen door de letter `k`. Kopmannen kunnen herkend worden doordat ze een rugnummer dragen dat eindigt op het cijfer 1. Van de twee regels uit het voorgaande voorbeeld, moet dus enkel de eerste regel aangepast worden. Dit levert het volgende resultaat op

```
2k,LUX,Andy,SCHLECK,11,TEAM SAXO BANK,91:59:27,00:00:39  
140.,ESP,Jesus,HERNANDEZ BLAZQUEZ,4,ASTANA,95:30:50,03:32:02
```

5. Herorganiseer het bestand zodanig dat de regels van de einduitslag alfabetisch worden gerangschikt op de teamnaam, en binnen elk team volgens stijgend volgnummer.

6. Plaats de inhoud van elk veld tussen dubbele aanhalingstekens. Zo moeten de twee voorbeeldregels worden omgezet naar

```
"2k","LUX","Andy","SCHLECK","11","TEAM SAXO BANK","91:59:27","00:00:39"  
"140. ","ESP","Jesus","HERNANDEZ BLAZQUEZ","4","ASTANA","95:30:50","03:32:02"
```

Probeer voor elke opdracht zo weinig mogelijk commando's te gebruiken en zorg ervoor dat elk van deze commando's bestaat uit zo weinig mogelijk tekens. De regels van de hoofding mogen door je commando's niet gewijzigd worden, zelfs niet als er bijkomende regels aan de hoofding worden toegevoegd. Alle wijzigingen moeten na elkaar uitgevoerd worden.

## Opgave 2

Gebruik filters, I/O redirection en pipes om telkens één enkel commando samen te stellen dat uitvoer genereert die voldoet aan onderstaande beschrijvingen.

1. De commando's `strace <commando>` (Linux) en `truss <commando>` (Solaris) voeren *commando* uit, en onderscheppen alle *system calls* die door het corresponderende proces worden aangeroepen. Als bijvoorbeeld het commando `truss date` wordt uitgevoerd op `genix`, dan wordt de volgende uitvoer gegenereerd (*ingekort*).

```
execve("/usr/bin/date", 0xFFBFFD94, 0xFFBFFD9C)  argc = 1
resolvepath("/usr/lib/ld.so.1", "/lib/ld.so.1", 1023) = 12
resolvepath("/usr/bin/date", "/usr/bin/date", 1023) = 13
stat("/usr/bin/date", 0xFFBFFB70)              = 0
open("/var/ld/ld.config", O_RDONLY)           Err#2 ENOENT
...
```

Op elke regel van de uitvoer gaat de naam van de aangeroepen system call het karakter ( vooraf. Vul één van de commandolijnen (afhankelijk van het systeem waarop je werkt)

```
strace date ...
truss date ...
```

verder aan, zodat een overzicht gegenereerd wordt van de vijf meest aangeroepen system calls. De system calls moeten in het overzicht worden opgelijst in dalende volgorde van het aantal aanroepen, en de naam van de system call moet telkens worden voorafgegaan door het aantal keer dat die werd aangeroepen. Zorg ervoor dat de uitvoer die door *commando* gegenereerd wordt niet wordt uitgeschreven. Het is niet toegelaten om gebruik te maken van de opties van de commando's `strace` of `truss`. Op `genix` moet bijvoorbeeld de volgende uitvoer gegenereerd worden.

```
8 mmap
6 brk
4 open
4 resolvepath
3 close
```

2. Als de inhoud van de *manual pages* van het commando `man` naar het bestand `out` wordt geschreven met behulp van het commando

```
man man > out
```

dan wordt deze inhoud opgemaakt met behulp van allerlei controlekarakters. Pas bovenstaande commandolijn aan, zodat de inhoud van de manual pages zonder controlekarakters naar het bestand `out` geschreven wordt.

3. Met het commando `df` kan informatie opgevraagd worden over de gemounte bestandssystemen. Op een voorbeeldsysteem hebben we met dit commando twee bestanden aangemaakt. De uitvoer van het commando `df -h` (vrije en gebruikte schijfruimte) werd weggeschreven naar het bestand `df_blocks.txt`. De uitvoer van het commando `df -ih` (informatie over inodes) werd weggeschreven naar het bestand `df_inodes.txt`. Het commando `df` kan zelf geen overzicht genereren waarin zowel informatie over schijfruimtegebruik als inodegebruik terug te vinden is. Geef een commando dat voor elk van de corresponderende regels van de twee gegeven bestanden de eerste vijf kolommen van het bestand `df_blocks.txt` en de kolommen twee, drie, vier en vijf van het bestand `df_inodes.txt` samenvoegt. In het finale overzicht moeten de kolommen van elkaar worden gescheiden door middel van tabs. Het eindresultaat moet er dus als volgt uitzien.

Filesystem	Size	Used	Avail	Use%	Inodes	IUsed	IFree	IUse%
/dev/sda1	4.7G	2.9G	1.7G	65%	611K	94K	518K	16%
/dev/sda6	91G	6.8G	80G	8%	12M	120K	12M	2%
/dev/sda3	1.9G	149M	1.7G	9%	245K	230	245K	1%
/dev/sda2	1.9G	323M	1.5G	18%	245K	823	244K	1%
/dev/home	5.5T	335G	5.2T	7%	5.7M	1.4M	4.4M	23%
/dev/scratch	27T	5.4T	21T	21%	27M	5.1M	22M	20%
/dev/sof2	1.9T	230G	1.6T	13%	2.0M	809K	1.2M	41%
/dev/data	30T	969G	29T	4%	30M	1.5M	28M	5%
/dev/sof1	1.9T	66G	1.8T	4%	2.0M	496K	1.5M	25%

4. Project Gutenberg is een project dat toelaat om gedigitaliseerde versies van boeken waarvoor het copyright is vervallen, gratis te downloaden. Gevraagd wordt om een eenvoudige spellingscontrole te ontwikkelen voor Project Gutenberg. Dit gebeurt in drie stappen, waarbij in elke stap één commando moet opgesteld worden dat een uitbreiding is van het commando uit de vorige stap.

- (a) Boeken zijn bij Project Gutenberg te vinden als tekstbestanden met de volgende vorm.

```
<sectie 1>
*** START OF THIS PROJECT GUTENBERG EBOOK ...
<sectie 2>
*** END OF THIS PROJECT GUTENBERG EBOOK ...
<sectie 3>
```

Geef een commando dat de tekst van het boek *Alice's Adventures in Wonderland* van Lewis Carroll afhaalt van de URL <http://www.gutenberg.org/files/11/11.txt>, en enkel de tweede sectie uitschrijft naar standaard uitvoer.

- (b) Geef een commando dat alle woorden van het boek *Alice's Adventures in Wonderland* alfabetisch gerangschikt oplijst (woorden worden van elkaar gescheiden door spaties). In deze lijst staat elk woord op een afzonderlijke regel en komt elk woord slechts één keer voor. Zorg ervoor dat hoofdletters worden omgezet naar kleine letters. Behoud enkel woorden die uitsluitend uit letters van het alfabet bestaan, en zorg ervoor dat de lijst geen lege regels bevat. De ingekorte uitvoer van het commando ziet er dan bijvoorbeeld als volgt uit.

```
a
abide
able
about
...
youth
zealand
zigzag
```

- (c) Breid het commando uit de vorige stap verder uit, zodat enkel die woorden weerhouden worden die niet voorkomen in het bestand `woordenlijst.txt`. Schrijf deze woorden genummerd uit, zoals hieronder wordt weergegeven.

```

1  arrum
2  chatte
3  dinn
4  hjckrrh
5  lefthand
6  morcar
7  ootiful
8  rosetree
9  seaography
10 stigand

```

**Opmerking:** De uitvoer van de eerste twee stappen van de spellingscontrole is terug te vinden in de bestanden `opgave2_4a.out` en `opgave2_4b.out`. Je kunt deze bestanden gebruiken indien je er (nog) niet in geslaagd bent om de eerste commando's voor de spellingscontrole op te stellen.

## Opgave 3

Code 39 is een techniek voor het opstellen van barcodes die in 1974 door Dr. David Allais and Ray Stevens werd ontwikkeld voor het bedrijf Intermec. Bij code 39 wordt elk letterteken voorgesteld door vijf verticale zwarte stroken, die van elkaar worden gescheiden door vier verticale witte stroken. Barcodes van opeenvolgende lettertekens worden steeds van elkaar gescheiden door een smalle witte strook. Onderstaande figuur illustreert hoe het woord `COMPUTERGEBRUIK` eruitziet als code 39 barcode.



Code 39 barcodes worden in ASCII-formaat gecodeerd door vier symbolen te gebruiken die de vier combinaties van smalle/brede zwarte/witte banden voorstellen. De twee formaten  $\Sigma_1$  en  $\Sigma_2$  (zie onderstaande tabel) die hiervoor gebruikt worden, zijn niets anders dan twee verschillende manieren om dezelfde onderliggende codering te omschrijven. Het gaat dus niet om twee varianten van code 39. Zo kan het letterteken A worden voorgesteld door de tekenreeks `BsSsSbSsB` of door de tekenreeks `ZwzWzWzWZ`. Ook een hybride voorstelling die gebruik maakt van een combinatie van de symbolen uit  $\Sigma_1$  en  $\Sigma_2$  is toegelaten. De tekenreeks `ZsZsZsZbZsZ` is dus ook een geldige code 39 voorstelling van het letterteken A. Bij de voorstelling van code 39 barcodes in ASCII-formaat wordt geen symbool opgenomen dat de scheiding van lettertekens aangeeft.



STROOK	$\Sigma_1$	$\Sigma_2$
breed-zwart	B	Z
smal-zwart	S	z
breed-wit	b	W
smal-wit	s	w

Het bestand `barcodes.txt` bevat op elke regel de code 39 barcode in ASCII-formaat van een woord, gevolgd door een spatie en het woord zelf. Sommige ASCII-barcodes zijn ongeldig omdat hun aantal stroken geen veelvoud is van negen, of omdat ze symbolen bevatten die niet in de formaten  $\Sigma_1$  of  $\Sigma_2$  voorkomen (waardoor ze niet tot  $\mathcal{B}$  behoren; zie hieronder). Gevraagd wordt:

1. Bepaal reguliere expressies voor elk van de onderstaande verzamelingen, waarbij  $\mathcal{B}$  de verzameling van alle tekenreeksen voorstelt waarvan de symbolen voorkomen in  $\Sigma_1$  of  $\Sigma_2$ , en waarvan het aantal symbolen een veelvoud is van negen. Probeer deze reguliere expressies zo kort mogelijk te houden.

- (a)  $\alpha = \{b \in \mathcal{B} \mid \text{aantal smalle witte banden in } b \text{ is oneven}\}$
- (b)  $\beta = \{b \in \mathcal{B} \mid b \text{ bevat minstens tien opeenvolgende stroken met dezelfde breedte}\}$
- (c)  $\gamma = \{b \in \mathcal{B} \mid \text{in } b \text{ wordt elke letter uitsluitend vertaald door symbolen uit } \Sigma_1 \text{ of } \Sigma_2\}$   
 voorbeelden:  $ZwzwzWzwZ \mid BsSsSbSsB \mid zwZwzwWZ \mid SsBsSbSsB \in \gamma$   
 $Zw\underline{Ss}S\underline{W}S\underline{S}Z \mid zw\underline{Bs}S\underline{b}z\underline{s}Z \mid \underline{Bs}S\underline{sz}W\underline{z}w\underline{B} \mid \underline{B}w\underline{Bs}z\underline{W}S\underline{w}Z \notin \gamma$
- (d)  $\delta = \{b \in \mathcal{B} \mid \text{voor en na een letterovergang in } b \text{ staat hetzelfde symbool}\}$   
 voorbeelden:  $ZwzwzWzwZ \mid ZwZwZwzwz \mid zwZwWZwz \mid zwzwZwzwZ \in \delta$   
 $BsSwzbSw\underline{Z} \mid \underline{B}szsSbSw\underline{Z} \mid \underline{B}sSwzsZbS \mid SwzsZbzwB \notin \delta$

**Opmerking:** In bovenstaande voorbeelden worden de letterovergangen met een verticale streep (*l*) aangegeven. Dit dient enkel ter verduidelijking van de voorbeelden. In de barcodes worden letterovergangen niet expliciet aangegeven.

Gebruik een commando uit de `grep` familie om enkel die regels van het bestand `barcodes.txt` te selecteren die barcodes bevatten die behoren tot de opgegeven verzameling. Vermeld in je antwoordbestand voor elke verzameling het gebruikte selectiecommando, en geef ook aan hoeveel regels je gevonden hebt.

2. Beschouw de verzamelingen  $\alpha$ ,  $\beta$ ,  $\gamma$  en  $\delta$  zoals hierboven gedefinieerd. Gebruik nu deze verzamelingen om op de volgende manier een boodschap bestaande uit vier woorden te achterhalen:

- (a) het eerste woord staat op de unieke regel met een barcode uit de verzameling  $\alpha \cap \beta$
- (b) het tweede woord staat op de unieke regel met een barcode uit de verzameling  $\beta \cap \gamma$
- (c) het derde woord staat op de unieke regel met een barcode uit de verzameling  $\gamma \cap \delta$
- (d) het vierde woord staat op de unieke regel met een barcode uit de verzameling  $\delta \cap \alpha$

Vermeld in je antwoordbestand de gevonden woorden, samen met het `unix` commando (of de commandosequentie) dat je gebruikt hebt om elk van deze woorden te vinden. Elk commando of elke commandosequentie moet dus als resultaat één van de gezochte woorden naar standaard uitvoer schrijven (zonder de barcode die het woord voorafgaat).

## Opgave 4

1. Geef  $\text{\LaTeX}$ -code die een reconstructie maakt van onderstaande tabel. Zorg er daarbij voor dat de opmaak zo getrouw mogelijk behouden blijft.

Het tekenverloop van  $f(x) = \cos(x) \cdot [2 \cdot \sin(x) - 1]$  (de periode is  $2\pi$ ):

$x$	$x$ - waarden					
	0	$\frac{\pi}{6}$	$\frac{\pi}{2}$	$\frac{5\pi}{6}$	$\frac{3\pi}{2}$	$(2\pi)$
$\cos(x)$	+	+	+	0	-	+
$2 \cdot \sin(x) - 1$	-	-	0	+	+	-
<b>f(x)</b>	-	-	<b>0</b>	<b>0</b>	-	<b>0</b>
						(-)

De oplossing is dus  $x \in [0, \frac{\pi}{6}[ \cup ]\frac{\pi}{2}, \frac{5\pi}{6}[ \cup ]\frac{3\pi}{2}, 2\pi[ + k \cdot 2\pi \leftrightarrow x \in ] - \frac{\pi}{2}, \frac{\pi}{6}[ \cup ]\frac{\pi}{2}, \frac{5\pi}{6}[ + k \cdot 2\pi$ .

2. Geef  $\text{\LaTeX}$ -code die precies hetzelfde resultaat oplevert als het tekstfragment in onderstaand kader. Zorg er daarbij voor dat formules en eigen omgevingen automatisch genummerd worden, en gebruik waar mogelijk verwijzingen naar deze nummeringen. De opdracht in de eerste zin van het tekstfragment moet je ook toepassen in je  $\text{\LaTeX}$ .

Maak eerst een macro voor  $\left( \begin{array}{c} a \\ b \end{array} \middle| c \right)$ , waarbij  $a$ ,  $b$  en  $c$  veranderlijken zijn. Gebruik die daarna om onderstaande tekst te reproduceren.

Dit gedeelte is gewijd aan de wiskundige berekening van de inproducten tussen de vectoren  $u_0(E)$ ,  $u_1(E)$ ,  $v(x)$  en  $w(p)$ . Twee bestaande formules zullen hierbij van cruciaal belang blijken te zijn:

**Formule 1** *De eerste formule zegt dat*

$$\begin{aligned} \sum_{n=0}^{\infty} \frac{(a)_n}{n!} F_1 \left( \begin{array}{c} -n, b \\ a \end{array} \middle| y \right) F_1 \left( \begin{array}{c} -n \\ a \end{array} \middle| x \right) t^n \\ = (1-t)^{b-a} (1-t+yt)^{-b} e^{\frac{xt}{t-1}} F_1 \left( \begin{array}{c} b \\ a \end{array} \middle| \frac{xyt}{(1-t)(1-t+yt)} \right), \end{aligned}$$

waarbij  $|t| < 1$ .

**Formule 2** *De tweede formule is de volgende benadering:*

$$\lim_{z \rightarrow \infty} F_1 \left( \begin{array}{c} c \\ a \end{array} \middle| z \right) = \lim_{z \rightarrow \infty} \frac{\Gamma(a)}{\Gamma(c)} e^z z^{c-a} \sum_{n=0}^{\infty} \frac{(a-c)_n (1-c)_n}{n!} z^{-n}$$

De limiet voor  $t \rightarrow 1$  kan gevonden worden met behulp van Formule 2. Na vereenvoudiging krijgen we

$$\begin{aligned} \langle v(x), u_0(E) \rangle &= \mathcal{A}_0(E) \mathcal{B}_0(x) \frac{\Gamma(a)}{\Gamma(\frac{a+iE}{2})} \lim_{t \rightarrow 1} e^{\frac{x^2 t}{1+t}} (|x| \sqrt{2t})^{-a+iE} (1+t)^{-iE} \\ &= \frac{|\Gamma(\frac{a+iE}{2})|}{\Gamma(\frac{a+iE}{2})} \frac{|x|^{iE-\frac{1}{2}}}{\sqrt{\pi} 2^{iE+2}} \\ &= \mathcal{C}_0(E) \frac{|x|^{iE-\frac{1}{2}}}{2\sqrt{\pi}}, \end{aligned}$$

waarbij  $|\mathcal{C}_0(E)|^2 = 1$ .

Plaats een PDF bestand met daarin de gecompileerde  $\text{\LaTeX}$  fragmenten in het ZIP-bestand dat je indient via Indianio.