OPLOSSING EXAMENS COMPUTERGEBRUIK 2006 🡪 2014 ( Jarre Knockaert )

Merk op dat oefeningen op Latex, ex en enkele oefeningen waarvan bestanden missen niet gemaakt werden. Kritieke blik op antwoorden vereist voor gebruik van dit document.   
DO NOT TRY THIS AT HOME

[jaar: 2006-2007, zittijd: 1](http://wina.ugent.be/data/examenvragen/108/Computergebruik__2006-2007__Examen_(1e_zit).pdf) (4 versies van examen)

Opgave 1 (ex1) (Enkel werkend op eigen vbn, verder niet getest  
(a) A = {x B | de eerste bit van x is verschillend van de laatste bit}  
(b) B = {x B | het aantal nullen in x is een veelvoud van 5}  
(c) C = {x B | enen komen in x steeds in paren voor}  
a) egrep ‘-v ^(.).\*\1$’  
b) egrep ‘^1\*(01\*01\*01\*01\*01\*)\*$’  
c) egrep ‘^(0\*(11)\*0\*)\*$’

Opgave 1 (ex2) (Enkel werkend op eigen vbn, verder niet getest)  
(a) A = {x 2 B | het aantal nullen in x is oneven}  
(b) B = {x 2 B | in x komen maximaal twee opeenvolgende enen voor}  
(c) C = {x 2 B | de eerste en laatste bit van x zijn 0, en x bevat geen opeenvolgende nullen}  
a) egrep –v '^1\*(01\*01\*)\*$'  
b) egrep -v '111'  
c) egrep '^0(1+01\*)\*0$'

Opgave 1 (ex3) (Enkel werkend op eigen vbn, verder niet getest)

(a) A = {x 2 B | het aantal nullen in x is even}

(b) B = {x 2 B | de reeks 01 komt in x evenveel voor als de reeks 10}

(c) C = {x 2 B | x bevat geen opeenvolgende enen}

a) egrep '^1\*(01\*01\*)\*$'  
b) egrep '^((0+1+)\*0+|(1+0+)\*1+)[^01]'  
c) egrep -v "11"  
  
Opgave 1 (ex4) (Enkel werkend op eigen vbn, verder niet getest)

(a) A = {x 2 B | de laatste drie bits van x zijn gelijk}

(b) B = {x 2 B | het aantal nullen in x is deelbaar door 3}

merk op: nul is deelbaar door 3

(c) C = {x 2 B | als 000 een substring is van x, dan is n0(x) = 3} ,

waarbij n0(x) het aantal nullen in x voorstelt

a) egrep ‘(.)\1\1$

b) egrep "^1\*(01\*01\*01\*)\*$"

c) egrep "^1\*(0001+)\*(000)?$"

Opgave 4 (ex1)

1. a)   
   i) tail -n +$1  
   ii) sed 's/./&\n/g'  
   iii) tr '[A-Z]' '[a-z]'
2. a)  
   i) paste $@ -   
   ii) paste -d "" $@ | sort | uniq -c | sort –nr | head -26

Opgave 4 (ex2)

1. a)   
   i) tr ‘[a-z]’ ‘[A-Z]’  
   ii) tail –n +$1  
   iii) sed 's/./&\n/g'
2. b)  
   i) paste $@ -   
   ii) paste -d "" $@ | sort | uniq -c | sort –nr | head -26

Opgave 4 (ex3)

1. a)   
   i) sed ‘s/[^A-z]//g’  
   ii) tr ‘[A-Z]’ ‘[a-z]’  
   iii) sed 's/./&\n/g'
2. a)  
   i) paste $@ -   
   ii) paste -d "" $@ | sort | uniq -c | sort –nr | head -26

Opgave 4 (ex4)

1. a)   
   i) sed 's/[^A-z]\*//g'  
   ii) sed 's/./&\n/g'  
   iii) sort -u
2. b)   
   i) paste - $@  
   ii)paste -d "" $@ | sort | uniq -c | sort –nr | head -26

[jaar: 2006-2007, zittijd: 2](http://wina.ugent.be/data/examenvragen/108/Ex_Computergebruik_06-07_2ezit.pdf)

Opgave 1

1. egrep ‘^s’
2. egrep "ap.?t"
3. egrep "aff?g.?k"
4. egrep '[.!?][^A-z]? [A-Z]'

[jaar: 2007-2008, zittijd: 1](http://wina.ugent.be/data/examenvragen/108/Ex_Computergebruik_07-08.pdf)

Opgave 1 (ex1) (Enkel werkend op eigen vbn, verder niet getest)

(a) A = {d ∈ D | eerste en laatste drie basen van d zijn dezelfde, maar volgorde is omgekeerd}  
(b) B = {d ∈ D | geen enkel kwintet van basen in d is hetzelfde}  
(c) C = {d ∈ D | gedegenereerde sequentie BADMAN matcht juist ´e´en keer in d}  
a) egrep ‘^[a-z]+ (.)(.)(.)[AGCT]\*\3\2\1 ’   
b) egrep -v ‘(.....)[AGCT]\*\1’  
c) egrep -v '([GTC]A[GAT][AC]A[AGCT])[AGCT]\*\1' | egrep '^[AGCT]\* [GTC]A[GAT][AC]A[AGCT] [AGCT]\* '

Opgave 1 (ex2) (Enkel werkend op eigen vbn, verder niet getest)  
(a) A = {d bevat triplet van basen dat minstens drie keer na elkaar herhaald wordt}  
(b) B = {aantal opeenvolgende basen C of T in d is altijd even}  
(c) C = {gedegenereerde DNA sequentie AGKYANBTD matcht juist ´e´en keer in d}  
a) egrep '^[AGCT]\*([AGCT])\1\1[AGCT]\* '  
b) egrep '^([AG]\*(CC)\*|(TT)\*[AG]\*)\* '  
c) egrep -v '(AG[GT][TC]A[AGCT][GTC]T[GAT])[AGCT]\*\1' | egrep '^[AGCT]\*AG[GT][TC]A[AGCT][GTC]T[GAT][AGCT]\* '

[jaar: 2007-2008, zittijd: 2](http://wina.ugent.be/data/examenvragen/108/Ex_Computergebruik_07-08_2ezit.pdf)

Opgave 1

1. A = {w ∈ W | w telt ten minste vijf lettertekens}  
2. B = {w ∈ W | letters van w staan in alfabetische volgorde}  
3. C = {w ∈ W | w heeft geen letters gemeenschappelijk met de tekenreeks ”klmnopqr”}  
4. D = {w ∈ W | w bevat juist twee dubbele letters, niet noodzakelijk opeenvolgend}

cat /usr/share/dict/words | …

1. egrep '[A-z]{5,}'
2. egrep'^a\*?b\*?c\*?d\*?e\*?f\*?g\*?h\*?i\*?j\*?k\*?l\*?m\*?n\*?o\*?p\*?q\*?r\*?s\*?t\*?u\*?v\*?w\*?x\*?y\*?z\*?$'
3. egrep '^[^klmnopqr]\*$'
4. egrep '(.)\1.\*(.)\2' | egrep -v '(.)\1.\*(.)\2.\*(.)\3'

Opgave 3

1. a) tr ' ' '+' | sed 's/[0-9]/&\*&/g' | sed 's/.\*/echo $((&))/' | bash | xargs echo som-kwadranten:  
   c) history | cut -d " " -f5 | uniq -c | sort -nr | head -5
2. a) cat ~/.bash\_history | sed 's/ \*| \*/\n/g' | cut -d" " -f1 | sort | uniq -c | sort -nr | head -5 | sed -r 's/^ \*([0-9]\*) (.\*)$/\2 (\1)/' | tr '\n' ',' | sed 's/,/, /g' | sed 's/, \*$//'; echo

[jaar: 2008-2009, zittijd: 1](http://wina.ugent.be/data/examenvragen/108/Ex_Computergebruik_08-09_1ezit.pdf)

Opgave 1 (ex1)(Enkel werkend op eigen vbn, verder niet getest)

(a) A = { geen enkel symbool komt meer dan twee keer na elkaar voor in m}  
(b) B = { eerste en laatste twee symbolen van m zijn gelijk, maar omgekeerde volgorde}  
(c) C = { op elke even positie van m komt een punt voor}  
(d) D = { m bevat minimaal drie en maximaal vijf koppeltekens}  
a) egrep -v '(\.\.\.)|(---)'  
b) egrep ' (.)(.).\*\2\1$'  
c) egrep ' ([.-]\.)\*-?$'  
d) egrep ' (\.\*-\.\*){3,5}$'

Opgave 1 (ex2)(Enkel werkend op eigen vbn, verder niet getest)

(a) α = { m bevat een oneven aantal koppeltekens}  
(b) β = { m bestaat uit een alternerende reeks van symbolen}  
(c) γ = { eerste helft van m is gelijk aan tweede helft}  
(d) δ = { twee gelijke symbolen worden in m nooit gevolgd door twee keer het andere symbool}  
  
a) egrep -v '^(\.\*-\.\*-\.\*)\* '  
b) egrep '^\.?(-\.)\*-? '  
c) egrep '^(.\*)\1 '  
d) egrep -v '(--\.\.|\.\.--)'

[jaar: 2008-2009, zittijd: 2](http://wina.ugent.be/data/examenvragen/108/Ex_Computergebruik_08-09_2ezit.pdf)

Opgave 1 (Enkel werkend op eigen vbn, verder niet getest)

(a) A= { geen enkel natuurlijk getal in i bestaat enkel uit twee cijfers}  
(b) B= { eerste en derde natuurlijk getal in i zijn gelijk}  
(c) C= { som van aantal vieren en aantal vijven in i is even en groter dan nul}  
(d) D= { i bevat minstens 3 zessen maar i bevat geen opeenvolgende zessen}  
  
a) egrep -v '(^|\.)[0-9][0-9](\.| )'  
b) egrep '^([0-9])[^0-9]\*[0-9][^0-9]\*\1.\*'  
c) egrep '^([^54]\*[54][^54]\*[54][^54]\*)\*$'  
d) egrep '(^([^6]\*6[^6]+){3,} )|(^(^6]+6[^6]\*){3,} )'

[jaar: 2009-2010, zittijd: 1](http://wina.ugent.be/data/examenvragen/108/Ex_Computergebruik_09-10_1ezit.pdf)

Opgave 1 (ex1) (Enkel werkend op eigen vbn, verder niet getest)

(a) A = {alle directorynamen van p zijn even lang en bevatten 3, 4 of 5 karakters}  
(b) B = {bestaat uit drie directorynamen, waarbij het laatste karakter van elke directorynaam gelijk is aan het eerste karakter van de volgende directory- of bestandsnaam}  
(c) C = {geen enkele directorynaam van p is juist 4 karakters lang}  
(d) D = {elke directorynaam van p bestaat enkel uit letters of enkel uit cijfers}  
  
a) egrep '^/?(.../)+[^/]+$|^/?(..../+)[^/]+$|^/?(...../)+[^/]+$'  
b) egrep '^/?([^/]\*([^/])/\2){3}[^/]+$'  
c) egrep -v '^.\*/[A-z1-9]{4}/(.\*/)?[^/]+$'  
d) egrep -v '^.\*/[^/]\*([A-z][1-9]|[1-9][A-z])[^/]\*/(.\*/)?[^/]+$'

Opgave 1 (ex2) (Enkel werkend op eigen vbn, verder niet getest)

(a) A = {p is relatieve padnaam, waarbij elke directorynaam exact twee cijfers bevat}  
(b) B = {er bestaat een paar opeenvolgende karakters dat in minstens drie opeenvolgende directory- of bestandsnamen voorkomt binnen p}  
(c) C = {elke directorynaam van p is alternerende reeks van cijfers en letters}  
(d) D = {laatste karakter van elke directorynaam in p is verschillend van eerste karakter  
van volgende directory- of bestandsnaam}

a) egrep "^/?([a-z]\*[0-9][a-z]\*[0-9][a-z]\*/)\*[^/]\*$"   
b) egrep "^.\*/[^/]\*([^/][^/])[^/]\*/([^/]\*\1[^/]\*/){2,}(.\*/)?[^/]\*$"  
c) egrep "^.\*/?[1-9]?([a-z][0-9])\*[a-z]?/[^/]\*$"  
d) egrep –v “^/?([^/]\*([^/])/\2){3}[^/]+$”

Opgave 4 (ex1) (Vb: <http://docs.oracle.com/cd/E19626-01/820-0386/aaqds/index.html>)

1. a) cat pam.conf | sed -r '/sufficient|required|requisite|binding/!d' | cut -d" " -f6 | sort | uniq -c   
   b) cat pam.conf | sed -r '/sufficient|required|requisite|binding/!d' | cut -d" " -f6 | sort | uniq -c | sort –nr
2. a) perl -e 'for ($i=1;$i<10;$i++) { print "$i\n"; }' | tr '\n' ' ' | sed -r 's/([369]) /\1\n/g'  
   b) perl -e 'for ($i=1;$i<10;$i++) { print "$i\n"; }' | tr '\n' ' ' | sed -r 's/([369]) /\1\n/g' | sed -r 's/^([0-9]) ([0-9]) ([0-9])/\1+\2\*\3/' | sed 's/.\*/echo $((&))/' | bash  
   c) perl -e 'for ($i=1;$i<10;$i++) { print "$i\n"; }' | tr '\n' ' ' | sed -r 's/([369]) /\1\n/g' | sed -r 's/^([0-9]) ([0-9]) ([0-9])/\1+\2\*\3/' | sed 's/.\*/echo $((&))/' | bash | tr '\n' '+' | sed -r 's/(.\*)\+$/echo $((\1))/' | bash

Opgave 4 (ex2)

1. a) cat /etc/services | egrep -o "/[a-z]\*" | tr -d "/" | sort | uniq -c | sort -n | tail -2  
   b) cat /etc/services | egrep -o "/[a-z]\*" | tr -d "/" | sort | uniq -c | sort -n | tail -2 |tr 'a-z' 'A-Z'
2. a) perl -e 'for ($k=9;$k>0;$k--) { print "$k\n"; }' | tr '\n' ' ' | sed -r 's/([147]) /\1\n/g'  
   b) perl -e 'for ($k=9;$k>0;$k--) { print "$k\n"; }' | tr '\n' ' ' | sed -r 's/([147]) /\1\n/g' | sed -r 's/^([0-9]) ([0-9]) ([0-9])/\1+\2-\3/' | sed 's/.\*/echo $((&))/' | bash  
   c) perl -e 'for ($k=9;$k>0;$k--) { print "$k\n"; }' | tr '\n' ' ' | sed -r 's/([147]) /\1\n/g' | sed -r 's/^([0-9]) ([0-9]) ([0-9])/\1+\2-\3/' | sed 's/.\*/echo $((&))/' | bash | tr '\n' '\*' | sed -r 's/(.\*)\\*$/echo $((\1))/' | bash

[jaar: 2009-2010, zittijd: 2](http://wina.ugent.be/data/examenvragen/108/Ex_Computergebruik_09-10_2ezit.pdf)

Opgave 2

1. strace date 2>&1 | cut -d"(" -f1 | sort | uniq -c | sort –nr
2. man man | cat | tr -d "[:cntrl:]" > out
3. paste <(df -h 2> /dev/null | egrep -o ".\*%") <(df -ih 2> /dev/null | egrep -o "[^ ]+ +[^ ]+ +[^ ]+ +[^ ]+%")
4. a) curl -s <http://www.gutenberg.org/files/11/11.txt> | sed -n '/START OF/,/END OF/p'  
   b) sed 's/\ /\n/g' 11.txt|sed -r 's/^.\*[^a-zA-Z].+$//g'|sed 's/[^a-zA-Z]//g'|sed -r '/^\s\*$/d'| sed 's/\(.\*\)/\L\1/' | sort | uniq  
   c) comm -23 <(sed 's/\ /\n/g' 11.txt|sed -r 's/^.\*[^a-zA-Z].+$//g'|sed 's/[^a-zA-Z]//g'|sed -r '/^\s\*$/d'| sed 's/\(.\*\)/\L\1/' | sort | uniq) woordenlijst.txt | nl

Opgave 3  
(Ik begrijp de opgave zelf niet volledig. Dus dit is de oplossing van iemand anders i.f.v. volledigheid.)  
a)egrep '^([BSbsZWzw]{9})\* ' barcodes.txt |   
 egrep -c '^[^sw]\*[sw]([^sw]\*[sw][^sw]\*[sw])\*[^sw]\* ‘  
b) egrep '^([BSbsZWzw]{9})\* ' barcodes.txt |   
 egrep -c '^.\*([BZbW]{10}|[Szsw]{10}).\* '  
c)egrep -c '^([BSbs]{9}|[ZWzw]{9})\* ' barcodes.txt  
d) egrep '^([BSbsZWzw]{9})\* ' barcodes.txt | egrep -c '^.(.{7}(.)\2)\*.{8} '

[jaar: 2010-2011, zittijd: 1](http://wina.ugent.be/data/examenvragen/108/Ex_Computergebruik_10-11_1ezit.pdf)

Opgave 2 (ex1)

1. for i in {1..42}; do echo $i $RANDOM; done | sort -R | cut -d" " -f1 | head -6 | tr '\n' '-' | sed 's/\(.\*\)./\1/'; echo
2. cat beurs.txt | sed -r 's/^([^:]\*:)([^:]\*):([^:]\*)$/echo "\1$\3: $(echo \3-\2 | bc -l)" /' | tr ',' '.' | bash | tr '.' ',' | sed -r 's/^([^:]\*),([^:]\*)/\1.\2/g' | sed 's/-,/-0,/g'   
   (Voor minder correct oplossing kan je na eerste bash stoppen)
3. a) find dir1/ -type f -exec openssl dgst -md5 {} \; 2> /dev/null | sed -r 's#.\*/([^/]\*)\)= (.\*)$#\1 \2#' | sort > md5dir1.txt  
   b) find dir2 -type f-exec openssl dgst -md5 {} \; 2> /dev/null | sed -r 's#.\*/([^/]\*)\)= (.\*)$#\1 \2#' | sort > md5dir2.txt  
   c) comm -23 md5dir1.txt md5dir2.txt  
   d) comm -23 <(find dir1/ -type f -exec openssl dgst -md5 {} \; 2> /dev/null | sed -r 's#.\*/([^/]\*)\)= (.\*)$#\1 \2#' | sort) <(find dir2 -type f-exec openssl dgst -md5 {} \; 2> /dev/null | sed -r 's#.\*/([^/]\*)\)= (.\*)$#\1 \2#')

Een betere oplossing: cat file2 | cut -d' ' -f2 | sed -n '1{x};1!H;${x;s/[^\n]\*/grep -v &/g;s/\n/ | /g;p;}' | sed 's/^/cat file1 | /g' | bash

Opgave 2 (ex2)

1. find /usr/bin/ -type f -exec egrep '^#!' {} \; | sed 's/#! \*/#!/g' | sed -r 's/([^ ]\*) .\*/\1/' | sort | uniq -c | sort -nr | head -5
2. cat BMI.txt | sed -r 's#^([^:]\*:)([0-9]+)[^A-z0-9] ([0-9]\*)[^A-z0-9]:([0-9]\*)$#echo \"\1$(echo "scale=2;\4/2.20462262185" | bc ):$(echo "scale=2;0.0254\*(12\*\2+\3)" | bc | egrep -o ".\...")"#' | bash | sed -r 's/(.\*):(.\*):(.\*)$/echo "\1:\2:\3:$(echo "scale=2;\2\/\3^2"|bc)"/' | bash
3. a) cat /dev/urandom | head -10000 | strings | tr 'A-Z' 'a-z' | sort > willekeurige\_tekst.txt  
   b) mkdir tmp; curl -s http://users.ugent.be/~pdawyndt/woordenboek.txt 2> /dev/null > ~/tmp/engelse\_woorden.txt  
   c) comm -12 willekeurige\_tekst.txt ~/tmp/engelse\_woorden.txt  
   d) comm -12 <(cat /dev/urandom | head -10000 | strings | tr 'A-Z' 'a-z' | sort) <(curl -s http://users.ugent.be/~pdawyndt/woordenboek.txt 2> /dev/null)

Opgave 3 (ex1) (Enkel werkend op eigen vbn, verder niet getest)

(a) A = {g ∈ G | eerste gen van g is voorwaarts en laatste gen van g is achterwaarts}  
(b) B = {g ∈ G | g heeft minstens zes opeenvolgende genen met dezelfde orientatie}  
(c) C = {g ∈ G | alle genen in g hebben minimaal lengte 2}  
(d) D = {g ∈ G | alle reeksen gelijktekens in g zijn even lang}  
a) egrep "^-\*=\*>.\*<=\*-\* "  
b) egrep "(=\*>){6,}|(<=\*){6,}"   
c) egrep -v "(<[^=]|[^=]>)"  
d) egrep "^[^=]\*([=]\*)([^=]+\1[^=]\*)\*\1? "

Opgave 3 (ex2) (Enkel werkend op eigen vbn, verder niet getest)

(a) α = {g ∈ G | eerste en laatste gen van g hebben dezelfde orientatie}  
(b) β = {g ∈ G | g bevat geen opeenvolgende genen met dezelfde orientatie}  
(c) γ = {g ∈ G | alle genen in g worden steeds gescheiden door een niet-coderend   
(d) δ = {g ∈ G | opeenvolgende genen met tegengestelde orientatie worden in g steeds  
gescheiden door een niet-coderend fragment met lengte minstens drie}

a) egrep "^[-|=]\*([<>]).\*\1[-|=]\* "  
b) egrep "^[-|=]\*>?([-=|]\*<[-|=]\*>[-|=]\*)\*<?[-|=]\* "  
c) egrep "^[-=|]\*([<>]=\*-+)?(=\*[<>]=\*-+)\*[-=|]\*([<>]=\*)?[-=|]\* "  
d) egrep -v "(<=\*-{0,2}=\*>|>=\*-{0,2}<=\*)"

[jaar: 2010-2011, zittijd: 2](http://wina.ugent.be/data/examenvragen/108/examen_CG_20110824VM.pdf)

Opgave 2 (Enkel werkend op eigen vbn, verder niet getest)

(a) A = { derde karakter van h is gelijk aan derde laatste karakter van h}  
(b) B = { h bevat vier opeenvolgende cijfers, die bestaan uit twee paren gelijke cijfers}  
(c) C = { elke letter komt maximaal tweemaal voor in h}  
(d) D = { aantal cijfers in h is een veelvoud van vier}  
a) egrep "^..(.).\*\1.. "  
b) egrep "(([0-9])\2([0-9])\3|([0-9])([0-9])\4\5|([0-9])([0-9])\7\6)"  
c) egrep -v "([A-Z]).\*\1.\*\1.\* "  
d) egrep "^(([A-F]\*[0-9][A-F]\*){4})\* "

Opgave 3

1. history | cut -d" " -f4- | sed -r 's/ \*\| \*/\n/g' | sed -r '/^ \*[A-z1-9]+ .\*/!d' | sed -r 's/ \*([^ ]\*) .\*/\1/' | sort | uniq -c | sort -n | tail -10
2. wget -qO- http://www.python.org/download/ | egrep "release-number" | sed -r 's/^.\*Python (.\*)<\/a>.\*$/\1/'
3. a) mcdfasta (alias mcdfasta="mkdir fasta; cd fasta")  
   b) Uitdaging: zonder shell script?

**###############**k=0

while read line

do

i=$((i+1))

if [ $(echo "$line" | egrep -c '^>') -eq 1 ]; then

sed -n "${i},/^>/p" multi.fasta > fasta0$k

if [ $(tail -1 fasta0$k | egrep -c '^>') -eq 1 ]; then

sed -i '$d' fasta0$k

fi

k=$((k+1))

fi

done < multi.fasta

**###############**  
c)  
**###############**  
if [ $# -ne 1 ]; then

echo "Syntax: $0: file1.fasta"

exit 1

fi

if [ ! -r $1 ]; then

echo "$1 is not readable"

exit 1

fi

acc=`cat $1 | head -1 | cut -d"|" -f2 | tr -d " "`

mv "$1" "$acc.fasta"  
**###############**  
d) find -regex ".\*\.fasta$" -exec ./hernoem.sh '{}' \; 2> /dev/null

[examen\_CG\_20120109NM](http://minerva.ugent.be/main/document/document.php?curdirpath=%2Fexamenvragen%2Fexamen_CG_20120109NM&cidReq=C00093902014)

Opgave 1

1. Egrep ‘(^.\*(.{9,}).\*\.\2.\.\*$)|(^.\*\.(.\*{9,})\.\4\..\*$)’
2. Egrep ‘(^|\.)([^.]\*)\.\2\.\2(\.|$)’
3. Egrep -v '(((<+[^<]+){2})+((>+[^>]+){2})+)+ '  
   (Fout antwoord!)
4. Egrep –v ‘^.{6}[^+]’

Opgave 3:

1. %s/\n/;/g | %s/;\([0-9]\*\);/\r\1;/g
2. %s/^\(.\*;.\*;[A-z .]\*\)\/.\*\(\(;.\*\)\{7\}\)$/\1\2/
3. :%s/^\([0-9]\*\);\(.\*\);\(.\*\);\(.\*\)$/\1;\3;\2;\4  
   :%s/^\([0-9]\*\);\(.\*\);\(.\*\);\(.\*\)$/\1;\3;\2;\4  
   Dit kan in één commando

P.S. procent vergeten in voorlaatste veld: :%s/^\(.\*;\)\(.\*\)\(;.\*\)$/\1\2%\3

1. Zie opl in Survival kit
2. …

Opgave 4:

1.a)  
i) /  
ii)   
#!/bin/bash  
read invoer  
echo $invoer | sed 's/\(.\)/\1\n/g' | sed '/^[ ]\*$/d'  
iii)   
#!/bin/bash  
read invoer  
echo $invoer | sed -r 's/(.\*)/\L\1/g'  
 b)  
i)

[examen\_CG\_20120109NM](http://minerva.ugent.be/main/document/document.php?curdirpath=%2Fexamenvragen%2Fexamen_CG_20120109NM&cidReq=C00093902014)

Opgave 1 - Regex

1. Egrep ‘^…(.).\*\1… ’
2. egrep '^.\*\[(.{10}).\*\]\1.\*$'
3. egrep -v '+[^+-]- '
4. egrep -v '<([^<>][^<>])+>'

Opgave 2 – Vim

1. %s/^\([^ ]\*\).\*\(\[.\*\]\) \(".\*"\) \([0-9]\*\) \([0-9]\*\)$/\1\\*\2\\*\3\\*\4\\*\5
2. %s/^\(.\*\)"\(.\*\) \(.\*\) .\*"\(\\*.\*\)$/\1\2\\*\3\4
3. %s/^\(.\*\)\\*\[\(.\*\/.\*\/.\*\):\(.\*:.\*:.\*\) .\*\]\\*\(.\*\)\\*\(.\*\)\\*\([0-9]\*\)\\*\([0-9]\*\)$/\5\\*\4\\*\6\\*\7\\*\2 \3\\*\1
4. g/\\*[^\*]\*crawl[^\*]\*$/d  
   g/\\*[^\*]\*bot[^\*]\*$/d
5. ! cut -d"\*" -f6 % | uniq -c | sort –n

Opgave 4 – Shell

1. i) tr ‘a-z’ ‘A-Z’  
   ii)   
   (( $1 - 1 ))  
   tail -$1  
   iii) zie vorig examen, via sed of fold –w1
2. Zie vorige examen

[examen\_CG\_20120820VM](http://minerva.ugent.be/courses2014/C00093902014/document/examenvragen/examen_CG_20120820VM/examen_CG_20120820VM.pdf?cidReq=C00093902014)

Opgave 1

Oef2

1. A) echo {b,z}{b,z}{b,z}{b,z}{b,z}{b,z}{b,z}{b,z}{b,z}{b,z} | tr ' ' '\n' | sed '/\(b\*z\)\{5,\}\|\(z\*b\)\{7,\}/d' | wc –l  
   B) echo {b,z}{b,z}{b,z}{b,z}{b,z}{b,z}{b,z}{b,z}{b,z}{b,z} | tr ' ' '\n' | sed '/\(b\*z\)\{5,\}\|\(z\*b\)\{7,\}/d' | grep -v 'zz' | wc -l
2. A) cat oneven.txt | tr '\n' '+' | sed -r 's/([0-9]\*)\+([0-9]\*)\+/(4\/\1)\-(4\/\2)\+/g' | sed -r "s/^(.\*)\+$/scale=10;\1\n/" | bc  
   B) seq 1 2 999 | tr '\n' '+' | sed -r 's/([0-9]\*)\+([0-9]\*)\+/(4\/\1)\-(4\/\2)\+/g' | sed -r "s/^(.\*)\+$/scale=10;\1\n/" | bc
3. find $1 -name "\*.$2" | sed "s/^.\*$/cat '&' | wc -l/" | bash | tr '\n' '+'| sed -r 's/^(.\*)\+$/echo "\$\(\(\1\)\)"/' | bash
4. cat raven.txt | tr -s ' ' '\n' | tr 'A-Z' 'a-z'| sed -r '/^[^A-z]\*(.{3,})[^A-z]\*$/!d' | sed 's/^[^A-z]\*//g' | sed 's/[^A-z]\*$//g' | comm -2 -- - stopwoorden.txt | sort | uniq -c | sort –n

oef1

1. cat oneven.txt | tr '\n' '+' | sed -r 's/([0-9]\*)\+([0-9]\*)\+/(4\/\1)\-(4\/\2)\+/g' | sed -r "s/^(.\*)\+$/scale=10;\1\n/" | bc
2. seq 1 2 999

Oef3

1. zie vorig ex

Oef4

1. cat stopwoorden.txt | sort; cat raven.txt | sed 's/ /\n/g' | tr 'A-Z' 'a-z' | sed 's/[^a-z]\*$//g' | sed 's/^[^a-z]\*//g' | egrep '.{3,}' | sort | uniq | comm -23 -- - <(sort stopwoorden.txt) | …

Opgave 2

1. %s/\(Women's.\*\|Men's.\*\)\t/\1, /g
2. %s/ \*\t \*/;/g

%s/Team GB/Great Britain/g

%s/Team USA/United States of America/g

1. %s/^\(.\)\([^;]\*\);\(.\)\([^;]\*\)/\3\L\4\E \1\L\2
2. :g!/^[^;]\*;[^;]\*;[^;]\*;\([^eauio]\*[eauio][^eauoi]\*[eauoi][^eauoi]\*\)\*;/d
3. :% !sort -t';' -k2 -n -k3 –n

Opgave 3

1. cat hexdump.txt | egrep '^(([0-9][a-f]) |([a-f][0-9]) )\*\|'
2. cat hexdump.txt | egrep -o '^.(.) \1((.) \3){14}. |'
3. Geen idee
4. cat hexdump.txt | egrep -v '.\*([a-fA-F]).\*\|.\*\1.\*'

examen\_CG\_20130109NM

Opgave 1

1. Zie oef
2. cp $(find testdir/ -type d) $(find testdir/ -type d | sed -r 's/(\/[^\/]\*)\_([\/]\*)$/\1\2/g')

probleem hier is dat hij vaak errors toont aangezien bepaalde mappen niet meer gevonden worden door substitutie

1. echo "$1" | sed "s/./&\^$2\+/g" | sed -r "s/(.\*)\+/echo '\1' | bc/ " | bash
2. Geen idee

Opgave 2

/

Opgave 3   
a) egrep '^[^3579]\*(([3579]) ?\2[^3579]\*)\* ?[a-z]+' multitap.txt  
b) egrep -v '3 ?5|3 ?7|3 ?9|5 ?3|5 ?7|5 ?9|7 ?3|7 ?5|7 ?9|9 ?3|9 ?5|9 ?7|2 ?4|2 ?6|2 ?8|4 ?2|4 ?6|4 ?8|6 ?2|6 ?4|6 ?8|8 ?2|8 ?4|8 ?6' multitap.txt  
c) egrep -o '(^3 ?[^3 ])|(.\*[^3 ] ?3 ?[^3 ].\*)|([^3 ] ?3 [a-z]+)'  
d) egrep -v '(([2-9]) ?\2 ?\2 ?\2)|(([2-9]) ?\4 ?\4.\*([2-9]) ?\5)|(([2-9]) ?\7.\*([2-9]) ?\8 ?\8)' | egrep -v '([2-9])\ ?\1.\*([2-9]) ?\2.\*([2-9])\ ?\3'

A en B: come  
B en C: as  
C en D: you  
D en A: are

a) egrep '^([^3579]+([3579 ][3579 ])+[^3579]+)+ .\*$' multitap.txt (2709 regels) b) egrep '^(( \*[0-9]) \*\2 \*| \*[02468] \*[13579] \*| \*[13579] \*[02468] \*)+ [a-z]\*$' multitap.txt (1749 regels)

Opgave 4

Opgave 5

[examen\_CG\_20130906VM](http://minerva.ugent.be/courses2014/C00093902014/document/examenvragen/examen_CG_20130906VM/examen_CG_20130906VM.pdf?cidReq=C00093902014)

Opgave 1

1. Stud1 : sed -r 's/ \*([0-9]\*) \*([A-z]\*) \*/\L\2 (\1)/'

Stud2 : sed -r 's/.\* ([A-z]\*)/\1/' | sort |uniq -c| sort -nr| head -4  
Stud3 : cut -d " " -f1 | sed 's/./&\n/g' | sed '/^$/d' | sort | uniq -c | sort -rn | head -3

Stud4 : sed -r "s/^ \*([a-z.]\*) \*([0-9]\*) \*$/sed -n '\2p' \1/g" | bash

Stud5 : xargs echo "Regel $(cat invoer.txt | cut -d" " -f2) van $(cat invoer.txt | cut -d" " -f1):"

Stud6 : sed -n "$(cat invoer.txt | cut -d' ' -f2),\$p" studentenlijst.txt | xargs echo "Regels $(cat invoer.txt | cut -d' ' -f2) van studentenlijst.txt" | sed -r 's/([0-9]+)/\n\1/g'

Opgave 2

/

Opgave 3

1. cat morse.txt | egrep -v '^(\.\*\-\.\*\-\.\*)\* \('
2. cat morse.txt | egrep '^\-?(\.\-)\*\.? \('
3. cat morse.txt | egrep '^(.\*)\1 \('
4. cat morse.txt | egrep -v '(\.\.\-\-)|(\-\-\.\.)'

A && B: durf  
B && C: denken  
C && D: denken  
A && D: ?

Opgave 4

[examen\_CG\_20140108VM](http://minerva.ugent.be/courses2014/C00093902014/document/examenvragen/examen_CG_20140108VM/examen_CG_20140108VM.pdf?cidReq=C00093902014)

#!/bin/bash

seq -s "" $1 | sed -r 's/([0-9])([0-9])/\1-\2+/g' | sed 's/.\*/echo $((&))/' | bash

[examen\_CG\_20140108NM](http://minerva.ugent.be/courses2014/C00093902014/document/examenvragen/examen_CG_20140108NM/examen_CG_20140108NM.pdf?cidReq=C00093902014)

opgave 1

3) uitdrukking=$(echo $@ | sed 's/\\//g' | sed 's/ //g' | sed 's/x/ x /g' | sed 's/\+/ \+ /g')

berekening=$(echo $uitdrukking | tr '|' '1' | sed -r 's/1/1\+/g' | tr 'x' '\*' | sed 's/\+$//' | sed 's/1+ /1 /g' | sed 's/1[^+]/1) /g' | sed 's/ 1/(1/g' | sed 's/^1/(1/' | sed 's/1$/1)/')

n=$(echo $berekening | bc)

m=$(echo $berekening | tr '\*' '+' | bc)

echo "$uitdrukking = $n ($m tandenstokers)"

4)

files=`find $1 -regex .\*\.$2`

for file in $files

do

filename=`echo $file`

cat $file | egrep -n "$3"

done

EXAMENS SCRIPTINGTALEN

Examen\_ST\_20100621NM

Opgave 3

rekensom

#!/bin/bash

####FUNCTIES####

max () {

oldlength=0

for line in $@

do

length=`echo $line | wc -c`

if [ $length -gt $oldlength ]; then

oldlength=`echo $(($length-1))`

fi

done

echo $oldlength

}

########

opmaak () {

max=`max $1 $2 $3`

echo -n " "; printf "%\*s\n" $max "$1"

echo -n "+"; printf "%\*s\n" $max "$2"

while [ $max -gt -1 ]; do

echo -n "-"

max=`expr $max - 1`

done

echo

echo -n "="; printf "%\*s\n" $max "$3"

}

########

controleer\_som () {

if [ $# -lt 2 ]; then

echo "Minstens 2 argumenten moeten meegegeven worden"

fi

for num in $@

do

sommatie=`expr $sum1 + $num`

echo $sommatie

done

if [ $(($sommatie-${!#})) -eq ${!#} ]; then

echo "Correct som!"

else

echo "Foutieve som!"

fi

}

}############################

#####CONTROLE ARG###########

############################

usage () { echo "Syntax: $0: ARG1 ARG2 ARG3 CODE"; exit 1; }

if [ ! $# -eq 4 ]; then

usage

fi

arg1=$(echo $1 | egrep -c "^[A-Z]+$")

arg2=$(echo $2 | egrep -c "^[A-Z]+$")

arg3=$(echo $3 | egrep -c "^[A-Z]+$")

if [ ! $arg1 = 1 -o ! $arg2 -eq 1 -o ! $arg3 -eq 1 ]; then

usage

fi

tmp=`echo "$1$2$3" | sed 's/[A-z]/&\n/g' | sort | uniq | tr -d '\n'`

if [ ! $(echo $tmp |wc -c) -eq $(echo $4 | wc -c) ]; then

usage

fi

num1=`echo "$1" | tr "$tmp" "$4"`

num2=`echo "$2" | tr "$tmp" "$4"`

sum1=`echo "$(($num1+$num2))"`

sum2=`echo "$3" | tr "$tmp" "$4"`

#PROGRAMMA

opmaak $1 $2 $3

opmaak $num1 $num2 $sum1

controleer\_som $num1 $num2 $sum2

Examen\_ST\_20080820VM

Opgave 2

som.sh

!/bin/bash

syntax () { echo "Syntax: $0: dir naam extensie regel"; exit 1; }

if [ $# -ne 4 ]; then

syntax

fi

for file in $(find $1 -name \*.$3 | egrep "\/[^/]\*$2[^/]\*$"); do

sum=$sum`cat $file | sed -n "$4p" | tr '\n' '+'`

done

echo $sum|sed -r 's/(.\*)\+$/echo "$((\1))"/' | bash

Product.sh

#!/bin/bash

syntax () { echo "Syntax: $0: file.zip [naam extensie regel]"; exit 1; }

if [ $# -lt 4 ]; then

syntax

fi

if [ $(echo $1 | egrep -c ".zip$") -ne 1 ]; then

syntax

fi

product=1

unzip -q $1

dir=`ls | grep "$1" | sed -r 's/(.\*)\.zip/\1/'`

shift

while [ $# -gt 0 ]; do

if [ $# -lt 3 ]; then

num=1

else

num=`./som.sh $dir $1 $2 $3`

fi

shift 3

product=`echo "$(($product \* $num))"`

done

echo $product

rm -r $dir

* [jaar: 2006-2007, zittijd: 1](http://wina.ugent.be/data/examenvragen/119/Ex_Scriptingtalen_06-07_1ezit.pdf)

Opgave 2

insertCitaat

#!/bin/bash

if [ -z $(ls ~ | egrep '^citaten.txt$') ]; then

touch ~/citaten.txt

fi

echo "Geef de auteur van het citaat:"

read auteur

echo $auteur >> ~/citaten.txt

echo "Geef de tekst van het citaar:"

read input

while [ "$input" != "%%" ]

do

read input

echo $input >> ~/citaten.txt

done

selectCitaat

#!/bin/bash

match=0

match2=`expr $1 - 1`

count=1

while [ $match -lt $match2 ]; do

if [ "$(sed -n "${count}p" citaten.txt)" = "%%" ]; then

match=$((match+1))

fi

count=$((count+1))

done

until [ "$(sed -n "${count}p" citaten.txt)" = "%%" ]; do

sed -n "${count}p" citaten.txt

count=$((count+1))

done

* [jaar: 2006-2007, zittijd: 2](http://wina.ugent.be/data/examenvragen/119/Ex_Scriptingtalen_06-07_2ezit.pdf)

Opgave 2

VERSIE 1 ( zonder files )

#!/bin/bash

if [ $# -ne 1 ]; then

echo "Usage: $0: uid"

exit 1

fi

getent passwd $1 | sed -r 's/^[^:]\*:[^:]\*:[^:]\*:([^:]\*):([^:]\*):([^:]\*):([^:]\*)/\2\n\3\n\4\n\1/'

!/bin/bash

if [ $# -ne 1 ]; then

echo "Usage: $0: uid"

exit 1

fi

getent passwd $1 | sed -r 's/^[^:]\*:[^:]\*:[^:]\*:([^:]\*):([^:]\*):([^:]\*):([^:]\*)/\2\n\3\n\4\n\1/'

VERSIE 2 ( met files )

<http://www.cyberciti.biz/faq/understanding-etcpasswd-file-format/> (handige site)

#!/bin/bash

if [ $# -ne 1 ]; then

echo "Usage: $0: uid"

exit 1

fi

if [ ! -r /etc/passwd ]; then

echo "/etc/passwd is not readable"

exit 1

fi

cat /etc/passwd | egrep "^([^:]\*):[^:]\*:$1:([^:]\*):[^:]\*:([^:]\*):([^:]\*)$" | sed -r "s/^([^:]\*):[^:]\*:$1:([^:]\*):[^:]\*:([^:]\*):([^:]\*)$/\1\n\4\n\3\n\2/"