
Computergebruik: EXAMEN

1^o Bachelor Informatica
prof. dr. Peter Dawyndt
groep 1

donderdag 09-01-2020, 08:30
academiejaar 2019-2020
eerste zittijd

Opgave 1 (TOP500)

(10 pt)

Het TOP500 project maakt een rangschikking op van de 500 krachtigste niet-gedistribueerde computersystemen ter wereld, en brengt ook zoveel mogelijk details van hun architectuur in kaart. Het project werd gestart in 1993 en publiceert twee keer per jaar een bijgewerkte lijst van supercomputers. De eerste publicatie valt altijd samen met de Internationale Supercomputing-conferentie in juni. De tweede wordt gepresenteerd op de ACM/IEEE Supercomputing-conferentie in november. Het project wil een betrouwbare basis bieden voor het opvolgen en detecteren van trends op het vlak van high-performance computing en baseert haar ranglijsten op HPL, een implementatie van de high-performance LINPACK benchmark geschreven in de programmeertaal Fortran voor computers met gedistribueerd geheugen.



De lijst wordt momenteel gedomineerd door China dat met 229 supercomputers in de lijst staat, op verre afstand gevolgd door de Verenigde Staten dat de tweede plek bezet met 121 supercomputers. Sinds juni 2018 is de Summit van Oak Ridge National Laboratory (Verenigde Staten) 's werelds krachtigste supercomputer met 143.5 petaFLOPS op de LINPACK-benchmarks.

Opgave

Het tekstbestand TOP500.txt bevat de 54^e editie van de TOP500 zoals die gepubliceerd werd in november 2019. De eerste regel van het bestand bevat een hoofding. Elke volgende regel bevat informatie over één supercomputer, bestaande uit de volgende negen velden die van elkaar worden gescheiden door een puntkomma (;): *i*) fabrikant (Manufacturer), *ii*) naam (Name), *iii*) land (Country), *iv*) aantal rekenkernen (Total Cores), *v*) maximaal aantal *floating point* bewerkingen per seconde, uitgedrukt in teraFLOPS per seconde (Rmax (TFlop/s)), *vi*) vermogen, uitgedrukt in kilowatt (Power (kW)), *vii*) besturingssysteem (Operating System), *viii*) soort processor (Processor Generation) en *ix*) processorsnelheid, uitgedrukt in megahertz (Processor Speed (MHz)). De velden bevatten zelf nooit puntkomma's (;).

Gevraagd wordt om — gebruikmakend van de teksteditors vi of vim — een reeks commando's op te stellen die achtereenvolgens de volgende opdrachten uitvoeren. Probeer voor elke opdracht zo weinig mogelijk commando's te gebruiken en zorg er voor dat elk van deze commando's bestaat uit zo weinig mogelijk tekens. Alle opdrachten moeten na elkaar uitgevoerd worden. De opdrachten mogen de eerste regel niet wijzigen, tenzij dit expliciet anders vermeld staat. Ter controle kan je gebruikmaken van de meegeleverde bestanden TOP500.i.txt (ZIP), die telkens de inhoud van het bestand bevatten nadat de *i*-de opdracht werd uitgevoerd.

1. Verwijder tekst die tussen ronde haakjes staat (inclusief de ronde haakjes zelf). Verwijder alle spaties aan het begin en het einde van een veld, en verwijder ook alle spaties voor en achter een slash (/). Vervang twee of meer opeenvolgende spaties door één enkele spatie. Zo moet

```

1 | Manufacturer;Name;Country;Total Cores;Rmax (TFlop/s);Power (kW);Operating System;Processor Generation;Processor Speed (MHz)
2 | IBM;Summit;United States;2414592;148600;10096;RHEL 7.4;IBM POWER9;3070
3 | IBM / NVIDIA / Mellanox;Sierra;United States;1572480;94640;7438,28;Red Hat Enterprise Linux;IBM POWER9;3100
4 | NRCPC;Sunway TaihuLight;China;10649600;93014,59388;15371;Sunway RaiseOS 2.0.5;Sunway;1450
5 | NUDT;Tianhe-2A;China;4981760;61444,5;18482;Kylin Linux;Intel Xeon E5 (IvyBridge);2200
6 | Dell EMC;Frontera;United States;448448;23516,4;;CentOS Linux 7;Xeon Platinum 82xx (Cascade Lake);2700
7 | Cray/HPE;Piz Daint;Switzerland;387872;21230,2384,24;Cray Linux Environment ;Intel Xeon E5 (Haswell);2600
8 | Cray/HPE;Trinity;United States;979072;20158,7;7578,1;Cray Linux Environment ;Intel Xeon Phi;1400
9 | ...
10 | Huawei;;China;41440;1142,232;577,69;Linux;Intel Xeon E5 (Broadwell);2400
11 | Inspur;;China;27520;1142;;CentOS;Xeon Platinum;2100
12 | Inspur;;China;27520;1142;;CentOS;Xeon Platinum;2100
13 | Inspur;;China;27520;1142;;CentOS;Xeon Platinum;2100

```

bijvoorbeeld omgezet worden naar (TOP500.1.txt)

```

1 | Manufacturer;Name;Country;Total Cores;Rmax (TFlop/s);Power (kW);Operating System;Processor Generation;Processor Speed (MHz)
2 | IBM;Summit;United States;2414592;148600;10096;RHEL 7.4;IBM POWER9;3070
3 | IBM/NVIDIA/Mellanox;Sierra;United States;1572480;94640;7438,28;Red Hat Enterprise Linux;IBM POWER9;3100
4 | NRCPC;Sunway TaihuLight;China;10649600;93014,59388;15371;Sunway RaiseOS 2.0.5;Sunway;1450
5 | NUDT;Tianhe-2A;China;4981760;61444,5;18482;Kylin Linux;Intel Xeon E5;2200
6 | Dell EMC;Frontera;United States;448448;23516,4;;CentOS Linux 7;Xeon Platinum 82xx;2700
7 | Cray/HPE;Piz Daint;Switzerland;387872;21230,2384,24;Cray Linux Environment;Intel Xeon E5;2600
8 | Cray/HPE;Trinity;United States;979072;20158,7;7578,1;Cray Linux Environment;Intel Xeon Phi;1400
9 | ...
10 | Huawei;;China;41440;1142,232;577,69;Linux;Intel Xeon E5;2400
11 | Inspur;;China;27520;1142;;CentOS;Xeon Platinum;2100
12 | Inspur;;China;27520;1142;;CentOS;Xeon Platinum;2100
13 | Inspur;;China;27520;1142;;CentOS;Xeon Platinum;2100

```

2. Voeg vooraan een extra veld Rank toe met het volgnummer van de supercomputers in de TOP500 lijst (waarbij ook de hoofding moet bijgewerkt worden). Daarbij worden de computers opeenvolgend genummerd vanaf 1. Er zijn geen ex-aequo's. We krijgen dan (TOP500.2.txt)

```

1 | Rank;Manufacturer;Name;Country;Total Cores;Rmax (TFlop/s);Power (kW);Operating System;Processor Generation;Processor Speed (MHz)
2 | 1;IBM;Summit;United States;2414592;148600;10096;RHEL 7.4;IBM POWER9;3070
3 | 2;IBM/NVIDIA/Mellanox;Sierra;United States;1572480;94640;7438,28;Red Hat Enterprise Linux;IBM POWER9;3100
4 | 3;NRCPC;Sunway TaihuLight;China;10649600;93014,59388;15371;Sunway RaiseOS 2.0.5;Sunway;1450
5 | 4;NUDT;Tianhe-2A;China;4981760;61444,5;18482;Kylin Linux;Intel Xeon E5;2200
6 | 5;Dell EMC;Frontera;United States;448448;23516,4;;CentOS Linux 7;Xeon Platinum 82xx;2700
7 | 6;Cray/HPE;Piz Daint;Switzerland;387872;21230,2384,24;Cray Linux Environment;Intel Xeon E5;2600
8 | 7;Cray/HPE;Trinity;United States;979072;20158,7;7578,1;Cray Linux Environment;Intel Xeon Phi;1400
9 | ...
10 | 497;Huawei;;China;41440;1142,232;577,69;Linux;Intel Xeon E5;2400
11 | 498;Inspur;;China;27520;1142;;CentOS;Xeon Platinum;2100
12 | 499;Inspur;;China;27520;1142;;CentOS;Xeon Platinum;2100
13 | 500;Inspur;;China;27520;1142;;CentOS;Xeon Platinum;2100

```

3. Zet een koppelteken (-) in het veld met de naam (Name) voor alle computers die geen naam hebben (d.i. waarvoor het veld met de naam leeg is). Alle computers in de TOP500 gebruiken één of andere Linux-distributie als besturingssysteem. Plaats bij alle computers Linux in het veld met het besturingssysteem (Operating System). We krijgen dan (TOP500.3.txt)

```

1 | Rank;Manufacturer;Name;Country;Total Cores;Rmax (TFlop/s);Power (kW);Operating System;Processor Generation;Processor Speed (MHz)
2 | 1;IBM;Summit;United States;2414592;148600;10096;Linux;IBM POWER9;3070
3 | 2;IBM/NVIDIA/Mellanox;Sierra;United States;1572480;94640;7438,28;Linux;IBM POWER9;3100
4 | 3;NRCPC;Sunway TaihuLight;China;10649600;93014,59388;15371;Linux;Sunway;1450
5 | 4;NUDT;Tianhe-2A;China;4981760;61444,5;18482;Linux;Intel Xeon E5;2200
6 | 5;Dell EMC;Frontera;United States;448448;23516,4;;Linux;Xeon Platinum 82xx;2700
7 | 6;Cray/HPE;Piz Daint;Switzerland;387872;21230,2384,24;Linux;Intel Xeon E5;2600
8 | 7;Cray/HPE;Trinity;United States;979072;20158,7;7578,1;Linux;Intel Xeon Phi;1400
9 | ...
10 | 497;Huawei;-;China;41440;1142,232;577,69;Linux;Intel Xeon E5;2400
11 | 498;Inspur;-;China;27520;1142;;Linux;Xeon Platinum;2100
12 | 499;Inspur;-;China;27520;1142;;Linux;Xeon Platinum;2100
13 | 500;Inspur;-;China;27520;1142;;Linux;Xeon Platinum;2100

```

4. Laat alle cijfers na de komma weg (inclusief de komma zelf) in alle velden waarvan de inhoud bestaat uit een kommagetal (d.i. waarvan de inhoud het formaat dd,dd heeft). We krijgen dan (TOP500.4.txt)

```

1 | Rank;Manufacturer;Name;Country;Total Cores;Rmax (TFlop/s);Power (kW);Operating System;Processor Generation;Processor Speed (MHz)
2 | 1;IBM;Summit;United States;2414592;148600;10096;Linux;IBM POWER9;3070
3 | 2;IBM/NVIDIA/Mellanox;Sierra;United States;1572480;94640;7438;Linux;IBM POWER9;3100
4 | 3;NRCPC;Sunway TaihuLight;China;10649600;93014;15371;Linux;Sunway;1450
5 | 4;NUDT;Tianhe-2A;China;4981760;61444;18482;Linux;Intel Xeon E5;2200
6 | 5;Dell EMC;Frontera;United States;448448;23516;;Linux;Xeon Platinum 82xx;2700

```

```

7 | 6;Cray/HPE;Piz Daint;Switzerland;387872;21230;2384;Linux;Intel Xeon E5;2600
8 | 7;Cray/HPE;Trinity;United States;979072;20158;7578;Linux;Intel Xeon Phi;1400
9 | ...
10 | 497;Huawei;-;China;41440;1142;577;Linux;Intel Xeon E5;2400
11 | 498;Inspur;-;China;27520;1142;;Linux;Xeon Platinum;2100
12 | 499;Inspur;-;China;27520;1142;;Linux;Xeon Platinum;2100
13 | 500;Inspur;-;China;27520;1142;;Linux;Xeon Platinum;2100

```

5. Voeg de laatste twee velden (Processor Generation en Processor Speed (MHz)) samen tot één enkel veld Processor. De nieuw inhoud van het veld wordt de inhoud van het eerste veld, gevolgd door een spatie en tussen ronde haakjes de inhoud van het tweede veld en de tekst GHz. Daarbij moet de processorsnelheid omgezet worden van megahertz naar gigahertz (1GHz = 1000MHz), waarbij er mag van uitgegaan worden dat alle waarden tussen 1000 en 4000 megahertz liggen. De waarde in gigahertz wordt genoteerd in het formaat d.dd, waarbij nullen op het einde van de decimale cijfers weggelaten worden maar er altijd minstens één decimaal cijfer moet overblijven (ook al is dat een nul). We krijgen dan (TOP500.5.txt)

```

1 | Rank;Manufacturer;Name;Country;Total Cores;Rmax (TFlop/s);Power (kW);Operating System;Processor
2 | 1;IBM;Summit;United States;2414592;148600;10096;Linux;IBM POWER9 (3.07GHz)
3 | 2;IBM/NVIDIA/Mellanox;Sierra;United States;1572480;94640;7438;Linux;IBM POWER9 (3.1GHz)
4 | 3;NRCPC;Sunway TaihuLight;China;10649600;93014;16371;Linux;Sunway (1.45GHz)
5 | 4;NUDT;Tianhe-2A;China;4981760;61444;18482;Linux;Intel Xeon E5 (2.2GHz)
6 | 5;Dell EMC;Frontara;United States;448448;23516;;Linux;Xeon Platinum 82xx (2.7GHz)
7 | 6;Cray/HPE;Piz Daint;Switzerland;387872;21230;2384;Linux;Intel Xeon E5 (2.6GHz)
8 | 7;Cray/HPE;Trinity;United States;979072;20158;7578;Linux;Intel Xeon Phi (1.4GHz)
9 | ...
10 | 48;HPE;Eagle;United States;75600;4850;;Linux;Xeon Gold (3.0GHz)
11 | ...
12 | 497;Huawei;-;China;41440;1142;577;Linux;Intel Xeon E5 (2.4GHz)
13 | 498;Inspur;-;China;27520;1142;;Linux;Xeon Platinum (2.1GHz)
14 | 499;Inspur;-;China;27520;1142;;Linux;Xeon Platinum (2.1GHz)
15 | 500;Inspur;-;China;27520;1142;;Linux;Xeon Platinum (2.1GHz)

```

Opgave 2 (PRIMEGAME)

(10 pt)

Hier is iets fantastisch — een machine gemaakt van breuken:

$$\frac{17}{91}, \frac{78}{85}, \frac{19}{51}, \frac{23}{38}, \frac{29}{33}, \frac{77}{29}, \frac{95}{23}, \frac{77}{19}, \frac{1}{17}, \frac{11}{13}, \frac{13}{11}, \frac{15}{14}, \frac{15}{2}, \frac{55}{1}$$

Begin met het getal 2 als *seed*. Vermenigvuldig de *seed* telkens met de volgende breuk, todat het product een geheel getal is (in dit geval is dit bij vermenigvuldiging met de breuk $\frac{15}{2}$). Neem nu dit geheel getal (15) als de nieuwe *seed*, en vermenigvuldig die *seed* opnieuw met elk van de breuken, todat je opnieuw een geheel getal bekomt. Blijf deze procedure herhalen, en houd bij wanneer de *seed* een macht van twee is.

STAP	SEED	$\log_2(\text{SEED})$
19	4	2
69	8	3
280	32	5
707	128	7
2363	2048	11
3876	8192	13
8068	131072	17
11319	524288	19
19201	8388608	23
36866	536870912	29
45551	2147483648	31
75224	137438953472	37
101112	2199023255552	41
117831	8796093022208	43
152025	140737488355328	47

De eerste tweede macht ($2^2 = 4$) verschijnt na 19 stappen. Vijftig stappen later duikt $2^3 = 8$ op. Daarna verschijnt $2^5 = 32$ ongeveer 200 stappen verderop. Er ontstaat een patroon: de exponenten zijn 2, 3, 5, 7, 11, 13, ...

Zo blijkt dat "alleen al deze veertien breuken een oneindig aantal priemgetallen bevatten, zelfs die waar nog niemand iets van afweet" schrijft Dominic Olivastro. "Er is iets enorm magisch aan." Deze PRIMEGAME techniek is gebaseerd op een Turing-complete esotherische programmeertaal genaamd FRACTRAN, die bedacht werd door John Horton Conway.

Opgave

Elke regel van het tekstbestand `fractions.txt` bestaat uit een patroon $p \in \mathcal{P}$, gevolgd door een spatie en een woord $w \in \mathcal{W}$. De verzameling \mathcal{P} bevat alle mogelijke reeksen van breuken. Een breuk wordt voorgesteld als twee natuurlijke getallen die van elkaar worden gescheiden door een slash (/). De getallen voor en na de slash worden respectievelijk de teller en de noemer van de breuk genoemd. In een reeks breuken worden de voorstellingen van de individuele breuken van elkaar gescheiden door één enkele spatie. De verzameling \mathcal{W} bevat alle woorden die enkel bestaan uit kleine letters. Gevraagd wordt:

1. Bepaal zo kort mogelijke reguliere expressies voor de volgende deelverzamelingen van \mathcal{P} :

(a) $\mathcal{P}_1 = \{p \in \mathcal{P} \mid p \text{ bestaat uit exact 7 breuken}\}$

voorbeeld: $4135/326 \ 1/44 \ 1/764 \ 14/1 \ 91/116 \ 1435/3576 \ 38/3 \in \mathcal{P}_1$
 $19/98 \ 671/97 \ 3668/1481 \ 703/841 \ 379/153 \notin \mathcal{P}_1$

(b) $\mathcal{P}_2 = \{p \in \mathcal{P} \mid \text{de noemers van alle breuken in } p \text{ zijn even}\}$

voorbeeld: $2781/52 \ 89/8 \ 601/856 \ 43/4 \ 2183/2712 \in \mathcal{P}_2$
 $23/2 \ 297/1822 \ 5966/4005 \ 67/6326 \ 1179/151 \notin \mathcal{P}_2$

(c) $\mathcal{P}_3 = \{p \in \mathcal{P} \mid p \text{ bevat minstens één breuk waarvan teller en noemer uit exact twee cijfers bestaan}\}$

voorbeeld: $3793/1586 \ 27/70 \ 107/88 \ 167/795 \ 61/2 \in \mathcal{P}_3$
 $2687/2548 \ 1657/53 \ 5077/7 \ 7/67 \ 80/121 \notin \mathcal{P}_3$

(d) $\mathcal{P}_4 = \{p \in \mathcal{P} \mid \text{in alle breuken van } p \text{ is het eerste cijfer van de teller gelijk aan het laatste cijfer van de noemer}\}$

voorbeeld: $91/379 \ 7/67 \ 6353/596 \ 495/4184 \ 743/2947 \in \mathcal{P}_4$
 $65/36 \ 7/26 \ 1/6 \ 746/9 \ 601/24 \ 441/4924 \notin \mathcal{P}_4$

Geef telkens een Unix commando waarin de reguliere expressie gebruikt wordt door een commando uit de grep familie om enkel de regels van het tekstbestand naar `stdout` te schrijven waarvan het patroon p behoort tot \mathcal{P}_i ($i = 1, 2, 3, 4$).

2. Bepaal als volgt de woorden $w_1 \ w_2 \ w_3 \ w_4$ van een geheime boodschap:

(a) het woord w_1 staat op de unieke regel waarvan p behoort tot $\mathcal{P}_1 \cap \mathcal{P}_2$

(b) het woord w_2 staat op de unieke regel waarvan p behoort tot $\mathcal{P}_2 \cap \mathcal{P}_3$

(c) het woord w_3 staat op de unieke regel waarvan p behoort tot $\mathcal{P}_3 \cap \mathcal{P}_4$

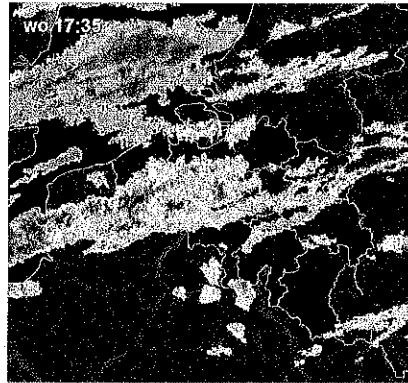
(d) het woord w_4 staat op de unieke regel waarvan p behoort tot $\mathcal{P}_4 \cap \mathcal{P}_1$

Geef telkens een Unix commando waarin de reguliere expressies voor de verzamelingen \mathcal{P}_i ($i = 1, 2, 3, 4$) gebruikt worden door commando's uit de grep familie om het woord w_j ($j = 1, 2, 3, 4$) op te zoeken in het tekstbestand en uit te schrijven naar `stdout`. Hierbij is het niet toegelaten om het woord w_j letterlijk uit te schrijven (bv. `echo w_j`).

Opgave 3 (buienradar)

(15 pt)

Online vind je tal van websites waarop je (regionale) informatie over het weer kan raadplegen. Vaak combineren die historische data met voorspellingen voor de nabije toekomst. Zo brengt Buienradar met behulp van radar- en satellietgegevens de actuele neerslag boven Europa in beeld.



Opgave

We hebben de Buienradar API aangesproken om historische data te downloaden over de hoeveelheid neerslag in verschillende Europese steden. Die informatie zit opgeslagen in het CSV-bestand `rain.csv`. Elke regel van dat bestand beschrijft één meting van de neerslag op een bepaald tijdstip in een bepaalde stad. Die beschrijving bestaat uit de volgende drie informatievelden die van elkaar gescheiden worden door komma's:

1. stad waar meting gebeurd is
2. tijdstip waarop meting gebeurd is (in Unix-tijd)
3. gemeten hoeveelheid neerslag (in millimeter)

De eerste regels van het bestand zien er bijvoorbeeld als volgt uit:

```
1 Amsterdam,1546297200,230
2 Amsterdam,1546383600,319
3 Amsterdam,1546470000,245
4 Amsterdam,1546556400,424
5 Amsterdam,1546642800,716
6 Amsterdam,1546729200,783
7 Amsterdam,1546815600,43
8 Amsterdam,1546902000,115
9 Amsterdam,1546988400,260
10 Amsterdam,1547074800,464
11 ...
```

Gevraagd wordt om een bash shell script `stats` te schrijven waarmee een overzicht kan gegenereerd worden met basisstatistieken over de neerslag voor een bepaalde stad over een bepaalde periode. De naam van de stad moet als argument aan het shell script doorgegeven worden. Daarnaast kan als tweede argument ook nog de padnaam doorgegeven worden van het CSV-bestand met historische data over de neerslag. Als er geen tweede argument wordt doorgegeven, dan moet het shell script de historische data inlezen via `stdin`.

Het shell script moet naar `stdout` een overzicht uitschrijven met de volgende informatie:

1. stad waar metingen gebeurd zijn (CITY)
2. aantal metingen binnen de gegeven periode waarop de statistieken gebaseerd zijn (COUNT)
3. kleinste gemeten hoeveelheid neerslag (in millimeter) binnen de gegeven periode (MIN)
4. grootste gemeten hoeveelheid neerslag (in millimeter) binnen de gegeven periode (MAX)

Het exacte formaat van het overzicht kan afgeleid worden uit onderstaande voorbeeldsessie. Om de periode vast te leggen waarover de statistieken moeten berekend worden, moet het shell script de volgende twee opties met verplicht argument ondersteunen:

- optie `-s <int>`: begintijdstip (in Unix-tijd) van periode; als deze optie niet gebruikt wordt dan start de periode bij de vroegste meting in de data
- optie `-e <int>`: eindtijdstip (in Unix-tijd) van periode; als deze optie niet gebruikt wordt dan eindigt de periode bij de laatste meting in de data

Het shell script moet volgende foutafhandeling voorzien:

- als het shell script niet de gepaste opties meekrijgt (enkel ondersteuning voor opties `-s` en `-e`), als er geen verplicht argument wordt meegegeven met de opties `-s` en `-e`, of als er niet één of twee argumenten worden doorgegeven aan het shell script, dan moet de gepaste boodschap (zie onderstaande voorbeeldsessie) uitgeschreven worden naar `stderr` en moet het shell script eindigen met `exit status 1`
- als de padnaam van het CSV-bestand dat aan het shell script wordt doorgegeven niet verwijst naar een gewoon leesbaar bestand, dan moet de gepaste boodschap (zie onderstaande voorbeeldsessie) uitgeschreven worden naar `stderr` en moet het shell script eindigen met `exit status 2`
- als de argumenten die aan de opties `-s` en `-e` wordt meegegeven geen natuurlijke getallen zijn, of als het gegeven eindtijdstip van de periode vóór het gegeven begintijdstip valt, dan moet de gepaste boodschap (zie onderstaande voorbeeldsessie) uitgeschreven worden naar `stderr` en moet het shell script eindigen met `exit status 3`
- als er in de data geen metingen van de gegeven stad gevonden worden die binnen de gegeven periode vallen, dan moet de gepaste boodschap (zie onderstaande voorbeeldsessie) uitgeschreven worden naar `stderr` en moet het shell script eindigen met `exit status 4`

Als er zich geen fouten voordoen, dan moet het shell script eindigen met `exit status 0`.

Voorbeeld

Onderstaande voorbeeldsessie geeft aan hoe het shell script `stats` moet kunnen gebruikt worden. Hierbij gaan we ervan uit dat het tekstbestand `rain.csv` zich in de huidige directory bevindt.

```
$ stats
Syntaxis: stats [-s START] [-e EIND] STAD [FILE]
$ echo $?
1
$ stats -abc Brussel
Syntaxis: stats [-s START] [-e EIND] STAD [FILE]
$ echo $?
1
$ stats Brussel xxx
stats: het opgegeven bestand bestaat niet of is niet leesbaar.
$ echo $?
2
$ stats Brussel rain.csv
CITY: Brussel
COUNT: 364
MIN: 2
MAX: 997
$ stats Brussel < rain.csv
CITY: Brussel
```

```

COUNT: 364
MIN: 2
MAX: 997
$ stats -s 1553209200 Brussel rain.csv
CITY: Brussel
COUNT: 284
MIN: 2
MAX: 997
$ stats -s 1553209200 -e 1562968800 Brussel rain.csv
CITY: Brussel
COUNT: 114
MIN: 27
MAX: 997
$ stats -e 1562968800 Brussel rain.csv
CITY: Brussel
COUNT: 194
MIN: 6
MAX: 997
$ stats -s "2019-01-09" Brussel rain.csv
stats: ongeldige periode
$ echo $?
3
$ stats -e "2019-01-09" Brussel rain.csv
stats: ongeldige periode
$ echo $?
3
$ stats -s 1562968800 -e 1553209200 Brussel rain.csv
stats: ongeldige periode
$ echo $?
3
$ stats -s 1646297200 Brussel rain.csv
stats: geen data voor de opgegeven periode
$ echo $?
4

```

Opgave 4 (examenfeedback)

(15 pt)

Bij het afnemen van een examen dient elke student één of meer pagina's met antwoorden in. Tijdens het beoordelen van het examen wordt bij elk antwoord een score geschreven die de student voor die vraag behaald heeft, eventueel aangevuld met wat bijkomende opmerkingen. Als feedback krijgt elke individuele student een PDF-bestand doorgestuurd met een scan van zijn of haar antwoorden, en de scores en opmerkingen die de verbeteraars daarop hebben aangebracht.

Om dit logistiek te kunnen bolwerken wordt eerst de volledige stapel met alle antwoorden van alle studenten in één keer ingescand. Daarna worden de gescande pagina's per student uitgesplitst. Nu kan het echter gebeuren dat al die pagina's tijdens het verbeteren willekeurig door elkaar geraakt zijn. Bijvoorbeeld omdat de verschillende vragen door verschillende personen verbeterd worden en de ingediende antwoorden eerst per vraag gerangschikt werden, of gewoon omdat de stapel papier per ongeluk uit iemands handen is gevallen. Bovendien kan het ook gebeuren dat er hier en daar wat pagina's verloren geraakt zijn, gewoon uit slordigheid of door technische tekortkomingen van de scanner.

Om te weten welke pagina's van welke student zijn en om de originele volgorde van de pagina's te kunnen reconstrueren, wordt aan elke pagina een hoofding toegevoegd. Die bestaat uit een unieke code per student, gevolgd door een spatie, een koppelteken, nog een spatie en een paginanummer. Het paginanummer gebruikt het formaat n/m, waarbij n het volnummer is van de antwoordbladen per student (waarbij de antwoordbladen van elke student vanaf 1 genummerd worden) en m het aantal antwoordbladen is die de student heeft ingediend (het aantal antwoordbladen dat wordt ingediend kan per student verschillend zijn).

081eb29d496a6f7634d411393ad99b10c9fdcc3a -- 2/3

Vraag 6: Donec dictum quam et ante elementum ultricies?

Ca
aga
na
egi
pa
fer

5078225f87bdc04b67ff96406ac2705537898409 -- 1/4

Examen Latijn 2020

NAAM: Jayden Randall

Vraag 1: Duis sed augue et turpis consequat volutpat?

Donec pellentesque libero convallis, sodales magna non, aliquet est. Nullam quis velit et ante tincidunt porta. Sed ut sapien sed urna posuere caeculis quis vulputate tellus. Praesent nec odio eget risus condimentum lobortis non semper tortor. Nam convallis dui eget purus elementum, eget dapibus metus bibendum. Aliquam efficitur ligula at tortor finibus varius. Praesent et velit a ante viverra dignissim. Morbi laoreet nulla volutpat toram scelerisque, sit amet rhoncus urna lacus. Nulla vestibulum arcu vel commodo finibus. Vivamus gravida mi eu ornare placerat.

Vraag 2: Curabitur gravida risus nec faucibus mattis?

Nam sed elit in uis velitula elementum. Aliquam quis quam scelerisque, rhoncus velit vel, suscipit tortor. Aliquam imperdiet nunc ac tristique bibendum. Invenit ultricies.

De unieke code uit de hoofding is de SHA checksum die berekend wordt op basis van de naam van de student. Voor student Jayden Randall kan die bijvoorbeeld als volgt berekend worden.

```
$ echo "Jayden Randall" | shasum
5078225f87bdc04b67ff96406ac2705537898409 -
```

Opgave

Schrijf een bash shell script feedback waaraan twee argumenten moeten doorgegeven worden: *i*) de naam van een student (mag spaties bevatten) en *ii*) een PDF-bestand met een scan van de volledige stapel met alle antwoorden die de studenten op een examen hebben ingediend. Elke pagina is van een hoofding voorzien zoals omschreven in de inleiding. Het shell script moet in de huidige directory een nieuw PDF-bestand *studentennaam.pdf* genereren met alle pagina's van de gegeven student, in hun originele volgorde (volgens de paginanummering uit de hoofding). Daarbij is *studentennaam* de naam van de student zoals die aan het shell script werd doorgegeven.

Hiervoor kan het shell script de volgende strategie gebruiken:

1. splits het gegeven PDF-bestand in PDF-bestanden met de individuele pagina's
2. bekijk de hoofding van elke individuele pagina om te zien of het om een pagina van de gevraagde student gaat, wat het volgnummer van de pagina is, en wat het totaal aantal pagina's is dat we verwachten
3. bundel de pagina's van de gevraagde student, in hun originele volgorde, tot één enkel PDF-bestand met de opgegeven naam

Het shell script mag ervan uitgaan dat de argumenten die eraan doorgegeven worden geldig zijn, zonder dat dit expliciet moet gecontroleerd worden.

Als in het gegeven PDF-bestand geen enkele pagina van de gegeven student wordt teruggegeven, dan moet het shell script een gepaste boodschap (zie onderstaande voorbeeldsessie) uitschrijven naar `stderr` en eindigen met `exit status 1` zonder dat er een PDF-bestand voor de student wordt aangemaakt.

Als in het gegeven PDF-bestand één of meer pagina's van de student ontbreken, dan moet voor elke pagina een gepaste boodschap (zie onderstaande voorbeeldsessie) uitgeschreven worden naar `stderr`. In dat geval moet evenwel nog altijd een PDF-bestand met de gevonden pagina's van de student aangemaakt worden.

PDF-bestanden verwerken

Voor het verwerken van PDF-bestanden kan je gebruikmaken van de Linux PDF *rendering library* poppler-utils. Deze library is beschikbaar op Dodona en op helios, maar is free software en kan je ook zelf lokaal installeren indien nodig. De volgende commando's uit de library komen handig van pas voor het implementeren van het shell script uit deze opgave:

- pdfseparate: individuele pagina's uit een PDF-bestand extraheren
- pdffunite: meerdere PDF-bestanden bundelen tot één enkel PDF-bestand
- pdftotext: PDF-bestanden omzetten naar platte tekst

Voorbeeld

Onderstaande voorbeeldsessie geeft aan hoe het shell script feedback moet kunnen gebruikt worden. Hierbij gaan we ervan uit dat het tekstbestand antwoorden.pdf zich in de huidige directory bevindt.

```
$ feedback "Tobias Woods" antwoorden.pdf      # "Tobias Woods.pdf"
$ feedback "Rogelio Mcdaniel" antwoorden.pdf  # "Rogelio Mcdaniel.pdf"
feedback: pagina 2/4 niet gevonden
$ feedback "Mollie Berg" antwoorden.pdf      # "Mollie Berg.pdf"
$ feedback "Maverick Dennis" antwoorden.pdf  # "Maverick Dennis.pdf"
extraheren: geen pagina's gevonden
$ echo $?
1
$ feedback "Dante Rasmussen" antwoorden.pdf  # "Dante Rasmussen.pdf"
extraheren: pagina 2/4 niet gevonden
extraheren: pagina 3/4 niet gevonden
$ feedback "Adriel Ferguson" antwoorden.pdf  # "Adriel Ferguson.pdf"
$ feedback "Irvin Johnston" antwoorden.pdf    # "Irvin Johnston.pdf"
extraheren: pagina 1/3 niet gevonden
```