

INLIGTINGSTEGNOLOGIE: VRAESTEL II

Tyd: 3 uur

120 punte

LEES ASSEBLIEF DIE VOLGENDE INSTRUKSIES NOUKEURIG DEUR

1. Hierdie vraestel bestaan uit 13 bladsye. Maak asseblief seker dat jou vraestel volledig is.
2. Hierdie vraestel moet beantwoord word deur objekgeoriënteerde programmeringsbeginsels te gebruik. Jou program moet sinvol gebruik maak van metodes en parameters.
3. Hierdie vraestel is in twee afdelings verdeel. Al die kandidate moet albei afdelings beantwoord.
4. Hierdie vraestel is opgestel in programmeerterme wat nie eie is aan enige spesifieke programmeertaal (Java/Delphi) of databasis (Access/MySQL) nie.
5. Maak seker dat jy die vrae beantwoord op die wyse wat beskryf word, aangesien punte vir jou oplossing toegeken sal word ooreenkomstig die spesifikasies wat in die vraag gegee word.
6. Beantwoord slegs wat in elke vraag gevra word. As die vraag byvoorbeeld nie vra vir datageldigheidstoetsing nie, word geen punte daarvoor toegeken nie en hoef geen kode dus geskryf te word nie.
7. As jy 'n gedeelte van die kode nie kan laat werk nie, moet jy dit as kommentaar merk sodat dit nie uitgevoer word nie en sodat jy met die eksamen kan voortgaan. Probeer indien moontlik om die fout te verduidelik om die nasiener te help.
8. Wanneer jy in jou kode toegang tot 'n lêer verkry, MOENIE die name van die volledige pad van die lêer gebruik nie, aangesien dit probleme sal skep wanneer die program nagesien word op 'n ander rekenaar as die een waarop jy werk. Verwys bloot na die lêers deur die name en uitbreidings daarvan te gebruik waar nodig.
9. Jou programme moet op so 'n manier gekodeer word dat hulle met enige data sal werk en nie slegs die voorbeelddata wat voorsien is of enige data-uittreksels wat in die vraestel voorkom nie. Jy word aangeraai om die datalêers wat voorsien word, noukeurig te bestudeer.
10. Maak seker dat roetines soos soektogte, sorterings en selekterings uit eerste beginsels ontwikkel word en dat jy nie die ingeboude eienskappe van 'n programmeertaal vir enigeen van hierdie roetines gebruik nie.

11. Alle datastrukture moet deur jou, die programmeerder, gedefinieer en verklaar word. Jy mag nie komponente wat binne die koppelvlak voorsien word, gebruik om data te stoor en later te herwin nie.
 12. Lees die hele vraestel voordat jy 'n datastruktuur kies. Jy kan vind dat daar 'n alternatiewe metode is om die data voor te stel wat dalk meer doeltreffend sal wees met inagneming van die vrae wat in die vraestel gevra word.
 13. Jy moet al jou werk gereeld stoor op die skyf wat jy ontvang het of die skyfspasie wat vir hierdie eksamen aan jou toegeken is.
 14. Indien daar 'n tegniese onderbreking is wat verhoed dat jy jou eksamen kan skryf, byvoorbeeld 'n kragonderbreking, sal jy wanneer jy weer aan jou eksamen begin skryf slegs die tyd gegee word wat oor was toe die onderbreking begin het. Geen ekstra tyd sal gegee word om werk wat nie gestoor is nie weer oor te doen nie.
 15. Maak seker jou eksamennommer verskyn as kommentaar in elke program wat jy kodeer, sowel as op elke bladsy sigkopie wat jy inlewer.
 16. Druk 'n kodelys van al die programme/klasse wat jy kodeer. Drukwerk moet na die eksamen gedoen word. Jy sal na afloop van die eksamen 'n halfuur ontvang om te druk. Jou onderwyser sal jou inlig oor die reëlins vir die druk van jou werk.
-

SCENARIO

Die Suid-Afrikaanse Amateurtennisspelersvereniging (SAATPA) wil verskeie sagteware-gebaseerde oplossings implementeer om hul bedrywighede te vereenvoudig. Meer spesifiek wil hulle 'n databasis hê om op die hoogte te bly van die belangrike grand slam-toernooie en 'n program om die spelers en wedstryde vir 'n komende internasionale toernooi te verwerk.

AFDELING A GESTRUKTUREERDE NAVRAAGTAAL

Professionele tennis het vier belangrike grand slam-toernooie wat op 'n jaarlikse basis gespeel word. SAATPA wil 'n rekord hou van al die weners van hierdie vier toernooie vir die laaste paar dekades. Hulle wil die besonderhede van elke toernooi, die besonderhede van die spelers en die uitslag van die eindwedstryde vir elke jaar van elke toernooi opteken.

Die databasis **TennisDB** bevat drie tabelle. Die eerste tabel **tblSpelers** bevat die besonderhede van elke speler, met inbegrip van hul naam, die land wat hulle verteenwoordig en hul geboortedatum. Die tweede tabel **tblToernooie** bevat inligting oor elkeen van die vier groot toernooie, met inbegrip van die naam van die toernooi, die gasheerstad, die tipe oppervlak, die jaar waarin die toernooi begin is en die prysgeld (in Amerikaanse dollar en Suid-Afrikaanse rand). Die prysgeldveld vir ZAR is doelbewus oop gelos. Die derde tabel **tblFinaal** bevat besonderhede van die wenner en naaswenner (tennisspelers) wat in elke jaar in die eindwedstryd van elke toernooi gespeel het, die jaar waarin die eindwedstryd gespeel is en die naam van die eindwedstryd. Jy moet nuttige inligting uit die databasis onttrek.

Die velde in die databasis word hieronder bespreek. Onder elke beskrywing is 'n skermkiekie van die eerste 10 rye data vir jou gerief. Die tabelle bevat nog meer data:

tblSpelers

Veldnaam	Datatype	Beskrywing
<u>SpelerID</u>	Outomatiese nommer	Hierdie veld ken 'n unieke ID vir elke speler toe. Hierdie veld is 'n outomatiesenommer-veld.
Land	Teks	Hierdie veld bevat die speler se land as teks.
VolleNaam	Teks	Hierdie veld bevat die speler se volle naam as teks.
GebDatum	Datum/Tyd	Hierdie veld bevat die speler se geboortedatum as 'n datum/tyd.

tblSpelers			
SpelerID	Land	VolleNaam	GebDatum
1	SWE	Stefan Edberg	1966-01-19
2	TCH	Ivan Lendl	1960-03-07
3	GER	Boris Becker	1967-11-22
4	USA	Jim Courier	1970-08-17
5	USA	Pete Sampras	1971-08-12
6	USA	Andre Agassi	1970-04-29
7	CZE	Petr Korda	1968-01-23
8	RUS	Yevgeny Kafelnikov	1974-02-18
9	SWE	Thomas Johansson	1975-03-24
10	SUI	Roger Federer	1981-08-08

tblToernooie

Veldnaam	Datatype	Beskrywing
<u>ToernooiID</u>	Outomatiese nommer	Hierdie veld bevat 'n unieke ID vir elke toernooi. Hierdie veld is 'n outomatiesenommer-veld.
ToernooiNaam	Teks	Hierdie veld bevat die toernooinaam as teks.
Stad	Teks	Hierdie veld bevat die toernooi se gasheerstad as teks.
Oppervlak	Teks	Hierdie veld bevat die tipe oppervlak as teks.
JaarBegin	Getal	Hierdie veld bevat die jaar wat die toernooi die eerste keer gehou is as getal.
PrysGeld	Getal	Hierdie veld bevat die bedrag prysgeld in Amerikaanse dollar as 'n reële getal.
PrysGeldZAR	Geldeenheid	Hierdie veld bevat die bedrag prysgeld in Suid-Afrikaanse rand. Doelbewus oop gelos.

tblToernooie						
ToernooiID	ToernooiNaam	Stad	Oppervlak	JaarBegin	PrysGeld	PrysGeldZAR
1	Australiese Ope	Melbourne	Hard	1905	2457000	
2	Franse Ope	Parys	Klei	1891	1873162	
3	Wimbledon	Londen	Gras	1877	2675200	
4	Amerikaanse Ope	New York	Hard	1881	1900000	

tblFinaal

Veldnaam	Datatype	Beskrywing
ToernooiID	Getal	Hierdie veld bevat die toernooi-ID van die eindwedstryd as 'n getal. Hierdie veld is 'n vreemde sleutel vir tblToernooie.
FinaalJaar	Getal	Hierdie veld bevat die jaar van die eindwedstryd as 'n getal.
FinaalNaam	Teks	Hierdie veld bevat die tipe eindwedstryd as teks. Moontlike waardes is "Mansenkelspel" en "Damesenkelspel".
WennerID	Getal	Hierdie veld bevat die ID van die speler wat die eindwedstryd gewen het. Hierdie veld is 'n vreemde sleutel vir tblSpelers.
NaasWennerID	Getal	Hierdie veld bevat die ID van die speler wat die eindwedstryd verloor het. Hierdie veld is 'n vreemde sleutel vir tblSpelers.

tblFinaal				
ToernooiID	FinaalJaar	FinaalNaam	WennerID	NaasWennerID
1	1990	Mansenkelspel	2	1
2	1990	Mansenkelspel	30	6
2	1990	Damesenkelspel	17	16
3	1990	Damesenkelspel	50	16
3	1990	Mansenkelspel	5	6
4	1990	Damesenkelspel	15	96
4	1990	Mansenkelspel	1	3
1	1990	Damesenkelspel	16	70
3	1991	Damesenkelspel	17	15
4	1991	Mansenkelspel	53	3

VRAAG 1

- 1.1 Skryf 'n navraag wat die **Land** en **VolleNaam** van alle **Spelers** alfabeties volgens land en dan volgens naam gesorteer, sal lys. (3)
- 1.2 Skryf 'n navraag wat al die besonderhede sal lys vir enige **Finaal** wat voor die jaar 2000 gespeel is. (3)
- 1.3 Die speler "Bob Rogers" (**SpelerID 101**) is nie 'n professionele tennisspeler nie, maar is verkeerdelik in die databasis ingevoer. Skryf 'n navraag om sy inligting uit die **Spelers**-tabel te skrap. (2)
- 1.4 Skryf 'n navraag wat die **PrysGeldZAR**-veld in die **Toernooie**-tabel met die ekwivalente bedrag in Suid-Afrikaanse rand sal bywerk.
(1 Amerikaanse dollar = 11.51 Suid-Afrikaanse rand) (3)
- 1.5 Skryf 'n navraag wat die totale getal verjaardae van **Spelers** wat in elke maand van die jaar verjaar, sal toon. Toon die maand van die jaar as 'n getal (Januarie synde 1 en Desember synde 12) sowel as die totale getal verjaardae in elke maand. Die afvoer moet soos volg wees: (4)

GebMaand	GetalVerjaardae
1	7
2	9
3	8
4	9
5	5
6	13
7	6
8	10
9	12
10	8
11	4
12	9

- 1.6 Skryf 'n navraag wat die totale getal **Spelers** wat elke land in die databasis het, sal vertoon. Vertoon slegs die lande wat meer as 10 spelers het. Vertoon die **Land** sowel as die totale getal spelers in dalende volgorde. 'n Voorbeeld van die afvoer word hieronder gegee: (6)

Land	GetalSpelers
USA	14
ESP	11

- 1.7 Skryf 'n navraag om die **ToernooiNaam** te vertoon van alle **Toernooie** wat **PrysGeld** van meer as die gemiddelde **PrysGeld** van al die toernooie het. 'n Voorbeeld van die afvoer word hieronder gegee: (4)

ToernooiNaam
Australiese Ope
Wimbledon

- 1.8 Skryf 'n navraag wat die **VolleNaam**, **ToernooiNaam** en **FinaalJaar** sal lys van alle spelers wat 'n toernooi in 'n skrikkeljaar ná 1995 gewen het. 'n Skrikkeljaar is 'n jaar wat presies deelbaar is deur 4, maar jare wat presies deelbaar is deur 400 word nie ingesluit nie. 'n Voorbeeld van die eerste 5 rye afvoer word hieronder gegee: (8)

VolleNaam	ToernooiNaam	FinaalJaar
Monica Seles	Australiese Ope	1996
Boris Becker	Australiese Ope	1996
Yevgeny Kafelnikov	Franse Ope	1996
Steffi Graf	Franse Ope	1996
Steffi Graf	Wimbledon	1996

- 1.9 Soos dikwels gebeur, het die man en vrou wat die Australiese Ope in 2015 gewen het, ook die Franse Ope teen dieselfde opponente gewen. Skryf 'n navraag wat die data in die **Finaal**-tabel sal gebruik om dieselfde weners van die 2015 Australiese Ope (mans- en damesenkelspelfinale) vir die 2015 Franse Ope in te voeg. Die enigste ingevoegde waarde wat jy vasgekodeer mag skrap, is die ToernooiID. (7)

40 punte

AFDELING B OBJEKGEORIËNTEERDE PROGRAMMERING

SAATPA wil 'n Suid-Afrikaanse Ope Amateurstennistoernooi in Suid-Afrika aanbied. Die eerste rondte van die manstoernooi sal uit 32 internasionale spelers bestaan. Elke speler sal gekeur word en teen een ander speler in 'n wedstryd speel.

Elke **Speler** het die volgende inligting:

- Keuring – die rangnommer van die speler vir die toernooi
- Volle naam – die naam van die speler
- Land – die land wat die speler verteenwoordig

In die eerste rondte sal elke speler teen een, en slegs een, ander speler speel, wat 'n totaal van 16 wedstryde beteken. Om die toernooi meer opwindend te maak, sal die hoogste gekeurde speler teen die laagste gekeurde speler speel. Die speler wat tweede gekeur is, sal in 'n wedstryd teen die speler speel wat tweede van onder af gekeur is, ensovoorts. Met ander woorde, Gekeurde 1 speel teen Gekeurde 32, Gekeurde 2 speel teen Gekeurde 31, Gekeurde 3 speel teen Gekeurde 30, ensovoorts.

Elke **Wedstryd** bevat die volgende inligting:

- 'n Unieke wedstrydkode (hieronder verduidelik)
- Die besonderhede van die twee spelers in die wedstryd
- Die eindtelling/uitslag van die wedstryd – gegee as die getal potte vir elke stel gewen
 - Die telling van 'n stel bestaan uit die getal potte wat elke speler in daardie stel gewen het. Speler 1 se telling word eerste gelys, gevolg deur Speler 2 se telling (6-0 beteken byvoorbeeld dat 6 potte deur Speler 1 gewen is en dat Speler 2 geen pot gewen het nie.)
 - Die telling van 'n wedstryd bestaan uit die telling vir elke stel (bv. 6-0, 6-8, 6-4, 6-3).

Die **WedstrydKode** is die letter "M" gevolg deur die keuring van die eerste speler en dan die keuring van die tweede speler. In die wedstryd waar Gekeurde 4 teen Gekeurde 29 speel, sal die WedstrydKode byvoorbeeld "M429" wees.

Jy het 'n lêer genaamd **spelers.txt** ontvang wat die inligting van presies 32 spelers bevat wat aan die eerste rondte van die manstoernooi deelneem. Daar sal altyd presies 32 spelers in die gegewe lêer wees. 'n Voorbeeld van die eerste 10 reëls van die lêer word hieronder gegee:

1,Colby Noble,Hong Kong
2,Craig Hamilton,Myanmar
3,Dante Franklin,South Korea
4,Chadwick Baird,India
5,Wallace Forbes,Cameroon
6,Tanek Jimenez,Italy
7,Jared Flynn,Burundi
8,Yasir Rosario,Taiwan
9,Levi Klein,Botswana
10,Kibo Ford,American Samoa

Elke reël van hierdie lêer bevat die inligting van 'n enkele speler in die manstoernooi. Die reël word soos volg geformateer:

keuring, naam, land

VRAAG 2

Gebruik die klasdiagram hieronder om 'n nuwe klas genaamd **Speler** te skep. Hierdie klas sal gebruik word om objekte te skep wat die besonderhede van een speler in die toernooi sal stoor. Die diagram hieronder dui die eienskappe/velde en metodes aan wat benodig word.

Speler
Eienskappe/Velde: – Heelgetal keuring – String vollenaam – String land
Metodes: + Konstruktor(Heelgetal inKeuring, String inVolleNaam, String inLand) + kryLandKode : String + toString() : String

- 2.1 Skryf kode om 'n nuwe klas genaamd **Speler** te skep. (1)
- 2.2 Skryf kode om die drie eienskappe/velde vir die **Speler**-klas te skep soos aangedui in die klasdiagram hierbo. (3)
- 2.3 Skryf kode om 'n konstruktormetode te skep wat 'n heelgetal en twee stringe as parameters aanvaar wat onderskeidelik die keuring, speler se volle naam en land verteenwoordig. Hierdie parameters moet gebruik word om die waardes van die drie eienskappe/velde van die **Speler**-klas te stel. (3)
- 2.4 Skryf kode om 'n metode genaamd **kryLandKode** te skep wat 'n string sal terugstuur wat 'n drieletterkode vir die speler se land bevat. Die drieletterkode is altyd hoofletters en word soos volg gegenereer:
- Indien die naam van die land 'n enkele woord is, gebruik die eerste drie letters van die landnaam ("China" word byvoorbeeld "CHI").
 - Indien die naam van die land uit twee woorde bestaan, gebruik die eerste letter van die eerste woord en die eerste twee letters van die tweede woord ("South Africa" word byvoorbeeld "SAF").
- (4)
- 2.5 Skryf kode om 'n **toString**-metode te skep wat die speler se volle naam en landkode tussen hakies, geskei deur 'n spasie, sal terugstuur. Die formaat is soos volg:
- vollenaam<space>(landkode)
- byvoorbeeld:
- Colby Noble (HK0)

(3)
[14]

VRAAG 3

Gebruik die klasdiagram hieronder om 'n nuwe klas genaamd **Wedstryd** te skep. Hierdie klas sal gebruik word om die besonderhede van 'n enkele wedstryd te stoor. Twee van die eienskappe/velde is **Speler**-objekte (die klas in die vorige vraag geskep). Die diagram hieronder dui die eienskappe/velde en metodes aan wat benodig word.

Wedstryd
Eienskappe/Velde: – String wedstrydKode – Speler speler1 – Speler speler2 – String telling
Metodes: + Konstruktor(String inWedstrydKode, Speler inSpeler1, Speler inSpeler2) + kryWedstrydKode : String + stelTelling (String inTelling) + toString() : String

3.1 Skryf kode om 'n nuwe klas genaamd **Wedstryd** te skep. (1)

3.2 Skryf kode om vier eienskappe/velde te skep wat die **wedstrydKode**, **speler1**, **speler2** en **telling** wat met 'n wedstryd geassosieer word, sal stoor. Kies toepaslike datatipes vir hierdie eienskappe/velde, maar let daarop dat die **speler1**- en **speler2**-eienskappe/velde objekte van die klas **Speler** is. (3)

3.3 Skryf kode om 'n konstruktormetode te skep wat al die eienskappe/velde van die **Wedstryd**-klas sal inisialiseer. Let daarop dat benewens die **wedstrydKode**-parameter die tweede en derde parameters objekte van die **Speler**-klas is. Gebruik die eerste drie parameters om waardes aan hul onderskeie velde toe te ken. Stel die tellingkenmerk op "X" om aan te dui dat die wedstryd nog nie 'n telling het nie. (4)

3.4 Skryf kode om 'n toegangermetode vir die **wedstrydKode**-eienskap te skep. Hierdie metode moet 'n string terugstuur. (1)

3.5 Skryf kode om 'n mutatormetode vir die tellingeieenskap te skep. Hierdie metode moet 'n enkele stringparameter neem en die waarde van die tellingveld dienoooreenkomstig stel. (1)

3.6 Skryf kode om 'n **toString**-metode te skep wat 'n string bestaande uit die inligting vir die wedstryd sal terugstuur. Die string wat teruggestuur word, moet in die volgende formaat wees:

Speler1 vs. Speler2 Telling

Indien die tellingveld op "X" gestel is, vertoon "Nog nie gespeel nie" vir die telling, of vertoon andersins net die inhoud van die tellingveld. Vir 'n wedstryd wat byvoorbeeld geen telling het nie:

M132 Colby Noble (HK0) vs. Jason Underwood (POR) Nog nie gespeel nie

'n Voorbeeld van 'n wedstryd wat 'n telling het, is soos volg:

M429 Chadwick Baird (IND) vs. Knox Olson (SOM) 6-0 6-8 6-4 6-3
Speler 1 wen

(6)
[16]

VRAAG 4

- 4.1 Skryf kode om 'n nuwe klas genaamd **ToernooiBestuurder** te skep. (1)
- 4.2 Skryf kode om die volgende twee skikkings as instansieveranderlikes te verklaar:
- 'n Skikking wat gebruik kan word om presies 32 **Speler**-objekte te stoor.
 - 'n Skikking wat gebruik kan word om presies 16 **Wedstryd**-objekte te stoor. (4)
- 4.3 Skryf kode om 'n konstruktormetode te skep wat die inhoud van 'n lêer met spelers sal lees. Die lêernaam moet as 'n stringparameter vir die konstruktormetode gegee word. Elke reël sal tot gevolg hê dat een **Speler**-objek by die skikking gevoeg word. Doen die volgende:
- Kontroleer of die lêer bestaan. Indien nie, vertoon 'n foutboodskap.
 - Maak die lêer oop om te lees.
 - Loop 32 keer deur die lêer. In elke herhaling van die lus:
 - Lees elke reël in en verdeel die **Speler**-data wat in die reël vervat is in die keuring, volle naam en land.
 - Skep 'n **Speler**-objek deur die data te gebruik.
 - Voeg die **Speler**-objek by die skikking.
 - Let daarop dat jy geen **Wedstryd**-objekte in hierdie metode hoef te skep nie. (10)
- 4.4 Skryf kode wat 'n metode genaamd **lysAlleSpelers** sal skep. Hierdie metode moet 'n string terugstuur wat die volleNaam en landKode van alle spelers bevat. Gebruik die objek se **toString**-metodes wat jy in die vrae hierbo geskep het. Elke speler se besonderhede moet op 'n nuwe reël wees. (5)
- 4.5 Skryf kode om 'n metode genaamd **vulWedstryde** te skep wat 16 **Wedstryd**-objekte sal genereer en hulle by die skikking sal voeg wat gebruik word om wedstryde te stoor. Die eerste speler in die spelersskikking (gekeurde 1) sal die eerste speler in die eerste wedstryd wees. Die laaste speler in die skikking (gekeurde 32) sal die tweede speler in die eerste wedstryd wees om die wedstrydKode "M132" te gee. Die tweede gekeurde speler sal teen die speler met keuring 31 in wedstryd 2 speel (wedstrydkode "M231"). Gebruik die toepaslike **Speler**-objekte in die spelersskikking as parameters wanneer 'n nuwe wedstryd geskep word. Sodra jy die wedstrydskikking met **Wedstryd**-objekte gevul het, moet jou metode 'n lys van wedstryde as 'n string terugstuur. (7)
- 4.6 Skryf kode om 'n metode genaamd **soekWedstryd** te skep wat 'n wedstrydKode as 'n stringparameter neem en die ooreenstemmende **Wedstryd**-objek uit die wedstrydskikking terugstuur. Hierdie metode sal die wedstrydskikking deursoek vir die wedstryd met dieselfde wedstrydKode en daardie objek terugstuur. Jy hoef nie voorsiening te maak vir die geval waar 'n wedstryd nie gevind word nie. (5)

[32]

VRAAG 5

- 5.1 Skryf kode om 'n eenvoudige gebruikerskoppelvlak genaamd **TennisGK** te skep wat eenvoudige afvoer moontlik sal maak. (1)
- 5.2 Verklaar en instansieer 'n **ToernooiBestuurder**-objek op die toepaslike plek in die kode. Roep die konstruktor deur "**spelers.txt**" as die lêernaamargument te gebruik. (1)
- 5.3 Skryf kode wat die volgende sal vertoon deur die toepaslike metodes in die **ToernooiBestuurder**-klas te roep. Jy moet die metodes in die volgende volgorde roep:
- Vertoon alle Spelers
Vul en vertoon Wedstryde
- (2)
[4]

VRAAG 6

Jy moet nou kode skryf om die uitslag van 'n wedstryd te verwerk wat in 'n tekslêer gegee word om 'n telling te genereer. In manstennistoernooie werk die telling soos volg:

- Elke wedstryd bestaan uit hoogstens 5 stelle.
- Die eerste speler wat 3 stelle wen, wen die wedstryd.
- Elke stel bestaan uit 'n aantal potte. Om 'n stel te wen, moet 'n speler aan albei die volgende voorwaardes voldoen:
 - Hy moet minstens 6 potte wen.
 - Hy moet minstens 2 potte meer as sy opponent wen.
- Vir doeleindes van hierdie vraag is die punte wat in individuele potte aangeteken word, nie relevant nie.
- Neem vir doeleindes van hierdie vraag aan dat daar geen valbylreël is nie.

Jy het 'n lêer genaamd **uitslag.txt** ontvang wat inligting bevat oor die uitslag van 'n bepaalde wedstryd. Die eerste 10 reëls van die lêer word hieronder gegee:

M429

1
1
1
1
1
1
1
1
2
2
2

Die eerste reël van die lêer bevat die **WedstrydKode** van die wedstryd waarop die uitslag in die lêer betrekking het. Elke reël na die eerste reël dui aan watter speler 'n bepaalde pot gewen het.

In die voorbeeld hierbo hou die uitslag byvoorbeeld verband met wedstryd M429. Die volgende reëls dui aan dat Speler 1 die eerste ses potte gewen het en Speler 2 die volgende drie potte. Met ander woorde, reëls 2 – 7 van die lêer is almal potte wat Speler 1 gewen het, wat beteken dat hy die eerste stel 6-0 gewen het. Die volgende drie reëls dui aan dat Speler 2 die volgende drie potte van die tweede stel gewen het, ensovoorts. Die eindtelling van wedstryd M429 is 6-0 6-8 6-4 6-3.

Hieronder is 'n volledige verduideliking van hoe die wedstryd in die gegewe lêer verwerk moet word. Let daarop dat jou oplossing vir enige uitslaglêer voorsiening moet maak. Vaskodering van 'n oplossing wat slegs vir die gegewe lêer werk, sal min of geen punte ontvang nie.

Reël 1	M429	Die Wedstrydkode
Reëls 2 – 7 verteenwoordig stel 1	1 1 1 1 1 1	Speler 1 wen 6 potte en Speler 2 wen 0 potte. Die eindtelling vir hierdie stel is 6-0.
Reëls 8 – 21 verteenwoordig stel 2	2 2 2 1 1 1 2 1 2 1 1* 2 2 2	Speler 1 wen 6 potte en Speler 2 wen 8 potte. Die eindtelling vir hierdie stel 6-8. *By hierdie punt is die telling 6-5, maar die stel moet gewen word met 'n verskil van 2 potte, dus gaan spel voort totdat die eindtelling 6-8 is.
Reëls 22 – 31 verteenwoordig stel 3	1 1 2 2 1 1 2 2 1 1	Speler 1 wen 6 potte en Speler 2 wen 4 potte. Die eindtelling vir hierdie stel is 6-4.
Reëls 32 – 40 verteenwoordig stel 4	1 1 1 2 2 1 1 2 1	Speler 1 wen 6 potte en Speler 2 wen 3 potte. Die eindtelling vir hierdie stel is 6-3.

Speler 1 wen die wedstryd aangesien hy 3 stelle (stelle 1, 3 en 4) wen en Speler 2 slegs 1 stel (stel 2) wen.

6.1 Skryf kode om 'n metode in die **ToernooiBestuurder**-klas genaamd **verwerkUitslag** by te voeg. Hierdie metode moet die inhoud van **uitslag.txt** verwerk en die besonderhede van die wedstryd (met inbegrip van die telling) as 'n string terugstuur.

Die formaat van die telling is soos volg:

- Elke stel word vertoon as die getal potte deur Speler 1 gewen, gevolg deur 'n streep, gevolg deur die getal potte deur Speler 2 gewen. Met ander woorde, 6-0 dui aan dat 6 potte deur Speler 1 gewen is en niks deur Speler 2 nie. Speler 1 se telling word altyd eerste gelys.
- Stel word deur 'n spasie geskei, byvoorbeeld 6-0 6-8 6-4 6-3.
- Las die speler wat wen aan die einde van die string by, byvoorbeeld 6-0 6-8 6-4 6-3 Speler 1 wen.

(13)

6.2 Skryf kode in **TennisGK** om die verwerkUitslag-metode te roep en die besonderhede te vertoon van die wedstryd wat geraak word. Jou afvoer moet soos volg wees:

Wedstryduitslag : M429 Chadwick Baird (IND) vs. Knox Olson (SOM)
6-0 6-8 6-4 6-3 Speler 1 wen

(1)

[14]

80 punte

Totaal: 120 punte