



INLIGTINGSTEKNOLOGIE: VRAESTEL II

Tyd: 3 uur

120 punte

LEES ASSEBLIEF DIE VOLGENDE INSTRUKSIES NOUKEURIG DEUR

1. Hierdie vraestel bestaan uit 16 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. Alle 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/JavaDB) 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 vir datageldigheidstoetsing geskryf te word nie.
7. As jy 'n gedeelte van die kode nie korrek 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 lêers verkry, MOENIE die name van die volledige pad van die lêers gebruik nie, aangesien dit probleme sal veroorsaak 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 data lêers wat voorsien word, noukeurig te bestudeer.

10. Maak seker dat roetines soos soektogte, sorterings en selekterings vir skikkings 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. Jy moet voor jy begin ook 'n rugsteunkopie (*backup*) van die oorspronklike lêers skep ingeval die oorspronklike weergawe per ongeluk deur jou oplossing gewysig word.
 14. Indien die eksamen deur 'n tegniese probleem soos 'n kragonderbreking onderbreek word, sal jy wanneer jy weer begin skryf, slegs die tyd wat oor was toe die onderbreking begin het, gegee word om jou eksamen te voltooi. Geen ekstra tyd sal gegee word om werk wat nie gestoor is nie oor te doen nie.
 15. Voeg jou eksamennummer as kommentaar in in elke program wat jy kodeer, en skryf dit op elke bladsy sigkopie (*hard copy*) 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 gegee word om te druk. Jou onderwyser sal jou inlig oor die reëlins vir die druk van jou werk.
-

SCENARIO

Om toeriste na die Stad Idayimani te lok, het die Stadstoerismeraad 'n aantal fasiliteite geskep.

Dit is:

- 'n Gesentraliseerde besprekingstelsel vir ekskursies van die middestad af na verskeie toeristebestemmings naby die stad.
- 'n Busdiens wat op verskillende roetes tussen toeristebestemmings in die stad bedryf word.

AFDELING A SQL**VRAAG 1**

Een van die fasiliteite laat toeriste toe om elke dag verskeie ekskursies te bespreek. Enige daaglikse ekskursie wat beskikbaar is, kan bespreek word deur enige toeris wat op die stelsel geregistreer is.

Drie tabelle word gebruik om data met betrekking tot die besprekings te stoor. Die velde in die databasis word in die volgende tabelle beskryf, tesame met voorbeelddata. Die eerste tien rye data word getoon, maar die tabelle bevat nog data.

TOERIS-tabel bevat die besonderhede van elke toeris:

VELD	DATATYPE	BESKRYWING
ToerisID	HEELGETAL	'n Unieke identifikasienommer vir elke toeris wat op die stelsel geregistreer is
ToerisNaam	TEKS	Die naam van die toeris
Epos	TEKS	Die eposadres soos op die stelsel geregistreer
Hotel	TEKS	Die naam van die hotel waar die toeris tans bly
DatumGeregistreer	DATUM	Die datum waarop die toeris op die stelsel geregistreer het

TOERIS-tabel – voorbeelddata (eerste 10 rekords)				
ToerisID	ToerisNaam	Epos	Hotel	DatumGeregistreer
1	Andris Nuth	anuth0@dyndns.org	Peninsula Hotel	2015-05-15
2	Flor Byres	fbyres1@cargocollective.com	President Hotel	2018-02-05
3	Dwight Crisell	dwight@crisell.com	Village Lodge	2018-04-18
4	Mary Crisell	mary@crisell.com	Village Lodge	2018-04-18
5	Darryl Poleykett	dpoleykett4@bbb.org	Mount Grace Hotel	2017-06-02
6	Kort McAndie	kmcandie5@diigo.com	Village Lodge	2015-08-22
7	Arron Haney	ahaney6@blogspot.com	Peninsula Hotel	2012-05-17
8	Jorrie Potten	jpotten@seattletimes.com	Mount Grace Hotel	2016-11-03
9	Mic MacArd	mmacard8@weebly.com	Peninsula Hotel	2018-08-02
10	Johnath Nixon	jnixon@seattletimes.com	Peninsula Hotel	2016-05-13

EKSKURSIE

Hierdie tabel bevat die besonderhede van alle daaglikse ekskursies wat vir bespreking beskikbaar is.

VELD	DATATYPE	BESKRYWING
EkskursieID	HEELGETAL	'n Unieke identifikasienommer vir elke ekskursie
EkskursieNaam	TEKS	Die naam wat aan die ekskursie gegee is
BeginUur	HEELGETAL	Die ekskursie begin op hierdie uur. Hierdie getal is in 24-uur-tydformaat, d.w.s. 0–11 word as oggend beskou, 12–16 as middag en 17–23 as aand.
EindUur	HEELGETAL	Die ekskursie eindig op hierdie uur. Dit is ook in 24-uur-tyd met waardes van 0 tot 23.
HuidigeKoste	DUBBEL	Die huidige koste van hierdie ekskursie
Bybetaling	DUBBEL	Die bewaringsbybetaling wat by die huidige koste bygetel word. Sommige ekskursies wat nie met bewaring verband hou nie, sal 'n 0-bypbetaling hê.

EKSKURSIE-tabel – voorbeelddata (eerste 10 rekords)

Data kan anders geformateer wees op jou rekenaar na gelang van sy instellings.

EkskursieID	EkskursieNaam	BeginUur	EindUur	HuidigeKoste	Bybetaling
1	Sunrise Breakfast River Cruise	5	8	550	25
2	Sunset River Cruise	17	20	700	25
3	Morning Safari	9	12	450	15
4	Afternoon Safari	13	16	450	15
5	Night Safari	21	23	450	20
6	Township Excursion 1	8	10	250	0
7	Township Excursion 2	10	12	250	0
8	Township Excursion 3	12	14	250	0
9	Township Excursion 4	14	16	250	0
10	Township Excursion 5	16	18	250	0

BESPREKING

Hierdie tabel bevat die toeriste se ekskursiebesprekingsbesonderhede.

VELD	DATATYPE	BESKRYWING
BesprekingID	OUTOMATIESE NOMMER	'n Unieke identifikasienommer vir elke bespreking
ToerisID	HEELGETAL	Die ToerisID van die toeris wat hierdie ekskursie bespreek het. Hierdie veld is die vreemde sleutel vir die TOERIS -tabel.
EkskursieID	HEELGETAL	Die EkskursieID van die ekskursie wat met hierdie bespreking verbind word. Hierdie veld is die vreemde sleutel vir die EKSKURSIE -tabel.
EkskursieDatum	DATUM	Die datum waarvoor die ekskursie deur die toeris bespreek is.
Koste	DUBBEL	Wat hierdie ekskursie die toeris gekos het

BESPREKING-tabel – voorbeelddata (eerste 10 rekords)

Data kan anders geformateer wees op jou rekenaar na gelang van sy instellings.

BesprekingID	ToerisID	EkskursieID	EkskursieDatum	Koste
1	1	1	2018-10-05	620
2	1	3	2018-10-05	465
3	1	9	2018-10-05	225
4	1	3	2018-10-06	465
5	1	13	2018-10-06	315
6	1	16	2018-10-07	720
7	2	6	2018-10-07	225
8	2	12	2018-10-07	315
9	2	2	2018-10-07	785
10	3	4	2018-10-07	475

- 1.1 Skryf 'n navraag wat die inligting sal vertoon van al die toeriste wie se eposadres met "seattletimes.com" eindig. 'n Voorbeeld van die korrekte afvoer word hieronder getoon. (Let daarop dat datumformaat op jou stelsel kan verskil.)

Toeris ID	Toeris Naam	Epos	Hotel	Datum Geregistreer
8	Jorrie Potten	jpotten@seattletimes.com	Mount Grace Hotel	2016-11-03
10	Johnath Nixon	jnixon@seattletimes.com	Peninsula Hotel	2016-05-13
11	Davis Eginton	degintona@seattletimes.com	Peninsula Hotel	2018-07-30

(4)

- 1.2 Alle gaste wat tans by die "Lunar Hotel" bly, is na die "Three Seasons Hotel" verskuif. Skryf 'n navraag om hierdie inligting by te werk.

(3)

- 1.3 Skryf 'n navraag wat die naam en duur van oggendekskursies wat hoogstens 3 uur duur, sal vertoon. Onthou dat oggendekskursies op die uur van 0 tot 11 begin. Noem die kolom wat die berekende ure vertoon "Duur". 'n Voorbeeld van die korrekte afvoer word hieronder getoon.

EkskursieNaam	Duur
Sunrise Breakfast River Cruise	3
Township Excursion 1	2
National Art Museum Excursion 1	3

(6)

- 1.4 Skryf 'n navraag wat die name van die hotelle met minstens 3 toeriste wat daar bly, sal vertoon. Vertoon ook die aantal toeriste wat by elkeen van hierdie hotelle bly.

Hotel	AantalToeriste
Mount Grace Hotel	5
Peninsula Hotel	6
President Hotel	3
Village Lodge	5

(6)

- 1.5 Skryf 'n navraag wat die name sal lys van die toeriste wat nog nie vir enige ekskursies bespreek het nie. Vertoon die toeriste se name alfabeties.

ToerisNaam (eerste 10 rye)
Arron Haney
Cornall Prout
Darryl Poleykett
Davis Eginton
Eba Gillison
Eleen Yeomans
Irina Gouny
Johnath Nixon
Jorrie Potten
Kort McAndie
...

(6)

- 1.6 Ekskursiekodes word geskep deur die eerste drie karakters van die ekskursienaam met 'n ewekansige getal van 10 tot 99 te kombineer. Skryf 'n navraag om 'n ekskursiekode vir elke ekskursie te genereer.

EkskursieNaam	EkskursieKode <i>Die laaste twee syfers sal in jou geval anders wees omdat hulle ewekansig gegenereer word.</i>
Sunrise Breakfast River Cruise	Sun96
Sunset River Cruise	Sun56
Morning Safari	Mor32
Afternoon Safari	Aft31
Night Safari	Nig25
Township Excursion 1	Tow36
Township Excursion 2	Tow84
Township Excursion 3	Tow82
Township Excursion 4	Tow18
Township Excursion 5	Tow25
...	...

(6)

1.7 Al die toeriste wat by die "President Hotel" bly, gaan vandag op al die volgende ekskursies:

- Sunrise Breakfast River Cruise (EkskursieID: 1)
- Township Excursion 2 (EkskursieID: 7)
- National Art Museum Excursion 3 (EkskursieID: 13)
- Night Safari (EkskursieID: 5)

Skryf 'n navraag wat hulle vir hierdie ekskursies sal aanteken deur toepaslike inskrywings in die **BESPREKING**-tabel in te voeg.

Let op die volgende:

- Die koste kan bereken word deur die bybetaling by die huidige koste by te tel.
- Vandag se datum moenie vasgekodeer word nie (gebruik 'n toepaslike funksie om vandag se datum te bepaal).
- Die EkskursieID 1, 7, 13 en 5 sowel as die naam van die hotel kan vasgekodeer word. (Hierdie waardes mag deel vorm van die SQL-stelling.)

(9)

40 punte

AFDELING B OBJEKGEORIËNTEERDE PROGRAMMERING

SCENARIO

Die Stad Idayimani bedryf 'n oopbusdiens wat toeriste na verskeie punte van belang, genaamd **stoppe**, in die stad neem. Daar is verskeie bus**roetes** wat elkeen 'n aantal **stoppe** besoek. Sommige van hierdie stoppe word deur veelvuldige roetes gedeel.

Stoppe

'n Klas sal geskep moet word om 'n **stop** te verteenwoordig. 'n **Stop** het die volgende inligting:

- **StopNaam** – die naam van die **stop**
- **RoeteKodes** – die roete(s) waarop hierdie **stop** gelys is. Hierdie string bestaan uit B, R, P, Y en enige kombinasie daarvan in enige volgorde. 'n RoeteKodes-string "BR" beteken byvoorbeeld dat hierdie **stop** besoek word deur beide roete B en roete R. Sommige **stoppe** word nie gebruik nie en sal dus 'n blanko roete hê.
- **StopTipe** – elke **stop** kan een van vier tipes wees:
 - 1 – **Kafee**: Die **stop** het 'n klein winkeltjie wat koffie en eetgoed aan toeriste verkoop.
 - 2 – **Skuiling**: Daar is 'n kleinerige skuiling met 'n paar sitplekke. Die bus sal hier stop en vir ten minste drie minute wag.
 - 3 – **Snel**: Daar is geen sitarea nie, slegs 'n bietjie staanplek en geen dak nie. Die bus sal nie stop indien daar niemand is wat wil af- of opklim nie.
 - 4 – **Ander**: Die terugvaltype indien die tipe nog nie bekend is nie of ontbreek.

Die klas **Stop** het ook vier konstante klas- (statiese) veranderlikes om die vier verskillende stoptipes wat hierbo genoem is, te verteenwoordig. Hul waardes is 1 vir *Kafee*, 2 vir *Skuiling*, 3 vir *Snel* en 4 vir *Ander*.

Tekslêer

Die tekslêer *data.txt* bevat inligting oor al die **stoppe** wat deur al die **roetes** gedeel word. Elke reël bevat inligting oor een **stop**. Hier is die eerste tien reëls van hierdie tekslêer:

```
Waterfront,1,RB
St Monicas Cathedral,3,R
Conference Centre,2,R
CC,3,R
Market Square,2,RBY
Clock Tower,2,Y
Idayimani Museum,3,Y
Grande Hotel,5,BY
SA Heritage Museum,2,Y
Apartheid Museum,2,Y
```

Elke reël van die tekslêer bestaan uit die volgende inligting:

```
StopNaam,StopTipe,RoeteKodes
```

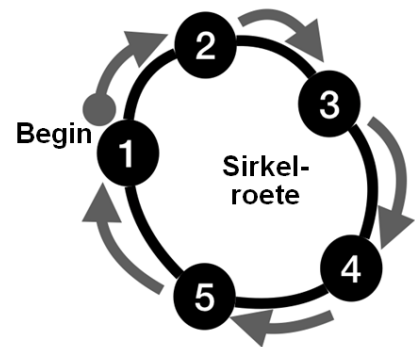
Die **stop** by Waterfront (in die eerste reël van die tekslêer gegee) is byvoorbeeld tipe 1 (wat ooreenstem met *Kafee*) en word deur roetes R en B besoek.

Roetes

'n Klas sal geskep moet word om **roetes** te verteenwoordig. 'n **Roete** bestaan uit veelvuldige **stoppe**. Daar is twee tipes **roetes**:

Sirkelroete

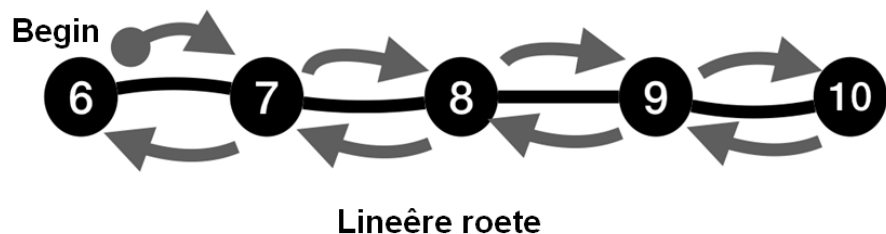
Sommige van die **roetes** vorm 'n *sirkel*, wat beteken dat die volgende **stop** na die laaste **stop** die oorspronklike vertrek**stop** is. 'n Bus wat byvoorbeeld 'n sirkel**roete** met **stoppe** 1, 2, 3, 4, 5 volg, sal hierdie patroon op sy roete volg: 1, 2, 3, 4, 5, 1, 2, 3, ... (sien regs).



Lineêre roete

Sommige van die roetes is *lineêr*, wat beteken dat sodra die laaste **stop** bereik is, die bus met dieselfde roete

terugkeer en al sy **stoppe** in omgekeerde volgorde besoek. 'n Lineêre roete met punte 6,7,8,9,10 sal byvoorbeeld hierdie patroon op sy roete volg: 6,7,8,9,10,9,8,7,6,7,8, ... (sien regs).



Elke **roete** het die volgende inligting:

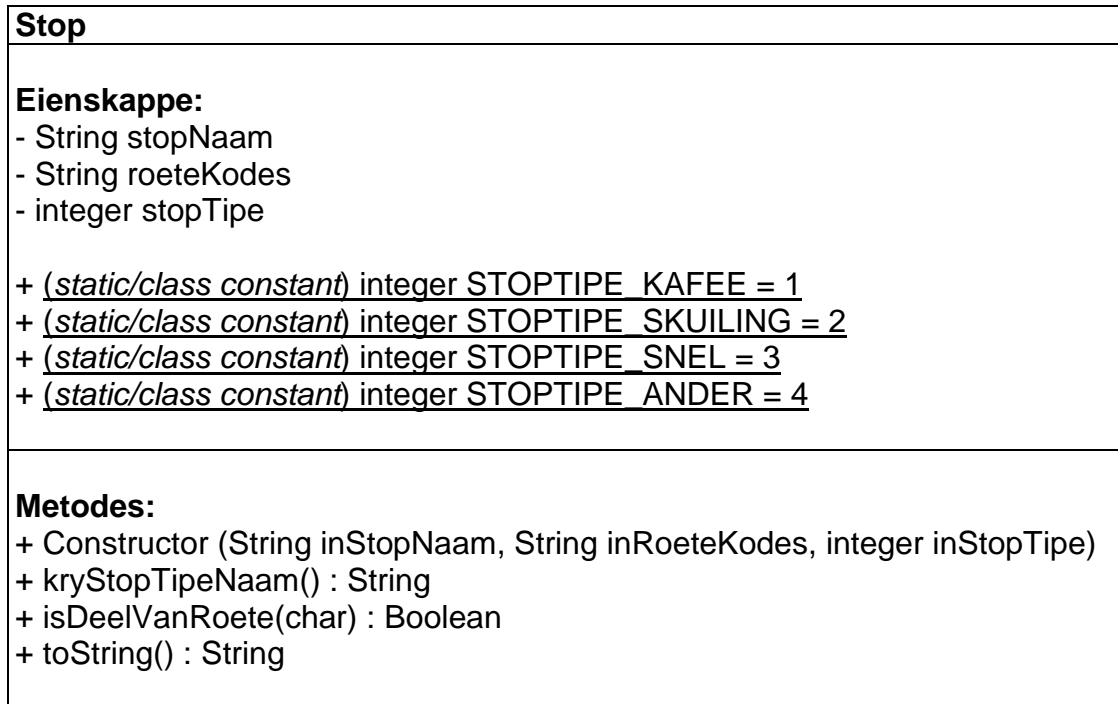
- **RoeteKode** – 'n enkele karakter wat die **roete** verteenwoordig.
- **Of die roete 'n sirkel is** – indien 'n **roete** nie 'n *sirkel* is nie, is dit *lineêr*.
- **'n Skikking van stoppe** wat deel vorm van hierdie **roete**. Hulle moet gelys word in die volgorde wat hulle besoek word en moet met die vertrek**stop** begin. Elke **stop** moet slegs een keer op die lys verskyn. Die skikking van stoppe lys slegs die stoppe wat vir hierdie roete van belang is, nie al die stoppe wat in die tekslêer *data.txt* gelys word nie.

Die **scenario** wat hierbo geskets is, moet gebruik word om **VRAAG 2 TOT VRAAG 7** te beantwoord.

Let op dat elke vraag op 'n nuwe bladsy begin.

VRAAG 2

Die klas **Stop** het die volgende VML-klasdiagram. Dit dui die eienskappe en metodes aan wat benodig word. Let sorgvuldig op die metode- en parametername in die diagram. **GEEN** bykomende *openbare* eienskap of metode moet geskep word nie. Jy mag egter enige bykomende *privaat* eienskap of metode skep soos nodig.



- 2.1 Skep 'n nuwe klas genaamd **Stop**. (1)
- 2.2 Skep drie eienskappe om **stopNaam**, **roeteKodes** en **stopTipe** te stoor. (3)
- 2.3 Skep die vier statiese/klaskonstantes STOPTIPE_KAFEE met die waarde 1, STOPTIPE_SKUILING met die waarde 2, STOPTIPE_SNEL met die waarde 3 en STOPTIPE_ANDER met die waarde 4. (3)
- 2.4 Skep 'n konstruktormetode wat twee stringe en 'n heelgetal as parameters aanvaar. Die eerste twee stringparameters verteenwoordig die naam van die **stop** en die kode(s) van die roetes wat hierdie **stop** besoek. Die heelgetal verteenwoordig die tipe van die **stop**.

Die konstruktorkonstruktor moet die volgende doen:

- Die twee stringparameters moet gebruik word om die waardes van die **stopNaam**- en **roeteKodes**-eienskappe toe te ken.
- Die heelgetal moet gebruik word om die stoptipe te verteenwoordig en moet een van die waardes van die statiese konstantes wees. Indien die gegewe heelgetalparameter nie een van die statiese konstantes is nie, moet die waarde van die **stopTipe**-eienskap op STOPTIPE_ANDER gestel word.

(7)

2.5 Skep 'n metode genaamd **kryStopTipeNaam()** wat die naam van die stoptipe terugstuur na gelang van die waarde van die **stopTipe**-eienskap:

- Indien die **stopTipe**-eienskap dieselfde waarde as STOPTIPE_KAFEE het, moet die metode "kafee" terugstuur.
- Indien die **stopTipe**-eienskap dieselfde waarde as STOPTIPE_SKUILING het, moet die metode "skuiling" terugstuur.
- Indien die **stopTipe**-eienskap dieselfde waarde as STOPTIPE_SNEL het, moet die metode "snel" terugstuur.
- Vir enige ander waarde vir die **stopTipe**-eienskap moet die metode "ander" terugstuur.

(3)

2.6 Skep die metode **isDeelVanRoete(..)** wat 'n karakter inneem. Die metode moet 'n Boole-waarde terugstuur om aan te dui of hierdie **stop** deel is van die roete wat deur die gegewe karakter verteenwoordig word of nie.

(3)

2.7 Skep 'n **toString()**-metode wat gebruik kan word om 'n string-verteenwoordiging van die **Stop**-instansie af te voer wat die **stopNaam** en die stoptipenaam in hierdie formaat insluit:

<Stoptipenaam><tab><Stopnaam>

Byvoorbeeld:

kafee Waterfront

(3)
[23]

VRAAG 3

Gebruik die klasdiagram hieronder om 'n nuwe klas genaamd **Roete** te skep. Hierdie klas sal gebruik word om die besonderhede van 'n busroete te stoor. Die diagram hieronder dui die eienskappe en metodes aan wat benodig word. **GEEN** bykomende *openbare* eienskap of metode moet geskep word nie. Jy mag egter enige bykomende *privaat* eienskap of metode skep soos nodig.

Roete	
Eienskappe: - char roeteKode - boolean isSirkel Let wel: Dit is <i>true</i> vir sirkelroete en <i>false</i> vir lineêre roete. - Stop[] stoppe	
Metodes: + Constructor(char roeteKode, boolean isSirkel) + stelStoppe(Stop[] s) + kryRoeteKode() : char + kryStopBy(int num) : Stop + toString() : String	

- 3.1 Skep 'n nuwe klas genaamd **Roete** met die drie eienskappe wat hierbo aangedui word. (3)
- 3.2 Skep 'n konstruktormetode wat een karakter aanvaar as 'n parameter wat die **roeteKode** verteenwoordig en 'n Boole-waarde wat aandui of die **roete** 'n sirkel is of nie. Die konstruktorklasse moet die gegewe parameters gebruik om die **roeteKode**- en **isSirkel**-eienskap van die **Roete**-instansie te stel. Die eienskap **stoppe** MOET NIE deur die konstruktorklasse gestel word nie, aangesien dit deur die **stelStoppe(..)**-metode gestel sal word. (2)
- 3.3 Skep 'n mutator- (stel-) metode vir die **stoppe**-eienskap wat 'n skikking van Stoppe as 'n parameter aanvaar. (2)
- 3.4 Skep 'n krymetode vir die eienskap **roeteKode**. (1)
- 3.5 Skep 'n metode genaamd **kryStopBy(..)**. Die metode sal die posisie van 'n stop as 'n heelgetal aanvaar en die ooreenstemmende **Stop**-objek in die skikking terugstuur. Die metode moet *nul* terugstuur indien die posisie wat gegee word nie geldig is nie. Indien die skikking byvoorbeeld slegs 5 stoppe het, sal 8 nie geldig wees nie en 'n *nul*-objek moet teruggestuur word. (5)

Die **toString()**-metode van die **Roete**-klas sal later in Vraag 5 voltooi word.

[13]

VRAAG 4

4.1 Skep 'n klas genaamd **ToerBestuurder**. (1)

4.2 Verklaar in hierdie klas die volgende skikking en tellerveranderlike as instansieveranderlikes:

- 'n Skikking wat gebruik kan word om tot 100 **Stop**-objekte te stoor.
- 'n Tellerveranderlike om op die spoor te bly van die aantal **Stop**-objekte wat gestoor is.

(3)

4.3 Skep 'n konstruktormetode wat die inhoud van 'n tekslêer wat die inligting van **AL** die stoppe bevat, sal lees. Die konstruktorklass moet 'n string wat die lêernaam verteenwoordig, inneem. Elke reël wat uit die tekslêer ingelees word, sal daartoe lei dat EEN **Stop**-objek by die skikking hierbo bygevoeg word.

Doen die volgende in die konstruktormetode:

- Maak die lêer oop om te lees. Jy kan aanneem dat die lêer bestaan.
- Lees deur die lêer totdat daar nie nog reëls is nie. In elke herhaling van die lus:
 - Lees elke reël in en verdeel die **Stop**-data in afsonderlike items.
 - Skep 'n **Stop**-objek met die inligting wat op elke reël aanwesig is.
 - Voeg die nuutgeskepte **Stop**-objek by die skikking en werk die tellerveranderlike toepaslik by.

(9)

4.4 Skep 'n metode genaamd **kryRoeteMetKode(..)** wat 'n karakter kan inneem wat 'n **roeteKode** verteenwoordig en 'n Boole-waarde wat aandui of die roete 'n sirkel is of nie. Hierdie metode moet 'n nuwe **Roete**-objek met die ooreenstemmende kenmerke terugstuur. Die **Roete**-objek se Stop-skikking moet SLEGS die **stoppe** bevat wat aan daardie **roete** behoort in die volgorde waarin hulle in die tekslêer verskyn.

Let daarop dat punte vir doeltreffendheid toegeken sal word. Om weer uit die tekslêer te lees in plaas daarvan om uit die bestaande skikking te werk, is byvoorbeeld nie doeltreffend nie.

Let ook daarop dat geen nuwe *openbare* eienskappe of *openbare* metodes in enige ander klasse geskep moet word nie. Jy mag egter bykomende *privaat* eienskappe en/of *privaat* metodes skep wat jy nodig ag.

(10)
[23]

VRAAG 5

In die volgende formate, na gelang daarvan of die **roete** 'n sirkel is of lineêr, sal jy nou die **toString()**-metode vir die **Roete**-klas skep wat gebruik kan word om 'n stringverteenvoording van 'n **Roete**-instansie te verkry. Alle roetes met geen stoppe nie moet as ongeldige roetes hanteer word.

Let daarop dat punte vir doeltreffendheid toegeken sal word.

(9)

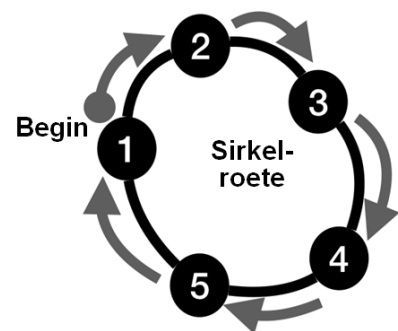
Formaat vir ongeldige roete

<Roetekode> - Ongeldig

Formaat vir sirkelroete

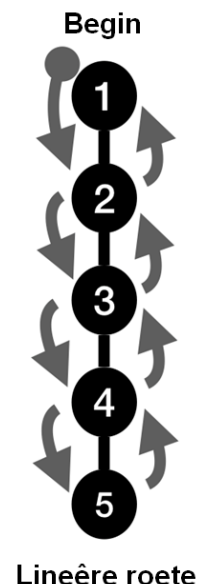
(Voorbeeld hieronder het 5 stoppe):

```
<Roetekode> - Sirkel
-> 1 <Stop 1 in stoppe' toString-formaat>
-> 2 <Stop 2 in stoppe' toString-formaat>
-> 3 <Stop 3 in stoppe' toString-formaat>
-> 4 <Stop 4 in stoppe' toString-formaat>
-> 5 <Stop 5 in stoppe' toString-formaat>
-> 1 <Stop 1 in stoppe' toString-formaat>
```

**Lineêre roete**

(Voorbeeld hieronder het 5 stoppe):

```
<Roetekode> - Lineêr
-> 1 <Stop 1 in stoppe' toString-formaat>
-> 2 <Stop 2 in stoppe' toString-formaat>
-> 3 <Stop 3 in stoppe' toString-formaat>
-> 4 <Stop 4 in stoppe' toString-formaat>
-> 5 <Stop 5 in stoppe' toString-formaat>
-> 4 <Stop 4 in stoppe' toString-formaat>
-> 3 <Stop 3 in stoppe' toString-formaat>
-> 2 <Stop 2 in stoppe' toString-formaat>
-> 1 <Stop 1 in stoppe' toString-formaat>
```



[9]

VRAAG 6

- 6.1 Skep 'n eenvoudige gebruikerskoppelvlak genaamd **ToerGK** wat eenvoudige afvoer moontlik sal maak. (1)
- 6.2 Verklaar en instansieer 'n **ToerBestuurder**-objek op die toepaslike plek in die kode. Die tekslêer *data.txt* moet gebruik word om die objek te instansieer. (1)
- 6.3 Skryf kode om twee **Roete**-objekte uit die **ToerBestuurder** te verkry deur die metode **kryRoeteMetKode(..)** te gebruik wat in Vraag 4.4 geskep is. Die eerste **roete** is 'n sirkelroete met roetekode "R" en die tweede is 'n lineêre roete met kode "Y". (2)
- 6.4 Skryf kode om beide roetes in hul **toString()**-formaat te druk. (1)
- [5]**

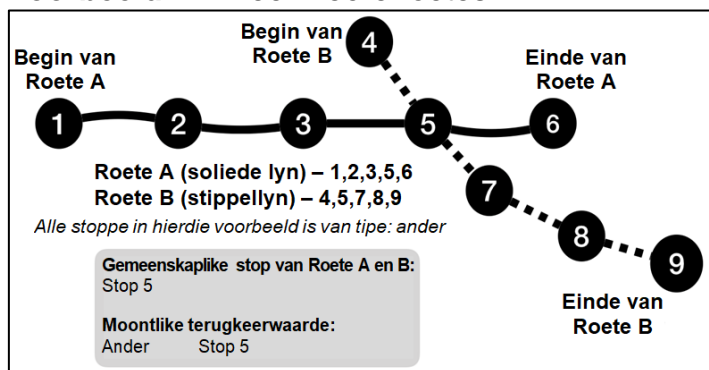
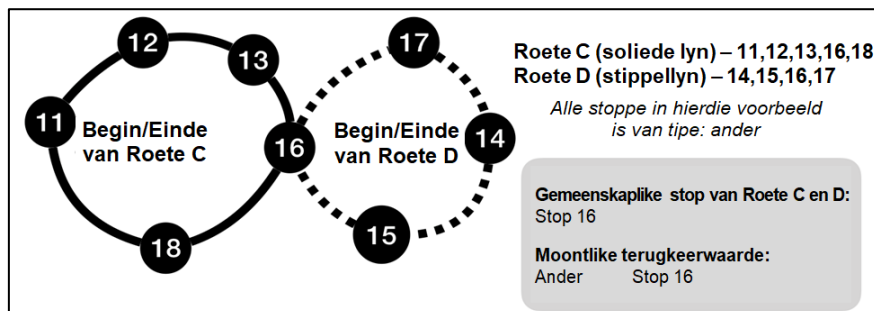
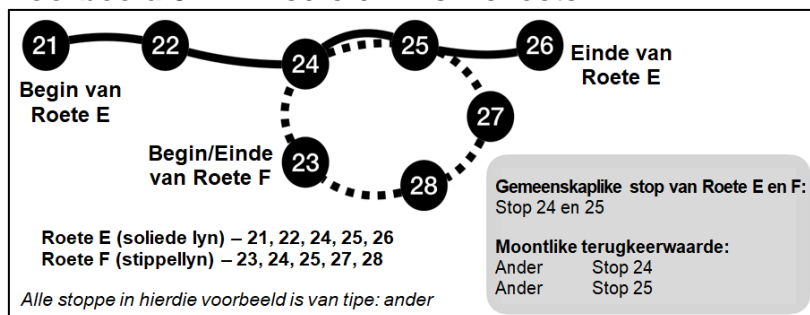
LW: BLAAI ASSEBLIEF OM VIR VRAAG 7

VRAAG 7

- 7.1 Skep in die **ToerBestuurder**-klas 'n geskikte metode wat twee **Roete**-objekte as parameters inneem. Die metode sal die **stoppe** uitwerk wat gemeenskaplik is aan beide **roetes**. Die metode moet 'n string terugstuur wat die gemeenskaplike **stoppe** op afsonderlike reëls lys. Elke gemeenskaplike **stop** moet EEN KEER in die **Stop**-klas se **toString()**-formaat ingesluit word.

(6)

Hier is 'n paar voorbeelde:

Voorbeeld 1: Twee lineêre roetes**Voorbeeld 2: Twee sirkelroetes****Voorbeeld 3: 'n Lineêre en 'n sirkelroete**

- 7.2 Gebruik in die **ToerGK**-klas die metode wat in Vraag 7.1 geskep is om die gemeenskaplike **stop(pe)** te druk wat gedeel word deur die twee **Roete**-objekte wat in Vraag 6.3 geskep is.

(1)
[7]

80 punte

Totaal: 120 punte