



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

SENIORSERTIFIKAAT-EKSAMEN

INLIGTINGSTEGNOLOGIE V1

2016

PUNTE: 150

TYD: 3 uur

Hierdie vraestel bestaan uit 19 bladsye.

INSTRUKSIES EN INLIGTING

1. Hierdie vraestel is in DRIE afdelings ingedeel. Kandidate moet AL DRIE afdelings beantwoord.
2. Die duur van hierdie eksamen is drie uur. As gevolg van die aard van hierdie eksamen is dit belangrik om daarop te let dat jy nie toegelaat sal word om die eksamenlokaal voor die einde van die eksamensessie te verlaat nie.
3. Hierdie vraestel is opgestel met programmeringsterme wat nie programmeringstaal-spesifiek is nie (Delphi/Java (deur die Netbeans IDE te gebruik)).
4. Maak seker dat jy die vrae beantwoord volgens die spesifikasies wat in elke vraag gegee word. Punte sal volgens die voorgeskrewe vereistes toegeken word.
5. Beantwoord slegs wat in elke vraag gevra word. Byvoorbeeld, indien die vraag nie datavalidering vra nie, sal geen punte vir datavalidering toegeken word nie.
6. Jou programme moet op só 'n manier gekodeer word dat dit met enige data sal werk en nie net met die voorbeelddata wat voorsien is of enige data-uittreksels wat in die vraestel verskyn nie.
7. Roetines, soos soek, sorteer en seleksie, moet vanuit eerste beginsels ontwikkel word. Jy mag NIE die ingeboude funksies van 'n programmeringstaal vir enige van hierdie roetines gebruik NIE.
8. Datastrukture wat nie voorsien is nie, moet deur jou, die programmeerder, verklaar word.
9. Jy moet jou werk gereeld stoor op die disket/CD/DVD/geheuestokkie wat aan jou gegee is of op die skyfspasie wat vir hierdie eksamensessie aan jou toegeken is.
10. Maak seker dat jou eksamennummer as kommentaar verskyn in elke program wat jy kodeer, asook op elke gebeurtenis wat aangedui word.
11. Indien dit vereis word, druk die programmeringskode van al die programme/klasse wat jy voltooi het. Jy sal ná die eksamensessie 'n halfuur tyd vir drukwerk gegun word.
12. Aan die einde van hierdie eksamensessie moet jy 'n disket/CD/DVD/geheuestokkie inlewer met al jou werk daarop gestoor OF jy moet seker maak dat al jou werk op die skyfspasie gestoor is wat vir hierdie eksamensessie aan jou toegeken is. Maak seker dat al die lêers gelees kan word.

13. Die lêers wat jy benodig om hierdie vraestel te voltooi, is aan jou gegee op 'n disket/CD/DVD/geheuestokkie of op die skyfspasie wat aan jou toegeken is. Die lêers word in die vorm van wagwoordbeskermdede uitvoerbare lêers verskaf.

LET WEL:

- Delphi-kandidate moet die lêer **DelphiDataAFRJunie2016.exe** gebruik.
- Java-kandidate moet die lêer **JavaDataAFRJunie2016.exe** gebruik.

Doen die volgende:

- Dubbelklik op die wagwoordbeskermdede uitvoerbare lêer.
- Klik op die 'Extract'-knoppie.
- Sleutel die volgende wagwoord in: **Hosp9@%\$**

Nadat dit onttrek ('extracted') is, sal die volgende lys met lêers in die lêergids **DelphiDataAFRJunie2016/JavaDataAFRJunie2016** beskikbaar wees:

DELPHI-LÊERS**Vraag1:**

Vraag1_P.dpr
Vraag1_P.res
Vraag1_U.dfm
Vraag1_U.pas

Vraag2:

VoorskrifItem.pas
Vraag2_P.dpr
Vraag2_P.res
Vraag2_U.dfm
Vraag2_U.pas

Vraag3:

Vraag3_P.dpr
Vraag3_P.res
Vraag3_U.dfm
Vraag3_U.pas
DataV3.txt

JAVA (NETBEANS)-LÊERS**Vraag1:**

Vraag1.form
Vraag1.java

Vraag2:

VoorskrifItem.java
Vraag2.form
Vraag2.java

Vraag3:

Vraag3.form
Vraag3.java
DataV3.txt

SCENARIO

Die Goeie Gesondheid Mediese Sentrum verskaf hoogs opgeleide dokters en mediese dienste aan die publiek.

AFDELING A**VRAAG 1: ALGEMENE PROGRAMMERINGSVAARDIGHEDE**

Die mediese sentrum het 'n aantal verskillende fasiliteite wat vir die gemeenskap beskikbaar is.

Doen die volgende:

- Kompileer en voer die program uit wat in die **Vraag1**-lêergids voorkom. Die koppelvlak vertoon vier verskillende afdelings wat Vraag1_1 tot Vraag1_4 genoem word. Die program het tans geen funksionaliteit nie.
- Voltooi die kode vir elke afdeling van VRAAG 1, soos wat in VRAAG 1.1 tot VRAAG 1.4 op die volgende bladsy beskryf word.

Voorbeeld van grafiese gebruikerskoppelvlak (GGK) ('GUI'):

The screenshot displays a graphical user interface (GUI) for a medical center system, organized into four distinct question panels labeled Vraag1_1 through Vraag1_4.

- Vraag1_1:** Contains the instruction "Vertoon die naam hier" (Display the name here).
- Vraag1_2:** Features a list of "Mediesehulpskema-opsies" (Medical aid scheme options) including Opsie A, Opsie B, Opsie C, and Opsie D. It also includes a "Getal afhanklikes" (Number of dependents) input field and a "Vraag1_2" button.
- Vraag1_3:** Includes a "Sleutel die balans van die rekening in" (Key the balance of the account in) input field and a "Vraag1_3" button.
- Vraag1_4:** Contains a "Sleutel die getal pasiënte in" (Key the number of patients in) input field, a "Dokters aan diens" (Doctors on duty) section with three input fields for "Dokter 1", "Dokter 2", and "Dokter 3", and a "Getal pasiënte" (Number of patients) input field. It also features a "Minute per pasiënt" (Minutes per patient) input field set to 15, a "Begintyd" (Start time) section with radio buttons for 8h00 (selected), 11h00, and 13h00, and two input fields for "Tyd by die werk" (Time at work) and "Uittekentyd" (Dismissal time).

- 1.1 Skryf kode om die fontgrootte van die teks op die byskrif **IbIVraag1_1** op 14 pt te stel en vertoon die teks 'Goeie Gesondheid Mediese Sentrum' op die byskrif wanneer die program uitgevoer word.



(2)

1.2 Knoppie [Vraag1_2]

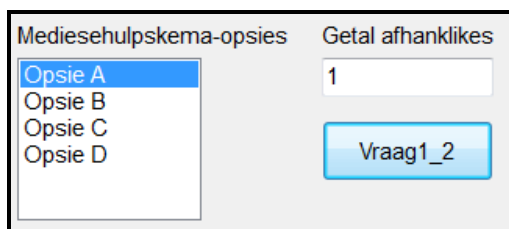
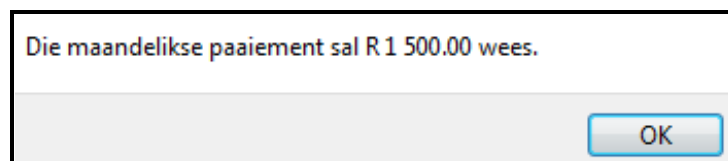
Die gemeenskap word aangemoedig om tot 'n mediese hulpskema by te dra. Die skema bied vier opsies: Opsie A tot Opsie D. Opsie A verskaf die minimum hoeveelheid voordele en is die goedkoopste opsie met 'n bedrag van R1 000,00 per maand as bydrae vir die hooflid van 'n gesin. Hierdie bedrag neem met 20% toe vir elke opsie (Opsie B, Opsie C en Opsie D) wat volg.

Gesinslede van die hooflid word as afhanklikes by die rekening van die hooflid gevoeg. 'n Bedrag van 50% van die hooflid se paaieiment sal vir elke afhanklike by sy/haar rekening gevoeg word.

Die gebruiker moet 'n opsie uit die lysblokkie ('list box') kies en die getal afhanklikes in die teksblokkie ('text box') insleutel.

Skryf kode om die totale maandelikse paaieiment wat die lid tot die mediese hulpskema moet bydra, te bereken. Vertoon die maandelikse paaieiment in 'n dialoogblokkie ('dialog box') met 'n geskikte boodskap. As geen opsie gekies is nie, vertoon 'n boodskap wat die gebruiker die instruksie gee om 'n opsie te kies.

Voorbeeld van afvoer as Opsie A gekies word en die getal afhanklikes een is:

Voorbeeld van afvoer as Opsie B gekies word en die getal afhanklikes drie is (op die volgende bladsy):

Mediese hulpskema-opsies Getal afhanklikes

Opsie A
Opsie B
Opsie C
Opsie D

3

Vraag1_2

Die maandelikse paaierement sal R 3 000.00 wees.

OK

(10)

1.3 **Knoppie [Vraag1_3]**

Die pasiënt se rekening kan oor 'n periode van 12 maande betaal word.

Die eerste paaierement word as 15% van die rekeningsaldo bereken. Nadat die eerste paaierement afgetrek is, moet die balans in gelyke paaieremente oor die res van die maande afbetaal word.

Die gebruiker moet die rekening se balans in die teksblokkie insleutel.

Skryf kode om die nommer van die paaierement, die bedrag wat maandeliks betaal moet word en die afnemende balans te bereken en in netjies kolomme met geskikte opskrifte te vertoon. Die bedrae moet as 'n geldeenheid met twee desimale plekke geformateer word.

Sleutel die balans van die rekening in

4500 Vraag1_3

Paaierement	Maandelikse Paaierement	Balans
1	R 675.00	R 3 825.00
2	R 347.73	R 3 477.27
3	R 347.73	R 3 129.55
4	R 347.73	R 2 781.82
5	R 347.73	R 2 434.09
6	R 347.73	R 2 086.36
7	R 347.73	R 1 738.64
8	R 347.73	R 1 390.91
9	R 347.73	R 1 043.18
10	R 347.73	R 695.45
11	R 347.73	R 347.73
12	R 347.73	R 0.00

(10)

- 1.4 Daar is elke dag drie dokters aan diens: Dokter 1, Dokter 2 en Dokter 3. Die totale getal pasiënte moet gelykop tussen die drie dokters verdeel word. As daar enige oorblywende pasiënte is, moet die eerste pasiënt aan die eerste dokter (Dokter 1) toegeken word en die tweede pasiënt aan die tweede dokter (Dokter 2).

Die algoritme hieronder is saamgestel om die getal pasiënte wat elke dokter deur die dag sal spreek, te bereken en te vertoon.

Algoritme:

```

1. getal ← Sleutel die totale getal pasiënte in
2. pasiënte_per_dokter ← getal heelgetaldeling deur 3
3. dokter1 ← pasiënte_per_dokter
4. dokter2 ← pasiënte_per_dokter
5. dokter3 ← pasiënte_per_dokter
6. res ← getal modulus 3
7. as res = 1 dan
    dokter1 ← dokter1 + 1
   as res = 2 dan
    dokter2 ← dokter2 + 1
8. vertoon dokter1, dokter2 en dokter3

```

- 1.4.1 (a) Skakel die instruksies wat in die algoritme hierbo verskaf is om na programmeringskode om die getal pasiënte per dokter te bereken en te vertoon. (10)
- (b) Die verkeerde getal pasiënte word aan die dokters toegeken as gevolg van 'n logikafout in een van die IF-stellings in die algoritme. Identifiseer die foutiewe IF-stelling en korrigeer dit in jou kode sodat die korrekte getal pasiënte aan elke dokter toegeken word.

Voorbeeld van die korrekte afvoer vir 'n totaal van vyf pasiënte:

Sleutel die getal pasiënte in	<input type="text" value="5"/>
Dokters aan diens	Getal pasiënte
Dokter 1	<input type="text" value="2"/>
Dokter 2	<input type="text" value="2"/>
Dokter 3	<input type="text" value="1"/>
<input type="button" value="Vraag1_4"/>	

Voorbeeld van korrekte afvoer vir 'n totaal van tien pasiënte (op die volgende bladsy):

Sleutel die getal pasiënte in	<input type="text" value="10"/>
Dokters aan diens	Getal pasiënte
Dokter 1	<input type="text" value="4"/>
Dokter 2	<input type="text" value="3"/>
Dokter 3	<input type="text" value="3"/>
<input type="button" value="Vraag1_4"/>	

(3)

- 1.4.2 Die dokters wat aan diens is, moet wag totdat al die pasiënte op die lys aandag geniet het voordat hulle as 'n span kan uitteken. Hulle moet die getal ure en minute wat hulle gewerk het, verskaf deur die getal ure en minute wat die eerste dokter gewerk het, te gebruik. Hulle mag slegs op die uur uitteken. Byvoorbeeld, as hulle vir 3 uur en 30 minute gewerk het en 08:00 begin werk het, sal hulle almal om 12:00 uitteken.

Voeg kode by die VRAAG1_4-knoppie om die volgende te doen:

Verkry die duur van een konsultasie in minute. Die verstek ('default')-tyd is 15 minute, maar die gebruiker kan enige ander waarde insleutel. Verkry die tyd van die dag wanneer konsultasie begin. Bereken en vertoon die tyd wat die dokters gewerk het en die tyd van die dag wanneer hulle sal uitteken.

Voorbeeld van afvoer as die totale getal pasiënte vyf is, die duur van 'n konsultasie 15 minute is en konsultasie om 08:00 begin:

Sleutel die getal pasiënte in	<input type="text" value="5"/>
Dokters aan diens	Getal pasiënte
Dokter 1	<input type="text" value="2"/>
Dokter 2	<input type="text" value="2"/>
Dokter 3	<input type="text" value="1"/>
<input type="button" value="Vraag1_4"/>	
Minute per pasiënt	<input type="text" value="15"/>
Begintyd	<input checked="" type="radio"/> 8h00 <input type="radio"/> 11h00 <input type="radio"/> 13h00
Tyd by die werk	<input type="text" value="0 uur 30 minute"/>
Uittekentyd	<input type="text" value="9h00"/>

Voorbeeld van afvoer as die totale getal pasiënte tien is, die duur van 'n konsultasie 20 minute is en konsultasie om 13:00 begin:

Sleutel die getal pasiënte in	<input type="text" value="10"/>
Dokters aan diens	Getal pasiënte
Dokter 1	<input type="text" value="4"/>
Dokter 2	<input type="text" value="3"/>
Dokter 3	<input type="text" value="3"/>
<input type="button" value="Vraag1_4"/>	
Minute per pasiënt	<input type="text" value="20"/>
Begintyd	<input type="radio" value="8h00"/> 8h00 <input type="radio" value="11h00"/> 11h00 <input checked="" type="radio" value="13h00"/> 13h00
Tyd by die werk	<input type="text" value="1 uur 20 minute"/>
Uittekentyd	<input type="text" value="15h00"/>

(10)

- Sleutel jou eksamennommer as kommentaar in die eerste reël van die programlêer in.
- Stoor jou program.
- Maak 'n drukstuk van die kode indien dit vereis word.

TOTAAL AFDELING A: 45

AFDELING B

VRAAG 2: OBJEK-GEORIËNTEERDE PROGRAMMERING

Die Goeie Gesondheid Apteek stel aptekers in staat om die voorskrif wat die pasiënt vir pille ontvang het, elektronies in te sleutel. Die etikette vir die pilhouers en 'n pasiënt se rekening sal dan geskep moet word. Die totale koste van al die voorskrifte wat vereis word, moet dan bereken word.

Doen die volgende:

DELPHI-PROGRAMMEERDERS	JAVA-PROGRAMMEERDERS
<ul style="list-style-type: none"> • Maak die onvoltooide program wat in die Vraag2-lêergids gevind word, oop. • Maak die onvoltooide objekklas Voorskriftem.pas oop. • Sleutel jou eksamennommer as kommentaar in die eerste reël van beide die lêers Vraag2_U.pas en Voorskriftem.pas in. 	<ul style="list-style-type: none"> • Maak die onvoltooide program wat in die Vraag2-lêergids gevind word, oop. • Maak die onvoltooide objekklas Voorskriftem.java oop. • Sleutel jou eksamennommer as kommentaar in die eerste reël van beide die klasse Vraag2.java en Voorskriftem.java in.

- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie.

Voorbeeld van die koppelvlak:

Goeie Gesondheid Apteek

Selekteer pil se kode

▼

Voorskriftem

Dosiskode

▼

Getal dae

2.2.2 - Instansieer Voorskriftem

2.2.3 - Vertoon

2.2.4 - Totaal van alle Voorskriftem's

2.2.5 - Maak alles skoon

Voorskrifrekening

LET WEL: Die **pnlGeneries**-paneel is gestel om nie sigbaar te wees as die program uitgevoer word nie.

- Voltooi die kode vir hierdie program soos gespesifiseer in VRAAG 2.1 en VRAAG 2.2 hieronder.

2.1 Die gegewe onvolledige objekklas (**TVoorskrifitem/Voorskrifitem**) bevat die verklaring van vier attribute wat die **Voorskrifitem**-objek beskryf.

Die gegewe attribute vir die **Voorskrifitem**-objek is soos volg:

NAME VAN ATTRIBUTE		BESKRYWING
Delphi	Java	
fKode	kode	'n Unieke ses-karakter-kode wat aan elke voorgeskrewe pil toegeken is
fDosis	dosis	'n String wat uit vier karakters bestaan om aan te dui hoeveel pille geneem moet word (Sien die verduideliking in die nota hieronder.)
fDae	dae	Die voorgeskrewe getal dae wat die pille geneem moet word
fPrys	prys	Die prys van 'n enkele pil

Voltooi die kode in die gegewe **Voorskrifitem**-klas (**TVoorskrifitem/PVoorskrifitem**) soos wat in VRAAG 2.1.1 tot VRAAG 2.1.6 hieronder beskryf word.

LET WEL: Die dosis-attribuut bestaan uit vier karakters in die formaat XX#X, waar die # gebruik word as 'n simbool om inligting te skei.

Voorbeeld 1: D1#B

Voorbeeld 2: H2#8

Eerste karakter	Verduideliking
D	Pille moet een keer per dag geneem word. Tweede karakter: Getal pille wat op 'n slag geneem moet word Vierde karakter: B: Na ontbyt S: Na aandete Voorbeeld: D1#B beteken: Neem een pil daagliks na ontbyt.
H	Pille moet uurliks geneem word. Tweede karakter: Getal pille wat op 'n slag geneem moet word Vierde karakter: Verwys na die uurlikse periode wat 'n pil geneem moet word. Die uurlikse periode kan 2, 3, 4, 6 of 8 wees. Voorbeeld: H2#8 beteken: Neem twee pille elke 8 uur.

2.1.1 Skryf kode vir 'n **konstruktor**-metode om die pil se kode, dosiskode, getal dae en die prys per pil as parameterwaardes te ontvang. Ken hierdie waardes aan die relevante attribute toe. (4)

2.1.2 Skryf 'n metode met die naam **berekenPillePerDag** wat die dosiskode sal gebruik om die getal pille wat per dag voorgeskryf word, te bepaal en terug te stuur. 'n Dag word as 'n periode van 24 uur beskou.

Voorbeeld 1: As die dosiskode D1#S is, beteken dit dat een pil daagliks na aandete geneem moet word.

Voorbeeld 2: As die dosiskode H2#4 is, beteken dit dat twee pille elke vier uur geneem moet word, wat 'n totaal van 12 pille per dag uitmaak. (8)

2.1.3 Skryf 'n metode met die naam **stelsaamGereeldheidBoodskap** wat die dosiskode sal gebruik om 'n boodskap saam te stel en terug te stuur wat aandui wanneer die pille geneem moet word.

Voorbeeld 1: As die dosiskode D1#S is, is die boodskap wat teruggestuur moet word:
Neem 1 pil(le) daagliks na aandete.

Voorbeeld 2: As die dosiskode H4#8 is, is die boodskap wat teruggestuur moet word:
Neem 4 pil(le) elke 8 uur. (8)

2.1.4 Skryf 'n metode met die naam **berekenTotaalPille** om die totale getal pille vir hierdie voorskrifitem te bereken en terug te stuur. (3)

2.1.5 Skryf **toegangs('accessor')**-metodes vir die attribute van die pil se kode en prys. (2)

2.1.6 Skryf 'n **stelsaamEtiket**-metode om die nodige inligting terug te stuur wat op die etiket van die pilhouer gedruk moet word, in die formaat hieronder getoon.

Pil se kode: <kode>

<Gereeldheid van gebruik boodskap> vir <getal dae> dae

(<Totale getal pille ingesluit> pille)

Voorbeeld van afvoer as die pil se kode HPCMCN, die dosiskode H2#8 en die getal dae vier is:

Pil se kode: HPCMCN

Drink 2 pil(le) elke 8 uur vir 4 dae.

(24 pille)

(5)

2.2 Twee parallelle skikkings word in die onvoltooide klas **Vraag2_U.pas/Vraag2.java** gegee.

Die skikking met die naam **arrKodes** bevat die kodes van al die beskikbare pille. Die skikking met die naam **arrPryse** bevat die pryse per enkele pil vir elke ooreenstemmende kode.

Voorbeeld van data in die skikkings wat verskaf is:

Die eerste vier kodes in die **arrKodes**-skikking:

CBLTAD, HPCMCN, TRVKMG, BGJKPT

Die eerste vier pryse in die **arrPryse**-skikking:

17.89, 14.60, 23.50, 22.75

Die data vir die eerste pil kan soos volg geïnterpreteer word:

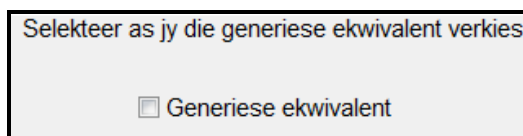
- Die kode van die pil is **CBLTAD**.
- Die prys van 'n enkele pil is **17.89**.

Doen die volgende om die kode te voltooi vir elke knoppie in die hoofvormeenheid (Delphi)/GGK-klas (Java):

LET WEL: Die objek **objVoorskrifItem** is in die program verklaar.

2.2.1 Kode vir die kombinasieels ('combo box')

Die gebruiker moet die kode van die pil uit die relevante kombinasieels selekteer. As die kode wat geselekteer is, met die letter 'G' eindig, beteken dit dat daar 'n generiese ekwivalent vir die pil beskikbaar is. Die **pnlGeneries**-paneel wat die gebruiker toelaat om die generiese ekwivalent van die pil te kies, moet vertoon word.



'n Generiese pil word gedefinieer as 'n pil wat gebruik word om dieselfde siekte te behandel as die oorspronklike pil, maar wat gewoonlik teen 'n laer prys verkoop word.

In hierdie program is die koste van die generiese ekwivalent 40% laer as die koste van die oorspronklike pil. Die **arrPryse**-skikking bevat die pryse van die oorspronklike pille.

Die **pnlGeneries**-paneel moet nie sigbaar wees as daar geen generiese ekwivalent vir die pil beskikbaar is nie.

WENK: Die indekse van die kodes in die **arrKodes**-skikking is dieselfde as dié van die kodes in die kombinasieels.

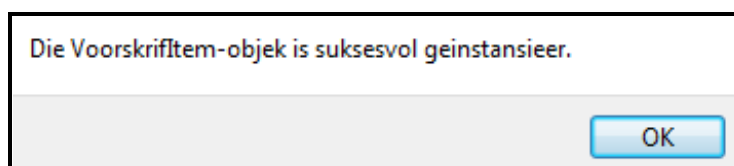
(9)

2.2.2 Knoppie [2.2.2 – Instansieer Voorskrifitem]

Gebruik die inligting wat deur die gebruiker verskaf is en instansieer 'n nuwe **Voorskrifitem**-objek. Die pil se kode, dosiskode, getal dae en die prys per pil moet as argumente gebruik word om die objek te instansieer.

As die **pnlGeneries**-paneel verskyn en die generiese kontroleblokkie ('check box') word geselekteer, moet die pil se prys met 40% verlaag word voordat die objek geïnstansieer word.

Vertoon 'n boodskap in 'n dialoogblokkie wat aandui dat die objek suksesvol geïnstansieer is.



(8)

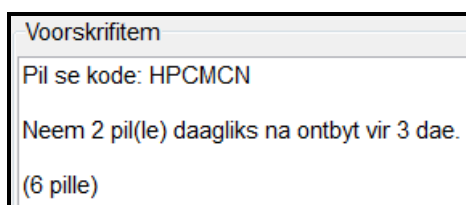
2.2.3 Knoppie [2.2.3 – Vertoon]

Gebruik die **ontwerpEtiket**-metode om die etiket se inligting in die afvoerarea op die **Voorskrifitem**-paneel te vertoon.

Vertoon ook die pil se kode en die koste van hierdie voorskrifitem in die **Voorskrifrekening**-paneel se afvoerarea. Die koste moet na 'n geldeenheid met twee desimale plekke geformateer wees.

LET WEL: Die totale koste van al die voorskrifitems moet bereken word sodat dit vertoon word as die **2.2.4 – Totaal van alle Voorskrifitems**-knoppie geklik word.

Voorbeeld van afvoer vir die voorskrifitem en rekening-besonderhede as die pil se kode HPCMCN, die dosiskode D2#B en die getal dae wat ingesleutel is, 3 is:



(8)

2.2.4 Knoppie [2.2.4 – Totaal van al die Voorskrifitems]

Vertoon die totale bedrag wat vir hierdie spesifieke voorskrif verskuldig is, geformateer na 'n geldeenheid met twee desimale plekke.

Voorbeeld van afvoer vir die **Voorskrifrekening**-afvoerarea nadat al die voorskrifitems ingesleutel is:

Voorskrifrekening	
HPCMCN	R87.60
PLOMNG	R575.06
BGHYTR	R845.64
Totaal:	R1,508.30

(2)

2.2.5 Knoppie [2.2.5 – Maak alles skoon]

Maak die afvoerarea vir die voorskrifitem en die voorskrifrekening skoon. Maak ook al die toevoerkomponente skoon en stel die relevante veranderlikes op 0.

(3)

- Sleutel jou eksamennommer as kommentaar in die eerste reël van die klas en die vorm in.
- Stoor jou program.
- Druk die kode van beide die klas en die vorm indien dit vereis word.

TOTAAL AFDELING B: 60

AFDELING C**VRAAG 3: PROBLEEMOPLOSSINGPROGRAMMERING****SCENARIO**

Daar is drie afdelings in die vroulike pasiënte se vleuel by die Goeie Gesondheid Mediese Sentrum. Afdeling 1 is vir noodgeval-pasiënte, Afdeling 2 vir kinders en Afdeling 3 vir volwasse pasiënte. Elke afdeling het vyf sale en elke saal het tien beddens.

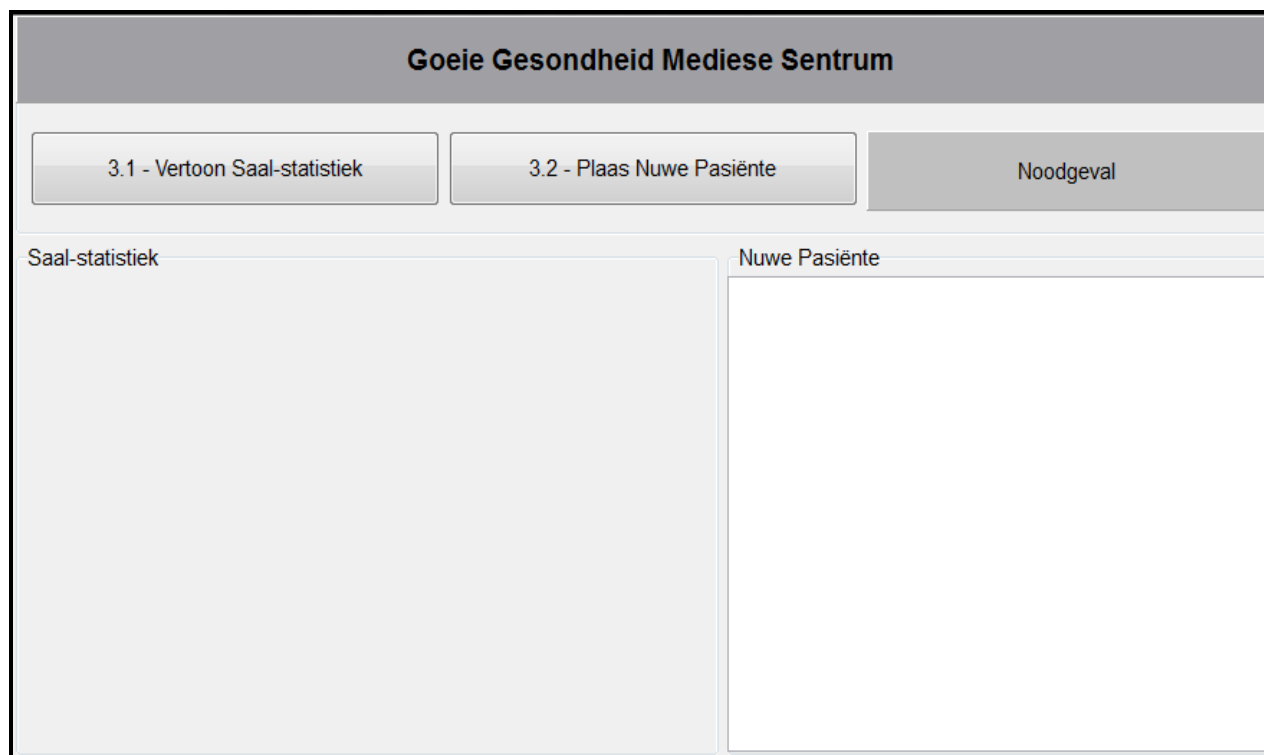
Doen die volgende:

- Kompileer en voer die program in die **Vraag3**-lêergids uit. Die program het tans geen funksionaliteit nie.
- Voltooi die kode vir elke vraag soos wat in VRAAG 3.1 en VRAAG 3.2 hieronder beskryf is.

GGK ('GUI') wat voorsien is:

Die GGK ('GUI') wat voorsien is, bevat komponente vir gebeurtenisse wat moet plaasvind.

'n Voorbeeld van die GGK ('GUI') word hieronder gegee.



Gebruik die programvereistes in die vrae wat volg om op 'n geskikte afvoercomponent te besluit wat in die afvoerarea op die GGK wat **Saal-statistiek** genoem is, geplaas moet word.

Data wat verskaf is:**LET WEL:**

- Jy mag nie data wat verskaf is, met die hand verander nie. Kode moet geskryf word om die data wat verskaf is, volgens die vereistes te manipuleer.
- Die gebruik van goeie programmeringstechnieke en modulêre ontwerp moet in die ontwerp en kodering van jou oplossing toegepas word.

Jy word voorsien van 'n tweedimensionele skikking met die naam **arrSaalStats** wat die getal beddens bevat wat tans in elk van die vyf sale deur pasiënte beset word.

Die waardes wat in die **arrSaalStats**-skikking voorkom, is die volgende:

LET WEL: Die byskrifte is nie deel van die inhoud van die skikking nie.

	Saal 1	Saal 2	Saal 3	Saal 4	Saal 5
Noodgeval	10	9	10	8	9
Kinders	9	10	10	9	10
Volwassenes	9	3	8	2	1

3.1 Knoppie [3.1 – Vertoon Saal-statistiek]

Die program moet die verskillende afdelings, saalnommers en die getal pasiënte in elke saal vertoon. Die inligting moet netjies in kolomme met geskikte opskrifte en byskrifte vertoon word.

Voorbeeld van afvoer:

Saal-statistiek					
	Saal 1	Saal 2	Saal 3	Saal 4	Saal 5
Noodgeval	10	9	10	8	9
Kinders	9	10	10	9	10
Volwassenes	9	3	8	2	1

(8)

3.2 Knoppie [3.2 – Plaas Nuwe Pasiënte]

'n Tekslêer met die naam **DataV3.txt** bevat inligting oor nuwe pasiënte wat opgeneem moet word. Elke teksreël in die lêer bevat die volle naam en die geboortedatum (JJJJ-MM-DD) van die pasiënt, geskei deur 'n hutskarakter. As die pasiënt in die noodgeval-saal opgeneem moet word, is 'n kommapunt, gevolg deur die karakter 'N' in die teksreël, bygevoeg.

Voorbeeld van die eerste drie teksreëls in die lêer:

Debra Jenkins#2005-10-12
 Geraldine Mathews#1992-01-09
 Nosipho Mbele#2010-09-23;N

Skryf kode om die volgende te doen:

Toets of die tekslêer **DataV3.txt** bestaan.

As die tekslêer nie bestaan nie, vertoon 'n geskikte boodskap en maak die program toe.

As die tekslêer bestaan, doen die volgende vir elke nuwe pasiënt:

- Bepaal die afdeling: Noodgeval, Kinders of Volwassenes. 'n Pasiënt word as 'n kind beskou as hy/sy jonger as agttien jaar oud is.
- Vind 'n saal met 'n beskikbare bed in die afdeling. Die soekproses moet by Saal 1 begin. As daar geen beskikbare beddens in Saal 1 is nie, moet Saal 2 nagegaan word, en so aan. Elke saal het 'n minimum van tien beddens.
- Stel 'n plasingkode saam wat die afdeling en die nommer van die saal waar die pasiënt geplaas sal word, bevat.

Voorbeeld:

V3 beteken dat die pasiënt in die afdeling vir volwassenes in saal 3 geplaas sal word.

As al die sale in die spesifieke afdeling vol is, moet die plasingkode vir die pasiënt die woord 'Waglys' wees.

As 'n noodgeval-pasiënt op 'n waglys geplaas word, moet die kleur van die 'Noodgeval'-paneel na rooi verander word.

- Vertoon die naam, ouderdom en plasingkode in die afvoerarea vir nuwe pasiënte.
- As 'n pasiënt in 'n saal geplaas word, moet die waardes in die **arrSaalStats**-skikking ooreenkomstig opgedateer word.

Vertoon die opgedateerde **arrSaalStats**-skikking nadat al die pasiënte in die tekslêer verwerk is.

Voorbeeld van afvoerarea vir **Nuwe Pasiënte** (op die laaste bladsy):

Nuwe Pasiënte		
Naam van pasiënt	Ouderdom	Saal
Debra Jenkins	11	C1
Geraldine Mathews	24	A1
Nosipho Mbele	6	E2
Samantha Myers	9	C4
Brenda Hofmeyer	32	A2
Annamarie James	15	Waglys
Elizabeth Morake	17	E4
Lee-Ann Johnson	14	E4
Ashley Brown	8	E5
Diane Malinga	16	Waglys

Voorbeeld van afvoerarea vir **Saal-statistiek** nadat nuwe pasiënte se besonderhede verwerk is:

Saal-statistiek					
	Saal 1	Saal 2	Saal 3	Saal 4	Saal 5
Noodgeval	10	10	10	10	10
Kinders	10	10	10	10	10
Volwassenes	10	4	8	2	1

PUNTETOEKENING VIR VRAAG 3.2	
Lees en verkry inligting uit die teksleër.	10
Bepaal die afdeling waarin die pasiënt opgeneem gaan word.	8
Bepaal die saal wat in die spesifieke afdeling beskikbaar is.	7
Stel die plasingkode of die woord 'Waglys' saam.	6
Dateer die arrSaalStats -skikking op.	2
Vertoon die naam, ouderdom en plasingkode of woord van die nuwe pasiënt.	3
Vertoon opgedateerde arrSaalStats -skikking.	1
TOTAAL	37

- Sleutel jou eksamennommer as kommentaar in die eerste reël van die programleër in.
- Stoor jou program.
- Maak 'n drukstuk van die kode indien dit vereis word.

TOTAAL AFDELING C: 45
GROOTTOTAAL: 150



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

SENIORSERTIFIKAAT-EKSAMEN

INLIGTINGSTEGNOLOGIE V1

2016

MEMORANDUM

PUNTE: 150

Hierdie vraestel bestaan uit 32 bladsye.

ALGEMENE INLIGTING:

- Hierdie nasienriglyne moet as basis tydens die nasiensessie gebruik word. Dit is geskep om deur die nasieners gebruik te word. Daar word van alle nasieners vereis om 'n standaardiseringsvergadering by te woon om te verseker dat die riglyne konsekwent geïnterpreteer en toegepas word tydens die nasien van leerders se werk.
- Let daarop dat leerders wat alternatiewe korrekte oplossings voorsien as dié wat as voorbeelde van oplossings in hierdie nasienriglyne gegee is, volle krediet vir die relevante antwoord sal ontvang tensy die instruksies in die vraestel nie gevolg is nie of die vereistes van die vraag nie nagekom word nie.
- **Bylae A, B en C** (bladsy 3–8) bevat die nasienrubriek vir elke vraag ongeag watter een van die twee programmeringstale gebruik is.
- **Bylae D, E en F** (bladsy 9–16) bevat voorbeelde van oplossings in programmeringskode vir Java vir VRAAG 1 tot 3.
- **Bylae G, H en I** (bladsy 17–29) bevat voorbeelde van oplossings in programmeringskode vir Delphi vir VRAAG 1 tot 3.
- Kopieë van **Bylae A, B en C** (bladsy 3–8) moet vir elke leerder gemaak word en tydens die nasiensessie voltooi word.

BYLAE A

AFDELING A

VRAAG 1: NASIENRUBRIEK – ALGEMENE PROGRAMMERINGSVAARDIGHEDE

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
	'n Leerder moet slegs een keer gepenaliseer word indien dieselfde fout herhaal word.		
1.1	Stel teks van die byskrif op 'Goeie Gesondheid Mediese Sentrum'✓ Verander die grootte van die skrif na 14pt ✓	2	
1.2	Knoppie – [Vraag 1_2] Stel die koste op 1000✓ Kry getal afhanklikes✓ Kry die indeks wat geselekteer is van die lysblokkie✓ <i>Toets die aarde van die indeks (if or case) ✓</i> <i>Bereken die koste van elke opsie</i> Indeks 1: Stel koste van hooflid op 1.2 van koste Indeks 2: Stel koste van hooflid op 1.4 van koste Indeks 3: Stel koste van hooflid op 1.6 van koste Allokeer 2 punte vir berekening van koste✓ Allokeer een punt vir herhaalde berekening van kos✓ Allokeer 1 punte vir foutboodskap waar geen opsie gekies is nie✓ <i>Formule om die koste van afhanklikes by te voeg</i> Koste = koste + ✓ koste * afhanklikes * 0.5✓ Vertoon die koste korrek✓ Nota: Indien saamgestelde rente berekening gebruik is aanvaar lus	10	

1.3	<p>Knoppie – [Vraag 1_3]</p> <p>Vertoon opskrifte in afvoerarea✓ Kry rekening se waarde van teksblokkie✓</p> <p>Bereken 15% van die eerste paaient✓ Trek die eerste paaient van die rekeningwaarde af✓ Deel die res van die rekening se balans deur 11 om gelyke paaiente te bereken oor die res van die maande✓ Lus van maand 2 tot maand 12✓ Trek die paaient af van die balans✓ Vertoon die paaient se nommer, maandelikse paaient en die balans vir elke maand ✓ Geformateer met geldeenheid en twee desimale plekke✓</p> <p>Alternatiewe oplossing: Lus van maand 1 tot maand 12</p> <p>Vertoon die paaient se nommer, maandelikse paaient en die balans vir elke maand met geldeenheid en twee desimale plekke Trek die paaient af van die balans</p>	10	
-----	---	----	--

1.4.1(a)	<p>Knoppie – [Vraag 1_4]</p> <p>Kry die getal pasiënte✓ en skakel om na heelgetal✓ Doen heelgetaldeling✓ en bepaal die getal pasiënte per dokter✓ Ken die getal pasiënte per dokter aan elke dokter-veranderlike toe✓✓ Doen modulus-bewerking om oorblywende getal pasiënte te bepaal✓ Toets of die res 1✓ is en voeg tel een by dokter1✓ Toets of die res 2 is en voeg 1 by dokter2</p> <p>Vertoon die getal pasiënte per dokter✓</p>	10	
1.4.1(b)	<p>Verskaf alternatiewe oplossing vir as-stellings ✓ Voeg die stelling by om een pasiënt by dokter 1 ook te tel✓ Maak seker dat die tweede if, 2 stellings bevat (begin end/{ })✓</p> <p>//Aanvaar ook enige ander korrekte oplossing</p>	3	
1.4.2	<p>Kry minute per pasiënt✓ Bepaal die begintyd wat geselekteer is ✓stel tyd-veranderlike✓ Gebruik die getal pasiënte van Dokter 1✓ om die totale getal minute te bereken✓ Bepaal die getal ure ✓en minute ✓ wat gewerk is Vertoon die ure en minute ✓ wat gewerk is Pas die ure aan as daar meer as 0 minute oor is✓ Stel Uitteken-tyd se afvoer saam en vertoon✓</p>	10	
TOTAAL AFDELING A:		45	

BYLAE B

AFDELING B

VRAAG 2: NASIENRUBRIEK – OBJEK-GEÖRIENTEERDE PROGRAMMERING

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
2.1.1	<p>Konstruktor: Definisie met vier regte parameters✓ en datatipes✓ Ken parameterwaardes toe aan: kode, dosis, dae en prys ✓✓ Parameterwaardes moet toegeken wees aan korrekte velde</p>	4	
2.1.2	<p>berekenPillePerDag-metode: Definisie van metode✓ Kyk of die eerste karakter van die dosiskode 'n D is✓ Stel getal pille per dag = tweede karakter✓ Skakel om na heelgetal✓ Anders✓ Stel getal pille per dag = tweede karakter * 24 gedeel✓ deur die vierde karakter✓ Stuur die resultaat terug✓ Nota: Eerste as-stelling kan kontroleer vir 'H' Twee As-stellings kan gebruik word i.p.v. if... else Switch/Case-stelling i.p.v. if-stellings</p>	8	

2.1.3	<p>stelsaamGebruikBoodskap-metode: Metode se definisie ✓</p> <p>Toets of die eerste karakter 'n 'D' of 'H' is ✓✓ Daaglik: toets of die vierde karakter 'n 'B' of 'n 'S' is ✓ Kombineer die waarde van die tweede karakter om die korrekte boodskap vir ontbyt of aandete saam te stel ✓✓</p> <p>Uurlik: Kombineer die waarde van die tweede karakter met die waarde van die vierde karakter om die korrekte boodskap saam te stel ✓✓</p> <p>Stel 2 ooreenstemmende boodskappe saam ✓✓</p> <p>Voorbeeld: Kyk of eerste karakter = D Kyk of vierde karakter = B dan boodskap = 'Neem ' + waarde van tweede karakter + 'pil(le) daaglik na ontbyt' anders boodskap = 'Neem ' + waarde van tweede karakter + 'pil(le) daaglik na aandete' anders boodskap = 'Neem ' + waarde van tweede karakter + 'pil(le) + ' elke ' + waarde van vierde karakter + 'uur'</p>	8	
2.1.4	<p>berekenTotaalPille-metode: Metode se definisie ✓ Stuur berekenPillePerDag-metode ✓ * getal dae terug ✓</p>	3	
2.1.5	<p>Accesor-metodes: Regte getKode-metode met die korrekte terugstuur tipe ✓ Regte getPrys-metode met die korrekte terugstuur tipe ✓</p>	2	
2.1.6	<p>stelsaamEtiket-metode: Formaat ✓ en regte attribute ✓ Roep stelsaamGebruikBoodskap-metode ✓ Roep berekenTotaalPille-metode ✓ (-1 vir elke verkeerde attribuut of metode) Regte terugstuur-stelling met korrekte string data tipe ✓</p>	5	

NOTA:

Moenie kandidate penaliseer vir die volgende nie:

As die roepstelling nie ooreenstem met die metode-definisie nie.

2.2.1	<p>Kombinasielys (Combo box)</p> <p>Kry die kode uit die kombinasielys ✓ Kry die indeks van die kode wat geselekteer is ✓ Kry die ooreenstemmende prys uit die prys-sikking ✓ ✓ Toets of ✓ die kode eindige met 'G' dan ✓ vertoon die pnlGeneries-paneel ✓ anders ✓ (of 2de As-stelling i.p.v. anders) steek die pnlGeneries-paneel weg ✓</p> <p>Nota: Die panel-komponent kan ook weggesteek word in die OnChange-event van die ComboBox Die kode vir Vraag 2.2.1 kan ook gekodeer wees as gedeelte van 2.2.2</p>	9	
2.2.2	<p>Knoppie – [2.2.2 – Instansieer Voorskrifitem]</p> <p>Toets of die generiese blokkie geselekteer is ✓ Verminder ✓ die prys met 40% ✓ Instansieer die objek ✓ met die regte getal argumente ✓ in die regte volgorde ✓ en van die regte datatipes ✓</p> <p>Vertoon die boodskap 'Die Voorskrifitem-objek is suksesvol geïnstansieer ✓</p>	8	
2.2.3	<p>Knoppie – [2.2.3 – Vertoon]</p> <p>Roep stelsaamEtiket-metode ✓ En vertoon in die Voorskrifitem se afvoerarea ✓</p> <p><i>Bereken die koste van die voorskrifitem</i> berekenTotaalPille-metode * ✓ getPrys-metode ✓</p> <p>Vertoon die kode en koste van voorskrif ✓ in die Voorskrifrekening afvoerarea ✓</p> <p>Voeg die koste van die voorskrif by die totale rekening ✓</p> <p>Allokeer 1 punt as die TotaleKoste veranderlike 'n globale veranderlike is. ✓</p>	8	
2.2.4	<p>Knoppie – [2.2.4 – Totaal van alle Voorskrifitems]</p> <p>Vertoon die totale koste van al die voorskrifitems in die Voorskrifrekening se afvoerarea ✓ geformateer met 'n geldeenheid en twee desimale plekke ✓</p>	2	

2.2.5	Knoppie – [2.2.5 – Maak alles skoon] Stel die totale koste op 0 ✓ Twee punte word toegeken vir enige 2 van die volgende: ✓✓ Maak die afvoerareas skoon Maak die teksblokkie van die getal dae skoon Verwyder die keuse van die kombinasie lys	3	
	TOTAAL AFDELING B:	60	

BYLAE C

AFDELING C

VRAAG 3: NASIENRUBRIEK – PROBLEEMOPLOSSINGSPROGRAMMERING

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
3.1	<p>Knoppie [3.1 – Vertoon saalstatistiek]</p> <p>Vertoon opskrif met sale se nommers ✓</p> <p>Buitenste lus rye (drie afdelings) ✓ Vertoon afdeling naam ✓ uit die skikking of enige ander struktuur ✓ Binneste lus vir kolomme (vyf sale) ✓ Vertoon die waardes vir onderskeie seksies ✓ in een lyn ✓</p> <p>Regte formaat en uitleg in kolomme ✓✓</p> <p>Tabcount in Delphi String.format in Java</p>	8	
3.2	<p>Knoppie [3.2 – Plaas Nuwe Pasiënte]</p> <p>Lees uit tekslêer en skei inligting (10) <i>{Delphi: AssignFile, Reset</i> <i>Java: Skep objek om uit leer te lees} ✓✓</i> As lêer nie bestaan nie, vertoon boodskap ✓ En maak die program toe ✓ Lus deur die lêer ✓ Lees 'n reël op 'n slag ✓ Kry die naam uit die reël ✓ Kry die jaar/datum uit die reël ✓ Bereken ouderdom as volg: Huidige jaar/datum ✓ – jaar /datum uit die reël ✓ Of enige ander metode om die ouderdom te bereken</p> <p>Bepaal die afdeling waarin die pasiënt opgeneem moet word (8) Toets of dit 'n noodgeval is ✓✓ Stel ry op 1 ✓ Anders ✓ As ouderdom < 18 ✓ Stel ry op 2 ✓ Anders Stel ry op 3 ✓ Bepaal die korrekte simbool vir die afdeling ✓</p>		

	<p>Bepaal die saal in 'n spesifieke afdeling (7)</p> <p>Inisialiseer saal indeks✓ Inisialiseer vlag✓ Gebruik 'n voorwaardelike lus om die indeks teen die hoeveelheid sale te vergelyk en toets vlag✓ As arrSaalStats by die ry, kolom < 10✓ Stel Boolese vlag op waar✓ Stel saalnommer op kolom✓ Vermeerder indeks met 1✓</p> <p>Of enige ander korrekte oplossing</p> <p>Bepaal die plasingkode (6)</p> <p>As die saalnommer = 0 dan✓ Stel kode op 'Waglys'✓ As afdeling = 'N' dan✓ Stel kleur van noodpaneel op rooi✓ Anders✓ Stel kode op afdeling + saalnommer✓</p> <p>Dateer arrSaalStats-skikking op (2)</p> <p>Vermeerder ✓ die waarde van die element wat deur die respektiewelike ry en kolom in die arrSaalStats-skikking aangedui word✓</p> <p>Vertoon die naam, ouderdom en kode van nuwe pasiënt (3)</p> <p>Vertoon naam✓, ouderdom✓, kode✓</p> <p>Vertoon opgedateerde arrSaalstats-skikking (1)</p> <p>Roep die metode wat in 3.1 gebruik is of herhaal die vertoonkode✓</p>		
	TOTAAL AFDELING C:	45	

OPSOMMING VAN LEERDERPUNTE:

SENTRUMNOMMER:		EKSAMENNOMMER:		
	AFDELING A	AFDELING B	AFDELING C	
	VRAAG 1	VRAAG 2	VRAAG 3	GROOTTOTAAL
MAKS.PUNTE	45	60	45	150
LEERDERPUNTE				

BYLAE D: OPLOSSING VIR VRAAG 1: JAVA

```
// *****
// Vraag 1.1
// *****
public Vraag1_Memo() {
    initComponents();
    setLocationRelativeTo(this);
    lblVraag1_1.setFont(new java.awt.Font("Arial", 0, 14));
    lblVraag1_1.setText("Goeie Gesondheid Mediese Sentrum");
}

// *****
// Vraag 1.2
// *****
private void btnVraag1_2ActionPerformed(java.awt.event.ActionEvent evt)
{
    double koste = 1000;
    int aantAfhanklikes =
        Integer.parseInt(txfAantAfhanklikes.getText());
    int medOpsie = lstMedieseSkemaOpsies.getSelectedIndex();
    switch (medOpsie) {
        case 1:
            koste = koste * 1.2;
            break;
        case 2:
            koste = koste * 1.4;
            break;
        case 3:
            koste = koste * 1.6;
            break;
    }
    koste = koste + koste * 0.5 * aantAfhanklikes;
    JOptionPane.showMessageDialog(rootPane, "Die maandelikse paaient
        sal " + String.format("R %.2f", koste) + " wees.");
}
// *****
// Vraag 1.3
// *****
private void btnVraag1_3ActionPerformed(java.awt.event.ActionEvent evt)
{
    txaAfvoer.setText(String.format("%-10s%-25s%-15s%n", "Paaient",
        "Maandelikse Paaient", "Balans"));
    double uitstaandeBedrag =
        Double.parseDouble(txfRekeningBalans.getText());
    double bedragMaand1 = uitstaandeBedrag * 0.15;
    uitstaandeBedrag = uitstaandeBedrag - bedragMaand1;
    txaAfvoer.append(String.format("%-10sR%-24.2fR%.2f%n", 1,
        bedragMaand1, uitstaandeBedrag));
    double bedragPerMaand = uitstaandeBedrag / 11.0;
    for (int k = 2; k <= 12; k++) {
        uitstaandeBedrag = uitstaandeBedrag - bedragPerMaand;
        txaAfvoer.append(String.format("%-10sR%-24.2fR%-13.2f%n", k,
            bedragPerMaand, Math.abs(uitstaandeBedrag)));
    }
}
}
```

```
*****
// Vraag 1.4
// *****
private void btnVraag14ActionPerformed(java.awt.event.ActionEvent evt) {

    // Vraag 1.4.1(a & b)
    int aantal = Integer.parseInt(txfAantal.getText());

    int aantPasiantePerDr = aantal / 3;
    int res = aantal % 3;

    int dokter1 = aantPasiantePerDr;
    int dokter2 = aantPasiantePerDr;
    int dokter3 = aantPasiantePerDr;
    switch (res) {

        case 1:
            dokter1++;
            break;
        case 2: //Stelling wat reggemaak is (b)
            dokter1++;
            dokter2++;
    }
    txfDokter1.setText(dokter1 + "");
    txfDokter2.setText(dokter2 + "");
    txfDokter3.setText(dokter3 + "");

    // Vraag 1.4.2
    int begin = 0;
    int minutePerPasiante = Integer.parseInt(txfMinute.getText());
    if (rbt8.isSelected()) begin = 8;
    if (rbt11.isSelected()) begin = 11;
    if (rbt13.isSelected()) begin = 13;

    int totaalMinute = dokter1 * minutePerPasiante;
    int ure = totaalMinute / 60;
    int minute = totaalMinute % 60;
    String timeBusy = ure + " uur " + minute + " minute";
    if (minute > 0)
        ure = ure + 1;
    int tekenUit = begin + ure;
    txfTyd.setText(timeBusy);
    txfUitteken.setText(tekenUit + "h00");

}
}
```

BYLAE E: OPLOSSING VIR VRAAG 2: JAVA**OPLOSSING VIR VRAAG 2:OBJEKKLAS**

```
package Vraag2Package;

public class VoorskrifItem {

    private String kode;
    private String dosis;
    private int dae;
    private double prys;

    // *****
    // Vraag 2.1.1
    // *****
    public VoorskrifItem(String kode,String dosis,int dae,double prys) {
        this.kode = kode;
        this.dosis = dosis;
        this.dae = dae;
        this.prys = prys;
    }
    // *****
    // Vraag 2.1.2
    // *****
    public int berekenPillePerDag() {
        int aantPillePerDag = 0;

        if (dosis.charAt(0) == ('D')) {
            aantPillePerDag = Integer.parseInt(dosis.substring(1, 2));
        }
        if (dosis.charAt(0) == ('H')) {
            String[] temp = dosis.split("#");
            aantPillePerDag = Integer.parseInt(dosis.substring(1, 2)) *
                24 / Integer.parseInt(temp[1]);
        }
        return aantPillePerDag;
    }

    // *****
    // Vraag 2.1.3
    // *****
    public String stelsaamGebruikBoodskap() {
        String boodskap = "";
        if(dosis.charAt(0)=='D')
        {
            if (kode.charAt(3) == 'B') {
                boodskap = "Neem "+ dosis.charAt(1)+" pil(le) daaglik na
                    ontbyt";
            }
            else
            {

```

```
        boodskap = "Neem "+ dosis.charAt(1)+" pil(le) daagliks na
                    aandete";
    }
    }
else
{
    boodskap = "Neem " + dosis.charAt(1) + " pil(le) elke " +
        dosis.charAt(3) + " uur";
}
return boodskap;
}

// *****
// Vraag 2.1.4
// *****
    public int berekenTotaalPille() {
        return berekenPillePerDag() * dae;
    }
// *****
// Vraag 2.1.5
// *****
    public String getKode() {
        return kode;
    }

    public double getPrys() {
        return prys;
    }
// *****
// Vraag 2.1.6
// *****
    public String stelsaamEtiket() {
        return "Pile se kode: " + kode + "\n\n" +
            stelsaamGebruikBoodskap() + " vir " + dae + " dae.\n\n("
            + berekenTotaalPille() + " pille)";
    }
}
```

GGK-KLAS: VRAAG2_OPLOSSING

```
// *****  
// Vraag 2.2.1  
// *****
```

Kode vir die kombinasie lys

```
private void cmbPilKodesActionPerformed(java.awt.event.ActionEvent evt)  
{  
    itemKode = "" + cmbPilKodes.getSelectedItem();  
    itemPos = cmbPilKodes.getSelectedIndex();  
    itemPrys = arrPryse[itemPos];  
    if(itemKode.endsWith("G"))  
    {  
        pnlGeneric.setVisible(true);  
    }  
    else  
    {  
        pnlGeneric.setVisible(false);  
    }  
}
```

```
// *****  
// Vraag 2.2.2  
// *****
```

```
private void btn222ActionPerformed(java.awt.event.ActionEvent evt) {  
    String dosisKode = "" + cmbDosisKodes.getSelectedItem();  
    int dae = Integer.parseInt(txfAantalDae.getText());  
    if(chbGeneries.isSelected())  
    {  
        itemPrys = itemPrys * 0.6;  
    }  
    objItem = new VoorskrifItem(itemKode, dosisKode, dae, itemPrys);  
    JOptionPane.showMessageDialog(rootPane, "Die VoorskrifItem-objek  
        is suksesvol geinstansieer.");  
}
```

```
// *****  
// Vraag 2.2.3  
// *****
```

```
private void btn223ActionPerformed(java.awt.event.ActionEvent evt) {  
    txaVoorskrif.setText(objItem.stelsaamEtiket());  
    double pilWaarde =  
        objItem.berekenTotaalPille()*objItem.getPrys();  
        txaVoorskrifRekening.append(String.format("%-  
            15sR%.2f%n",objItem.getKode(),pilWaarde));  
    totaleKoste = totaleKoste + pilWaarde;  
}
```

```
// *****  
// Vraag 2.2.4  
// *****  
private void btn224ActionPerformed(java.awt.event.ActionEvent evt) {  
    txaVoorskrifRekening.append(String.format("%-15sR%.2f",  
        "Totaal", totaleKoste));  
}  
  
// *****  
// Vraag 2.2.5  
// *****  
private void btn225ActionPerformed(java.awt.event.ActionEvent evt) {  
    totaleKoste = 0;  
    txfAantalDae.setText("");  
    txaVoorskrifRekening.setText("");  
    txaVoorskrif.setText("");  
}
```

BYLAE F: OPLOSSING VIR VRAAG 3: JAVA

```
// *****
// Data wat voorsien is
// *****

int[][] arrSaalStats = {{10, 9, 10, 8, 9}, {9, 10, 10, 9, 10},
                       {9, 3, 8, 2, 1}};

// Nie aan leerders voorsien nie
String[] arrAfdelings = {"Noodgeval", "Kinders", "Volwassenes"};

// *****
// Vraag 3.1
// *****
private void btnVraag31ActionPerformed(java.awt.event.ActionEvent evt) {
    txtaStatistiek.setText(String.format("%-12s", ""));
    for (int k = 1; k <= 5; k++) {
        txtaStatistiek.append(String.format("%-10s", "Saal " + k));
    }
    for (int ry = 0; ry < arrSaalStats.length; ry++) {
        txtaStatistiek.append(String.format("%n%-12s", arrAfdelings[ry]));
        for (int kol = 0; kol < arrSaalStats[0].length; kol++) {
            txtaStatistiek.append(String.format("%-10s",
                                                arrSaalStats[ry][kol]));
        }
    }
}

// *****
// Vraag 3.2
// *****
private void btnVraag32ActionPerformed(java.awt.event.ActionEvent evt) {
    txtaPasiënte.setText(String.format("%-18s%-10s%-10s%n", "Naam van
                                                pasiënt", "Ouderdom", "Saal"));
    Calendar now = Calendar.getInstance();
    int huidigeJaar = now.get(Calendar.YEAR);
    try {
        Scanner scFile = new Scanner(new FileReader("DataV3.txt"));
        while (scFile.hasNext()) {
            String[] temp = scFile.nextLine().split("#");
            int gebJaar = Integer.parseInt(temp[1].substring(0, 4));
            int oud = huidigeJaar - gebJaar;
            txtaPasiënte.append(String.format("%-18s%-10s", temp[0], oud));
            String afdeling = "";
            int ry = 0;
            if (temp[1].contains(";")) {
                afdeling = "N";
            } else
            {
                if (oud < 18) {
                    afdeling = "K";
                    ry = 1;
                } else
                {

```

```
        afdeling = "V";
        ry = 2;
    }
}
boolean gevind = false;
int kol = 0;
while (gevind == false && kol < 5) {
    if (arrSaalStats[ry][kol] < 10) {
        gevind = true;
        afdeling = afdeling + (kol + 1);
        arrSaalStats[ry][kol]++;
    }
    kol++;
}
if (!gevind) {
    afdeling = "Waglys";
    if (temp[1].contains(";"))
        pnlNoodgeval.setBackground(new java.awt.Color(255, 0, 0));
}
txaPasiante.append(String.format("%-10s%n", afdeling));
}
btnVraag31.doClick();
} catch (FileNotFoundException e) {
    JOptionPane.showMessageDialog(rootPane, "Leer bestaan nie");
    System.exit(0);
}
}
```

BYLAE G: OPLOSSING VIR VRAAG 1: DELPHI

```
interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, StdCtrls, ExtCtrls, ComCtrls;

type
  TfrmVraag1 = class(TForm)
    gpbVraag1_1: TGroupBox;
    gpbVraag1_2: TGroupBox;
    btnVraag1_2: TButton;
    lstMedieseSkema: TListBox;
    gpbVraag1_3: TGroupBox;
    btnVraag1_3: TButton;
    edtBalans: TEdit;
    lblRekening: TLabel;
    redvertoon: TRichEdit;
    lblAfhanklik: TLabel;
    edtAfhanklikes: TEdit;
    lblVraag1_1: TLabel;
    lblOpsies: TLabel;
    gpbVraag1_4: TGroupBox;
    lblPasiante: TLabel;
    lblUitteken: TLabel;
    edtAantal: TEdit;
    edtTyd: TEdit;
    lblMinute: TLabel;
    edtMinute: TEdit;
    btnVraag1_4: TButton;
    lblDokOpskrif: TLabel;
    lblPasianteOpskrif: TLabel;
    lblDok1: TLabel;
    lblDok2: TLabel;
    lblDok3: TLabel;
    edtDok1: TEdit;
    edtDok2: TEdit;
    edtDok3: TEdit;
    rgpBegin: TRadioGroup;
    lblTyd: TLabel;
    edtUitteken: TEdit;
    procedure FormActivate(Sender: TObject);
    procedure btnVraag1_2Click(Sender: TObject);
    procedure btnVraag1_3Click(Sender: TObject);
    procedure btnVraag1_4Click(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
  end;
```

```
var
  frmVraag1: TfrmVraag1;
  iLastNum, iNumDoctors, iNumPatientsPerDoc:integer;

implementation

{$R *.dfm}
// *****
// Vraag 1.1
// *****
procedure TfrmVraag1.FormActivate(Sender: TObject);
begin
  redvertoon.Paragraph.TabCount := 3;
  redvertoon.Paragraph.Tab[0] := 70;
  redvertoon.Paragraph.Tab[1] := 120;
  redvertoon.Paragraph.Tab[2] := 350;
  lblVraag1_1.Font.Size := 14;
  lblVraag1_1.Caption := 'Goeie Gesondheid Mediese Sentrum';
  CurrencyString := 'R';
end;

// *****
// Vraag 1.2
// *****
procedure TfrmVraag1.btnVraag1_2Click(Sender: TObject);
var
  iAantAfhanklik, iNomOpsie: Integer;
  rKoste: Real;
begin
  rKoste := 1000;
  iAantAfhanklik := StrToInt(edtAfhanklikes.Text);
  iNomOpsie := lstMedieseSkema.itemIndex;
  case iNomOpsie of
    1:
      rKoste := rKoste * 1.20;
    2:
      rKoste := rKoste * 1.40;
    3:
      rKoste := rKoste * 1.60;
  end;
  rKoste := rKoste + rKoste * iAantAfhanklik * 0.5;
  showMessage('Die maandelikse paaieiment sal ' + FloatToStrF(rKoste,
    ffCurrency, 6, 2) + ' wees.');
```

```

redVertoon.Lines.Add('Paaieiment' + #9 + 'Maandelikse Paaieiment' + #9 +
'Balans');
rBalans := StrToFloat(edtBalans.Text);
rMaand1 := rBalans * 0.15;
rBalans := rBalans - rMaand1;
redVertoon.Lines.Add('1' + #9 + FloatToStrF(rMaand1, ffCurrency, 10,
2) + #9#9#9#9 + FloatToStrF(rBalans, ffCurrency, 10, 2));

rMaand2 := rBalans / 11;
for I := 2 to 12 do
begin
rBalans := rBalans - rMaand2;
redVertoon.Lines.Add(IntToStr(I) + #9 + FloatToStrF(rMaand2,
ffCurrency, 10,
2) + #9#9#9#9 + FloatToStrF(rBalans, ffCurrency, 10, 2));
end;
end;

// *****
// Vraag 1.4
// *****
procedure TfrmVraag1.btnVraag1_4Click(Sender: TObject);
var
iAantal, iPasientePerDok , iRes : Integer;
iDokter1, iDokter2, iDokter3 : Integer;
iMinutePerPasiënt, iTotaalMinute, iUre, iMinute, iBegin, iUitteken :
Integer;
sTydBesig : String;
begin
// Vraag 1.4.1 (a & b)
iAantal := StrToInt(edtAantal.Text);
iPasientePerDok := iAantal div 3;
iDokter1 := iPasientePerDok ;
iDokter2 := iPasientePerDok ;
iDokter3 := iPasientePerDok ;
iRes := iAantal mod 3;
if iRes = 1 then
iDokter1 := iDokter1 + 1;
if iRes = 2 then // Stelling wat reggemaak is
begin
iDokter1 := iDokter1 + 1;
iDokter2 := iDokter2 + 1;
end;

edtDok1.Text := IntToStr(iDokter1);
edtDok2.Text := IntToStr(iDokter2);
edtDok3.Text := IntToStr(iDokter3);

// Vraag 1.4.2
iMinutePerPasiënt := StrToInt(edtMinute.text);
case rgpBegin.ItemIndex of
0 : iBegin := 8;
1 : iBegin := 11;
2 : iBegin := 13;
end;

```

```
iTotaalMinute := iDokter1 * iMinutePerPasiënt;
iUre := iTotaalMinute div 60;
iMinute := iTotaalMinute mod 60;
sTydBesig := IntToStr(iUre) + ' uur ' + IntToStr(iMinute) + '
                minute';
if (iMinute > 0) then
    iUre := iUre + 1;
iUitteken := iBegin + iUre;
edtTyd.Text := sTydBesig;
edtUitteken.Text := IntToStr(iUitteken) + 'h00';
end;
end.
```

BYLAE H: OPLOSSING VIR VRAAG 2: DELPHI**OBJEKKLAS:**

```

unit VoorskrifItem;
interface
uses SysUtils, Math, Messages, Dialogs;

type
  TVoorskrifItem = class(TObject)
  private
    fKode : string;
    fDosis: string;
    fDae  : integer;
    fPrys : real;
  public
    { public declarations }
    constructor Create(sKode:string; sDosis:string;iDae:integer;
                      :real);

    function berekenPillePerDag: integer;
    function stelsaamGebruikBoodskap: string;
    function berekenTotaalPille: integer;
    function stelsaamEtiket:string;
    function getKode:string;
    function getPrys:real;
  end;

implementation

{ TVoorskrifItem }
// *****
// Vraag 2.1.1
// *****
constructor TVoorskrifItem.Create(sKode:string;
sDosis:string;iDae:integer; rPrys:real);
begin
  fKode := sKode;
  fDosis := sDosis;
  fDae := iDae;
  fPrys := rPrys;
end;

// *****
// Vraag 2.1.2
// *****
function TVoorskrifItem.berekenPillePerDag:integer;
Var
  iAantPillePerDag: integer;
begin
  if fDosis[1] = 'D' then
    iAantPillePerDag := StrToInt(fDosis[2])
  else
    iAantPillePerDag := StrToInt(fDosis[2]) * 24 div
      StrToInt(fDosis[4]);
  Result := iAantPillePerDag;
end;

```

```

// *****
// Vraag 2.1.3
// *****
function TVoorskrifItem.stelsaamGebruikBoodskap:string;
Var
  sBoodskap: string;
begin
  if fDosis[1] = 'D' then
    begin
      if fDosis[4] = 'B' then
        sBoodskap := 'Neem '+ fDosis[2]+' pil(le) daaglik na ontbyt'
      else
        sBoodskap := 'Neem '+ fDosis[2]+' pil(le) daaglik na aandete'
      end
    else
      sBoodskap := 'Neem '+fDosis[2]+' pil(le) elke ' + fDosis[4] + ' uur';
    Result := sBoodskap;
  end;

// *****
// Vraag 2.1.4
// *****
function TVoorskrifItem.berekenTotaalPille: integer;
begin
  Result := berekenPillePerDag * fDae;
end;

// *****
// Vraag 2.1.5
// *****
function tVoorskrifItem.getKode:string;
begin
  Result := fKode;
end;

function TVoorskrifItem.getPrys:real;
begin
  Result := fPrys;
end;

// *****
// Vraag 2.1.6
// *****
function TVoorskrifItem.stelsaamEtiket:string;
begin
  result := 'Pil se kode: ' + fKode + #13#13 + stelsaamGebruikBoodskap
+ ' vir ' + IntToStr(fDae) + ' dae.'+#13#13+ '('+
IntToStr(berekenTotaalPille)+ ' pille)';
end;

end.

```

HOOFVORMEENHEID: VRAAG2_U.PAS

```

unit Vraag2_U;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, VoorskrifItem, StdCtrls, ExtCtrls, ComCtrls;

type
  TfrmVraag2 = class(TForm)
    pnlGUI: TPanel;
    grpPilKode: TGroupBox;
    cmbKode: TComboBox;
    grpDosisDae: TGroupBox;
    edtAantDae: TEdit;
    grpVoorskrif: TGroupBox;
    redItem: TRichEdit;
    grpRekening: TGroupBox;
    redRekening: TRichEdit;
    btn222: TButton;
    btn223: TButton;
    btn224: TButton;
    pnlGeneries: TPanel;
    lblGeneries: TLabel;
    btn225: TButton;
    pnlOpskrif: TPanel;
    chkGeneries: TCheckBox;
    cmbDosis: TComboBox;
    lblDosis: TLabel;
    lblDae: TLabel;
    procedure FormActivate(Sender: TObject);
    procedure cmbKodeChange(Sender: TObject);
    procedure btn222Click(Sender: TObject);
    procedure btn223Click(Sender: TObject);
    procedure btn224Click(Sender: TObject);
    procedure btn225Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag2: TfrmVraag2;
  arrKodes : array[1..10] of string =
('CBLTAD', 'HPCMCN', 'TRVKMG', 'BGJKPT', 'KJUJTG', 'NJUKHT', 'PLOMNG', 'NMKBGF',
, 'AMNJKG', 'BGHYTR');
  arrpryse : array[1..10] of real = (17.89, 14.60, 23.50, 22.75, 21.99,
32.50, 79.87, 12.90, 16.79, 23.49);
implementation

{$R *.dfm}
var

```

```

objVoorskrifItem : TVoorskrifItem;
sKode      : string;
rPrys     : real;
iIndeks  : integer;
iTotaal  : real;

// *****
// Vraag 2.2.1
// *****
procedure TfrmVraag2.cmbKodeChange(Sender: TObject);
begin
    sKode := cmbKode.Items[cmbKode.ItemIndex];
    iIndeks := cmbKode.ItemIndex;
    rPrys := arrpryse[iIndeks+1];
    if sKode[length(sKode)] = 'G' then
        pnlGeneries.Show
    else
        pnlGeneries.Hide;
    redItem.Lines.Clear;
end;

// *****
// Vraag 2.2.2
// *****
procedure TfrmVraag2.btn222Click(Sender: TObject);
begin
    if chkGeneries.Checked then
        rPrys := rPrys * 0.6;

    objVoorskrifItem :=
TVoorskrifItem.Create(sKode, cmbDosis.Items[cmbDosis.ItemIndex], StrToInt(
edtAantDae.Text), rPrys);
    ShowMessage('Die VoorskrifItem-objek is suksesvol geinstansieer')
end;

// *****
// Vraag 2.2.3
// *****
procedure TfrmVraag2.btn223Click(Sender: TObject);
Var
    rTabletWaarde :real;
begin
    redItem.Lines.Clear;
    redItem.Lines.Add(objVoorskrifItem.stelsaamEtiket);
    rTabletWaarde := objVoorskrifItem.berekenTotaalPille *
objVoorskrifItem.getPrys;
    redRekening.Lines.Add(objVoorskrifItem.getKode+#9+
FloatToStrF(rTabletWaarde, ffCurrency, 5, 2));
    iTotaal := iTotaal + rTabletWaarde;
end;

// *****
// Vraag 2.2.4
// *****
procedure TfrmVraag2.btn224Click(Sender: TObject);
begin
    redRekening.Lines.Add('Totaal:  '+#9+
FloatToStrF(iTotaal, ffCurrency, 5, 2));
end;

```

```
// *****  
// Vraag 2.2.5  
// *****  
procedure TfrmVraag2.btn225Click(Sender: TObject);  
begin  
    redItem.Lines.Clear;  
    redRekening.Lines.Clear;  
    iTotaal := 0;  
    edtAantDae.Text := '';  
end;  
  
procedure TfrmVraag2.FormActivate(Sender: TObject);  
begin  
    pnlGeneries.Hide;  
    redRekening.Paragraph.TabCount := 2;  
    redRekening.Paragraph.Tab[0] := 60;  
    redRekening.Paragraph.Tab[1] := 100;  
  
end;  
  
end.
```

BYLAE I: OPLOSSING VIR VRAAG 3: DELPHI

```

unit Vraag3_U;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls, ComCtrls, ExtCtrls;

type
  TfrmVraag3 = class(TForm)
    gpbButtons: TGroupBox;
    grpSaalStats: TGroupBox;
    btn32: TButton;
    btn31: TButton;
    pnlNoodgeval: TPanel;
    gpbNuwe: TGroupBox;
    redNuwePasiante: TRichEdit;
    pnlOpskrif: TPanel;
    redSaalStats: TRichEdit;
    procedure Vertoon;
    function BepaalSaal(iRy: integer): integer;
    procedure btn31Click(Sender: TObject);
    procedure btn32Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmVraag3: TfrmVraag3;
  // Data wat voorsien is

  arrSaalStats: array [1 .. 3, 1 .. 5] of integer = ((10,9,10,8,9),
    (9,10,10,9,10), (9,3,8,2,1));

  // Nie aan die leerders voorsien nie

  arrAfdelings: array [1 .. 3] of string = (
    'Noodgeval',
    'Kinders',
    'Volwassenes' );

implementation
uses DateUtils;
{$R *.dfm}

// *****
// Vraag 3.1
// *****
procedure TfrmVraag3.btn31Click(Sender: TObject);
begin
  Vertoon(); // roep vertoon-metode (modulêre programmering)
end;

```

```
// *****  
// Vraag 3.2  
// *****  
procedure TfrmVraag3.btn32Click(Sender: TObject);  
var  
    Teksleer: Textfile;  
    sReel, sNaam, sJaar, sAfdeling, sKode: string;  
    iJaar, iOud, iRy, iKol: integer;  
begin  
    redNuwePasiënte.Paragraph.TabCount := 3;  
    redNuwePasiënte.Paragraph.Tab[0] := 60;  
    redNuwePasiënte.Paragraph.Tab[1] := 120;  
    redNuwePasiënte.Paragraph.Tab[2] := 200;  
    if not FileExists('DataV3.txt') then  
        begin  
            MessageDlg('Leer bestaan nie.', mtError, [mbOk], 0);  
            Application.terminate; //close  
        end;  
  
    AssignFile(Teksleer, 'DataV3.txt');  
    Reset(Teksleer);  
  
    redNuwePasiënte.Clear;  
    redNuwePasiënte.Lines.Add('Naam van pasiënt' + #9 + 'Ouderdom' + #9 +  
        'Saal' + #13);  
  
    while NOT EOF(Teksleer) do  
        begin  
            readln(Teksleer, sReel);  
            sNaam := copy(sReel, 1, pos('#', sReel) - 1);  
            Delete(sReel, 1, pos('#', sReel));  
            sJaar := copy(sReel, 1, 4);  
            iOud := YearOf(Today()) - StrToInt(sJaar);  
            if (pos(';', sReel) > 0) then  
                begin  
                    sAfdeling := 'N';  
                    iRy := 1;  
                end  
            else  
                begin  
                    if (iOud < 18) then  
                        begin  
                            sAfdeling := 'K';  
                            iRy := 2;  
                        end  
                    else  
                        begin  
                            sAfdeling := 'V';  
                            iRy := 3;  
                        end;  
                end;  
        end;  
  
    if (BepaalSaal(iRy) = 0) then // roep BepaalSaal-metode  
        begin // (modulêre programmeing)  
            if iRy = 1 then  
                pnlNoodgeval.Color := clRed;
```

```

        sKode := 'Waglys';
    end
    else
    begin
        iKol := BepaalSaal(iRy);
        sKode := sAfdeling + IntToStr(iKol);
        Inc(arrSaalStats[iRy][iKol]);
    end;
    redNuwePasiante.Lines.Add(sNaam + #9 + IntToStr(iOud) + #9#9 + sKode
+ #13);
    Vertoon;
end; // while
CloseFile(Teksleer);
end;

```

```

// *****
// Vraag 3.1 (vervolg)
// *****
procedure TfrmVraag3.Vertoon;

```

```

var
    iC, iSaal, iKategorie: integer;
    sReel: string;

begin
    redSaalStats.Clear;
    redSaalStats.Paragraph.TabCount := 5;
    redSaalStats.Paragraph.Tab[0] := 10;
    redSaalStats.Paragraph.Tab[1] := 100;
    redSaalStats.Paragraph.Tab[2] := 150;
    redSaalStats.Paragraph.Tab[3] := 200;
    redSaalStats.Paragraph.Tab[4] := 250;
    sReel := #9 + #9;
    for iC := 1 to 5 do
        sReel := sReel + 'Saal ' + IntToStr(iC) + #9;
        redSaalStats.Lines.Add(sReel + #10);
        for iKategorie := 1 to 3 do
            begin
                sReel := arrAfdelings[iKategorie];
                for iSaal := 1 to 5 do
                    sReel := sReel + #9 + IntToStr(arrSaalStats[iKategorie, iSaal]);
                    redSaalStats.Lines.Add(sReel);
                end;
            end;
        end;
end;

```

```

// *****
// Vraag 3.2 vervolg
// *****
function TfrmVraag3.BepaalSaal(iRy: integer): integer;

```

```

var
    iKol : integer;
    bFound: boolean;
begin
    iKol := 1;
    bFound := false;
    while ((iKol <= 5) AND (bFound = false)) do
        begin
            if (arrSaalStats[iRy][iKol] < 10) then

```

```
begin
  bFound := true;
  Result := iKol;
end;
iKol := iKol + 1;
end;
if (bFound = false) then
  Result := 0;
end;
end.
```