



**NASIONALE
SENIOR SERTIFIKAAT**

GRAAD 12

SEPTEMBER 2022

INLIGTINGSTEGNOLOGIE V1

PUNTE: 150

TYD: 3 uur

Hierdie vraestel bestaan uit 18 bladsye.

INSTRUKSIES EN INLIGTING

1. Hierdie vraestel is in VIER afdelings ingedeel. Kandidate moet AL die vrae in elk van die VIER 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 spesifiek tot die Delphi-programmeringstaal is.
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 verskaf is of enige data-uittreksels wat in die vraestel verskyn nie.
7. Roetines, soos vind, soek, sorteer en seleksie, moet vanuit eerste beginsels ontwikkel word. Jy mag NIE die ingeboude funksies van Delphi vir enige van hierdie roetines gebruik NIE.
8. Alle datastrukture moet deur jou, die programmeerder, verklaar word, tensy die datastrukture verskaf 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 naam 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 gegee 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 wagwoordbeskermdes uitvoerbare lêers verskaf.

Doen die volgende:

- Dubbelklik op die wagwoordbeskermdes uitvoerbare lêer.
- Klik op die 'Extract'-knoppie.
- Sleutel die volgende wagwoord in: **Oh%t22**

Nadat dit onttrek ('extracted') is, sal die volgende lys lêers in die lêergids ('folder') **DataSept2022** beskikbaar wees:

Vraag 1:

Vraag1_u.pas
Vraag1_u.dfm
Vraag1_p.dpr
Vraag1_p.res

Vraag 2:

dbConnection_u.pas
Aflewering.mdb
AfleweringRugsteun.mdb
Vraag2_u.pas
Vraag2_u.dfm
Vraag2_p.dpr
Vraag2_p.res

Vraag 3:

Vraag3KlasDefinisie.pas
Vraag3_u.pas
Vraag3_u.dfm
Vraag3_p.dpr
Vraag3_p.res

Vraag 4:

stemme.txt
Vraag4_u.pas
Vraag4_u.dfm
Vraag4_p.dpr
Vraag4_p.res

VRAAG 1: ALGEMENE PROGRAMMERINGSVAARDIGHEDE

Doen die volgende:

- Maak die onvoltooide program in die **Vraag 1**-lêergids oop.
- Sleutel jou naam en van as kommentaar in die eerste reël van die **Vraag1_u.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie.
- Volg die instruksies hieronder om die kode vir elke afdeling van VRAAG 1, soos beskryf in VRAAG 1.1, VRAAG 1.2, VRAAG 1.3 en VRAAG 1.4, te voltooi.

1.1 Knoppie [1.1 Vorm]

Skryf kode om die volgende te doen:

- Verander die vorm van die vorm-komponent, **shpToets** na 'n sirkel.
- Verander die kleur van die vorm, **shpToets** na groen.
- Verander die 'group box' **gbxVraag12** sodat dit sigbaar is.
- Maak die 'group box' **gbxVraag13** aktief.

LET WEL: Veranderinge wat in die 'Object Inspector' gemaak word, sal nie gemerk word nie.

(4)

1.2 Knoppie [1.2 Ontsyfer]

Die volgende kode is in die redigeerblokkie, **edtToevoer**, ingetik:

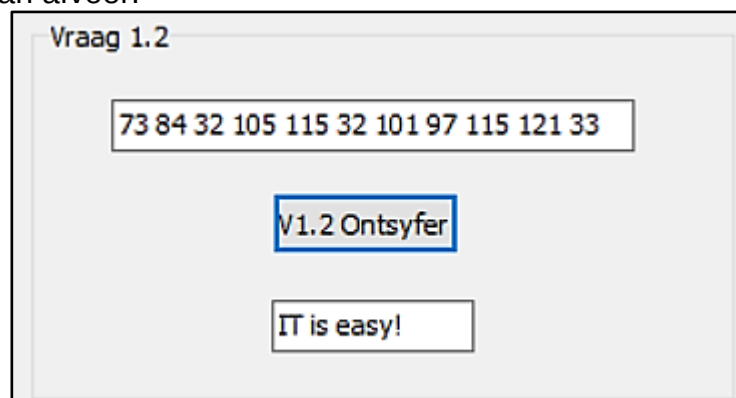
```
73 84 32 105 115 32 101 97 115 121 33
```

Elke getal in die string is die ordinale getal van die karakter of simbool en elke ordinale getal is deur 'n spasie geskei.

Skryf kode om die sin, nadat die ordinale getalle na die ooreenstemmende karakters en simbole verwerk is, te vertoon. Maak gebruik van die CHR-funksie.

LET WEL: 'n Spasie het 'n ordinale waarde van 32.

Voorbeeld van afvoer:



(8)

1.3 Knoppie [1.3 Toets]

'n Priemgetal is 'n positiewe heelgetal wat presies twee positiewe faktore het. Die syfer 1 is nie 'n priemgetal nie.

Die getal 16 kan as die som van twee priemgetalle voorgestel word, byvoorbeeld $3 + 13$ en $5 + 11$. Geen ander priemgetalle wat bymekaar getel word, sal gelyk wees aan 16 nie.

Gebruik die gegewe heelgetal en bepaal die maniere waarop die syfer voorgestel kan word, deur twee priemgetalle bymekaar te tel. Die som, $x + y$ en $y + x$, kan as dieselfde berekening aanvaar word.

Doen die volgende:

- Maak die 'listbox', **IstAfvoer**, skoon.
- Kry die toevoer van die spinredigeerblokkie, **sedGetal**.
- Vind al die priemfaktore van die toevoer en stoor die faktore in 'n skikking.
- Bereken en vertoon, in die 'listbox' **IstAfvoer**, al die kombinasies van twee priemgetalle wat bymekaar getel kan word, wat aan die toevoer gelyk is.
- Vertoon 'n geskikte boodskap as die toevoer nie bereken kan word deur twee priemgetalle bymekaar te voeg nie.

Voorbeeld van afvoer:

(21)

1.4 Knoppie [1.4 Area]

'n Sirkel het 'n omtrek van 0.9 m en 'n vierkant met 'n sy lengte van 0.2 m word uit 'n sirkel gesny.

Area van 'n sirkel = $\pi \times \text{radius} \times \text{radius}$.

Omtrek van 'n sirkel = $2 \times \pi \times \text{radius}$

Radius van 'n sirkel = $\text{omtrek} / (2 \times \pi)$

Gebruik die volgende formules:

Area van 'n sirkel = $\pi \times (0.9 / (2 \times \pi)) \times (0.9 / (2 \times \pi)) \text{ m}^2$

Area van 'n vierkant = $0.2 \times 0.2 \text{ m}^2$

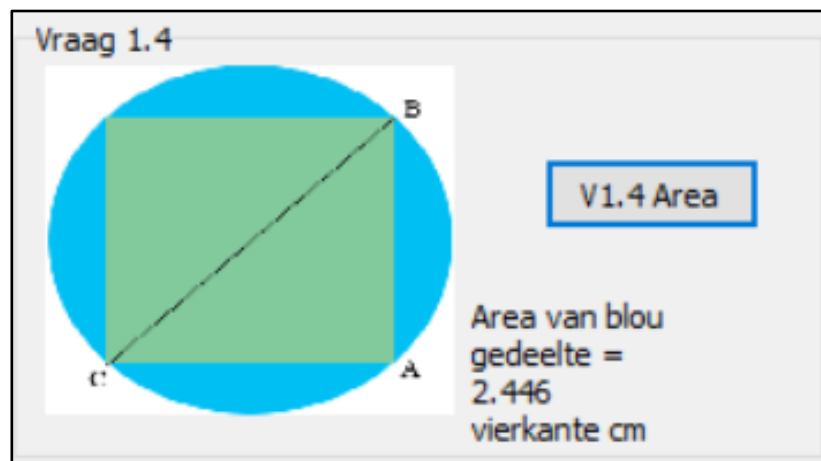
Oorblywende area van die sirkel = area van sirkel – area van vierkant

Skryf kode om die oorblywende area van die sirkel te bereken. Dit moet na cm^2 verwerk word. ($1 \text{ m}^2 = 100 \text{ cm}^2$)

Vertoon die antwoord afgerond na drie desimale plekke in die **lblAfvoer**-komponent.

LET WEL: Voeg die antwoord by die huidige teks wat in die byskrif ('label') vertoon word.

Voorbeeld van afvoer:



(7)

- Sleutel jou naam en van as kommentaar in die eerste reël van die programlêer.
- Stoor jou program.
- 'n Drukstuk van jou kode mag vereis word.

[40]

VRAAG 2: SQL EN DATABASISPROGRAMMERING

Die databasis **Aflwering.mdb** bevat inligting van winkels en bestellings wat deur die kliënte geplaas is en deur 'n aflweringsbesigheid afgelewer word. Die databasis bestaan uit twee tabelle, naamlik **tblWinkels** en **tblBestellings**.

Tabel: **tblWinkels**

Hierdie tabel bevat die beskrywings van winkels wat die aflweringsbesigheid gebruik.

Veldnaam	Datatype	Beskrywing
WinkelID	AutoNumber	'n Unieke getal wat aan elke winkel toegeken is
WinkelNaam	Text (50)	Die naam van die winkel
Aanlyn	Yes/No	Dui aan of die winkel goedere aanlyn verkoop of nie

Voorbeeld van data in die **tblWinkels**-tabel:

WinkelID	WinkelNaam	Aanlyn
1	Grocery Checkout	☒
2	Pay Less Purchasing	☐
3	Economy All Foods	☒
4	WellWorthIt Suppliers	☒
5	Faster Fashions	☐
6	Edjoy Retailers	☐
7	Get Stores	☐
8	Span Stores	☒
9	Discover Pharmacy	☐
10	Collect Pharmacy	☒

Tabel: **tblBestellings**

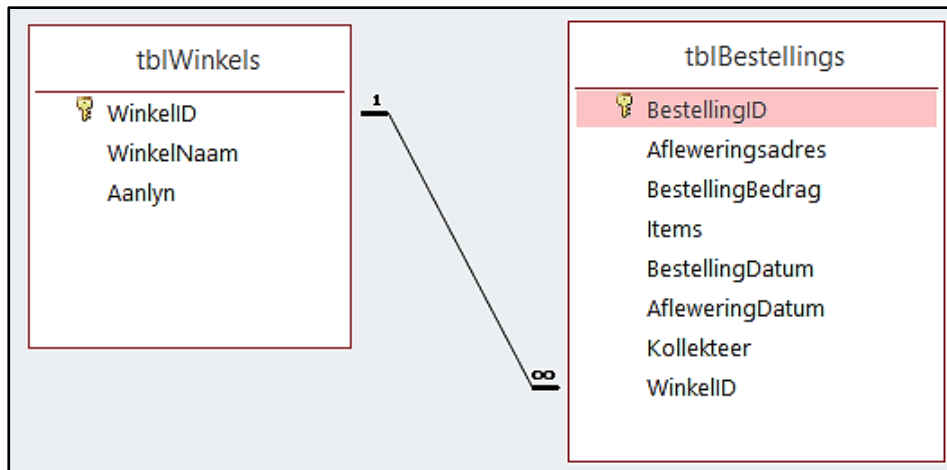
Hierdie tabel bevat inligting van bestellings en aflweringsadresse.

Veldnaam	Datatype	Beskrywing
BestellingID	Autonumber	'n Unieke getal wat aan bestelling toegeken is
Aflweringsadres	Text (50)	Die adres van koper waar die bestelling afgelewer moet word.
BestellingBedrag	Currency	Geldwaarde van bestelling
Items	Number	Aantal items wat bestel is
BestellingDatum	Date (short date)	Datum wanneer bestelling geplaas is
AflweringDatum	Date (short date)	Datum wanneer bestelling afgelewer word
Kollekteer	Yes/No	Dui aan of die kliënt die bestelling wil afhaal in plaas van gebruik maak van aflwering
WinkelID	Number	Vreemde sleutel wat die tblWinkel verbind

Voorbeeld van data van die eerste sewe rekords van die **tblBestellings**-tabel:

BestellingID	Aflweringsadres	BestellingBedrag	Items	BestellingDatum	AflweringDatum	Kollekteer	WinkelID
1	26 Noel Street	R435.54	8	2022/01/24		☒	1
2	6 Busby Avenue	R1 323.98	18	2022/02/12	2022/02/15	☐	5
3	2 Nxolo Street	R304.71	8	2022/02/08	2022/02/10	☐	3
4	46 Ekasi Avenue	R1 230.60	15	2022/08/01	2022/08/02	☐	6
5	18 Hope Street	R672.75	5	2022/04/03	2022/04/07	☐	3
6	12 Bains Broadway	R348.00	7	2022/04/16	2022/04/18	☐	1
7	23 Nxolo Street	R145.00	6	2022/02/08	2022/02/10	☐	2

Die verwantskap tussen die twee tabelle word hieronder gewys:



Doen die volgende:

- Maak die onvoltooide projeklêer, **Vraag2_p.dpr** in die Vraag 2-gidslêer oop.
- Tik jou naam en van as kommentaar in die eerste reël van die **Vraag2_u.pas** lêer in.
- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie.
- Volg die instruksies om die kode vir elke vraag te voltooi, soos in VRAAG 2.1 en VRAAG 2.2 beskryf is.
- **Gebruik SQL-kode om VRAAG 2.1 te beantwoord en Delphi-kode om VRAAG 2.2 te beantwoord.**

LET WEL:

- Die [**Herstel Databasis**]-knoppie word verskaf om die data wat in die databasis voorkom, na die oorspronklike inhoud te herstel. As jy jou kode op die oorspronklike data moet toets, mag jy die knoppie klik om die data te herstel.
- MOENIE gegewe kode verander NIE.
- Kode word gegee om die GJK-komponente met die databasis te verbind.
- **TWEE** veranderlikes is as globale veranderlikes verklaar, soos in die onderstaande tabel beskryf word.
- **Gebruik tblWinkels en tblBestellings komponente net in VRAAG 2.2.**

Veranderlike	Datatype	Beskrywing
tblWinkels	TADOTable	Verwys na die tabel met die naam tblWinkels
tblBestellings	TADOTable	Verwys na die tabel met die naam tblBestellings

2.1 In hierdie afdeling mag jy SLEGS **SQL-stellings** gebruik om VRAAG 2.1.1 tot VRAAG 2.1.6 te beantwoord.

Kode word voorsien om die SQL-stellings uit te voer en die resultate van die navrae te vertoon. Die SQL-stellings is onvolledig.

Doen die volgende om die onvoltooide **SQL-stellings** wat respektiewelik per vraag aan die veranderlikes sSQL1, sSQL2, sSQL3, sSQL4, sSQL5 en sSQL6 toegeken is, te voltooi.

2.1.1 Knoppie [V 2.1.1]

Skryf SQL-kode wat die **Afleweringadres** en **AfleweringDatum** van al die bestellings vertoon, gesorteer met die mees onlangse **AfleweringDatum** eerste.

Voorbeeld van afvoer:

Afleweringadres	AfleweringDatum
24 Strand Crescent	2022/11/20
8 Breslar Crescent	2022/11/20
11 Mtuli Avenue	2022/10/27
42 Cuyler Avenue	2022/10/27
46 Ekasi Avenue	2022/08/02
6 Mtutu Highway	2022/06/09
12 Brent Street	2022/06/09
17 Bwambani Road	2022/05/10
101 Constance Road	2022/05/10
23 Sizemore Road	2022/04/18
12 Bains Broadway	2022/04/18
25 Chase Street	2022/04/18

(3)

2.1.2 Knoppie [V 2.1.2]

Skryf SQL-kode om die aantal bestellings waar die aantal **Items** 20 of meer is, te vertoon. Vertoon die antwoord as **Items20ofMeer**.

Voorbeeld van afvoer:

Items20ofMeer
4

(4)

2.1.3 Knoppie [V 2.1.3]

Die gebruiker moet 'n straatnaam intik. Kode word verskaf vir straatnaam "**Nxolo Street**" om in die toevoerkomponent ingelees te word en in die veranderlike met die naam **sStraat** te stoor.

Skryf SQL-kode om al die velde van die bestellings waar die straatnaam, wat in **sStraat** gestoor is, in die **Afleweringadres** voorkom, te vertoon.

Voorbeeld van afvoer:

BestellingID	Afleweringadres	BestellingBedrag	Items	BestellingDatum	AfleweringDatum	Kollekteer	WinkelID
3	2 Nxolo Street	304.71	8	2022/02/08	2022/02/10	False	3
7	23 Nxolo Street	145	6	2022/02/08	2022/02/10	False	2

(4)

2.1.4 Knoppie [V 2.1.4]

Skryf 'n SQL-stelling om die **Kollekteer**-veld te verander na 'True' vir al die bestellings waar die **AfleweringDatum**-veld leeg is.

(Kode is geskryf om die tabel te vertoon nadat die veranderinge gemaak is.)

Voorbeeld van afvoer van die eerste 7 rekords:

BestellingID	Afleweringadres	BestellingBedr	Items	BestellingDatum	AfleweringDatum	Kollekteer	WinkelID
1	26 Noel Street	435.54	8	2022/01/24		True	1
2	6 Busby Avenue	1323.98	18	2022/02/12	2022/02/15	False	5
3	2 Nxolo Street	304.71	8	2022/02/08	2022/02/10	False	3
4	46 Ekasi Avenue	1230.6	15	2022/08/01	2022/08/02	False	6
5	18 Hope Street	672.75	5	2022/04/03	2022/04/07	False	3
6	12 Bains Broadway	348	7	2022/04/16	2022/04/18	False	1
7	23 Nxolo Street	145	6	2022/02/08	2022/02/10	False	2

(3)

2.1.5 Knoppie [V 2.1.5]

Skryf SQL-kode om die **Afleweringadres** te vertoon asook die aantal dae wat dit geneem het om die bestelling af te lewer in 'n berekende veld met die naam **AfleweringDae**.

LET WEL: Alle bestellings is in 2022 geplaas en die **BestellingDatum** en **AfleweringDatum** vir alle bestellings vind in dieselfde maand plaas.

Voorbeeld van afvoer van die eerste 10 rekords:

Afleweringadres	AfleweringDae
6 Busby Avenue	3
2 Nxolo Street	2
46 Ekasi Avenue	1
18 Hope Street	4
12 Bains Broadway	2
23 Nxolo Street	2
24 Strand Crescent	3
17 Busby Avenue	3
25 Chase Street	2
17 Bwambani Road	3

(4)

2.1.6 Knoppie [V 2.1.6]

Skryf SQL-kode om die **WinkelNaam** en die totale verkope, in geldeenheid, vir elke winkel te vertoon met die opskrif 'TOTALEVerkope'.

Voorbeeld van afvoer:

WinkelNaam	TOTALEVerkope
Collect Pharmacy	R3 530.18
Discover Pharmacy	R1 029.64
Economy All Foods	R1 877.66
Edjoy Retailers	R1 230.60
Faster Fashions	R1 777.85
Get Stores	R3 736.78
Grocery Checkout	R1 884.80
Pay Less Purchasing	R984.65
Span Stores	R1 198.91
WellWorthIt Suppliers	R1 409.32

(7)

TOTAAL VRAAG 2.1: (25)

- 2.2 In hierdie afdeling mag slegs Delphi-programmeringskode gebruik word om VRAAG 2.2.1 en VRAAG 2.2.2 te beantwoord.

Gebruik die globale veranderlikes, **tblWinkels** en **tblBestellings**, wat verskaf is.

GEEN punte sal vir SQL-stellings in VRAAG 2.2 toegeken word NIE.

2.2.1 Knoppie [V2.2.1]

Skryf kode om al die **winkelname** te vertoon van die winkels wat aanlynwebwerwe het. Kode word vir die opskrif verskaf.

Voorbeeld van afvoer:

Winkels wat aanlynverkope toelaat

Grocery Checkout
Economy All Foods
WellWorthIt Suppliers
Span Stores
Collect Pharmacy

(5)

2.2.2 Knoppie [V 2.2.2]

- Kode is geskryf om die **WinkelNaam** van die kombinasieblok, cmbWinkel, te kry en dit in die veranderlike **sWinkel** te stoor.
- Kode is geskryf om al die opskrifte in die 'richedit', **redAfvoer** te vertoon.
- Skryf kode om al die bestellings vir die winkelnaam, wat in die kombinasieblok gekies is, in die **redAfvoer** te vertoon, asook die **BestellingDatum** en **BestellingBedrag** van elke bestelling van die gekose winkel.
- Jy moet die totale bedrag van al die bestellings aan die einde van die lys vertoon.

Voorbeeld van afvoer:

Winkelnaam: Discover Pharmacy

V 2.2.1 V 2.2.2

Datum van Bestelling	Bedrag van Bestelling
2022/11/17	R183.74
2022/04/16	R845.90

Totale bedrag bestel: R1 029.64

Winkelnaam: Get Stores

V 2.2.1 V 2.2.2

Datum van Bestelling	Bedrag van Bestelling
2022/05/07	R563.61
2022/04/03	R774.30
2022/02/08	R2 398.87

Totale bedrag bestel: R3 736.78

Winkelnaam: Grocery Checkout

V 2.2.1 V 2.2.2

Datum van Bestelling	Bedrag van Bestelling
2022/01/24	R435.54
2022/04/16	R348.00
2022/03/14	R180.96
2022/04/16	R285.43
2022/06/08	R634.87

Totale bedrag bestel: R1 884.80

(10)

TOTAAL VRAAG 2.2: (15)

- Sleutel jou naam en van as kommentaar in die eerste reël van die programlêer.
- Stoor jou program.
- 'n Drukstuk van jou kode mag vereis word.

[40]

VRAAG 3: OBJEK-GEÏORIENTEERDE PROGRAMMERING

Die besigheid, TLC Car Repair, het 'n program nodig om rekord te hou van hul kwotasies om motors te herstel. Die koste van arbeid vir herstelwerk is R70 per uur.

Doen die volgende:

- Maak die onvoltooide program in die Vraag 3-lêergids oop.
- Maak die onvoltooide objekklas **Vraag3KlasDefinisie.pas** oop.
- Sleutel jou naam en van as kommentaar in beide die **Vraag3KlasDefinisie.pas** en **Vraag3_u.pas**.
- Kompileer en voer die program uit. Tans het die program geen funksionaliteit nie.
- MOENIE gegewe kode verander of verwyder NIE.

Die volgende gebruikerskoppelvlak word vertoon:

Voltooi die kode vir hierdie program, soos wat dit in VRAAG 3.1 en VRAAG 3.2 gespesifiseer is.

- 3.1 Die onvolledige klas (**THerstel**) bevat vyf attribute wat die **objHerstel**-objek beskryf. Twee metodes, **getUre** en **getTotaleKoste** is vir jou geskryf.

NAME VAN ATTRIBUTE	BESKRYWING
fHerstelwerk	Die beskrywing van die herstelwerk wat nodig is
fUre	Die aantal ure wat nodig is om herstelwerk te voltooi
fKoste	Die arbeidskoste vir EEN herstel
fTotaleKoste	Die totale koste vir alle herstelwerk

Voltooi die kode in die objekklas, soos dit hieronder in VRAAG 3.1.1 tot VRAAG 3.1.4 beskryf word.

- 3.1.1 Skryf kode vir 'n **konstruktor**-metode, **Create**, wat die naam van die herstelwerk (beskrywing van wat op die motor herstel moet word) en die aantal ure wat herstelwerk sal neem as 'n heelgetal, sal ontvang. Ken die parameterwaardes aan die korrekte attribute toe en stoor die koste van herstelwerk as $70 * \text{ure}$ in die koste-attribuu (R70 arbeidskoste per uur). (4)
- 3.1.2 Skryf kode vir 'n metode met die naam **setTotaleKoste** wat 'n heelgetal wat die aantal ure verteenwoordig, moet ontvang en dit toe ken aan die attribuu, **fTotaleKoste**, nadat die parameter met 70 vermenigvuldig is. (4)
- 3.1.3 Skryf kode vir 'n metode met die naam **BerekenDae** wat 'n heelgetal parameter ontvang en 'n heelgetal waarde terugstuur gebaseer op die volgende reëls:
- Deel die parameter deur 8 en as die antwoord 'n breuk is, dan moet die getal na die naaste heelgetal opgerond word.
 - Enige herstelwerk wat minder as of gelyk aan 8 ure neem, word beskou as een dag se werk.

Voorbeelde:

URE	DAE
10	2
9	2
23	3
2	1
1	1

(6)

- 3.1.4 Skryf kode vir 'n metode met die naam **toString** wat 'n string moet terugstuur. Die string moet die attribute van die klas in die volgende formaat bymekaar voeg:

<Naam van herstelwerk>

Ure om te voltooi: <ure>

Arbeidskoste: <koste per eenheid in geldeenheid>

Voorbeeld van afvoer:

<i>Air and Cabin Filter Replacement</i> <i>Ure om te voltooi: 1</i> <i>Arbeidskoste: R70.00</i>

(4)

TOTAAL VRAAG 3.1: (18)

3.2 'n Onvolledige eenheid **Vraag3_u.pas** word voorsien.

Dit bevat kode vir die objekklas om toeganklik te wees. 'n Globale objekveranderlike, **objHerstel**, is reeds verklaar.

MOENIE gegewe kode verwyder of verander NIE.

Volg die instruksies hieronder om die kode te voltooi:

3.2.1 Knoppie V 3.2.1 Voeg by kwotasie

Die gebruiker moet 'n herstelwerk en die aantal ure wat benodig word vir herstel in die gegewe toevoerkomponente, **IstHerstelwerk** en **sedUre**, kies.

Skryf kode om die volgende te doen:

- Instansieer die objek, **objHerstel**, deur die naam van herstelwerk en aantal ure vir herstel te gebruik.
- Gebruik 'n metode van die klas om die naam van herstelwerk, aantal ure en herstelkoste te vertoon.
- 'n Globale veranderlike, **iTotaleUre**, is reeds geskep en jy moet 'n metode van die klas gebruik om die ure by die globale veranderlike te tel.

Voorbeeld van afvoer:

```
TLC Car Repairs Kwotasie

Oil Change and Oil Filter Replacement.
Ure om te voltooi: 4
Arbeidskoste: R280.00

Air and Cabin Filter Replacement
Ure om te voltooi: 6
Arbeidskoste: R420.00

Battery Replacement
Ure om te voltooi: 1
Arbeidskoste: R70.00

Brake Repair.
Ure om te voltooi: 6
Arbeidskoste: R420.00
```

(11)

3.2.2 Knoppie V 3.2.2 Finaliseer Kwotasie

Doen die volgende:

- Gebruik die globale veranderlike, **iTotaleUre** en metode van die klas om die totale koste van herstelwerk te bepaal.
- Kry die totale koste van herstelwerk deur die metode van die klas te gebruik en stoor dit in 'n veranderlike.
- Gebruik die 'checkbox', **cbxParte**, en voeg R150 vir vervoer/ afleweringkoste by die totale koste wanneer die parte gekoop moet word.
- Vertoon die aantal dae wat dit sal neem om al die herstelwerk in die 'richedit' **redAfvoer** te voltooi deur 'n metode van die klas en die globale veranderlike **iTotaleUre** te gebruik.
- Vertoon die totale koste vir herstelwerk in die 'richedit' met die naam **redAfvoer**.

Voorbeeld van afvoer:

TLC Car Repairs Kwotasie

Oil Change and Oil Filter Replacement.
Ure om te voltooi: 4
Arbeidskoste: R280.00

Air and Cabin Filter Replacement
Ure om te voltooi: 6
Arbeidskoste: R420.00

Battery Replacement
Ure om te voltooi: 1
Arbeidskoste: R70.00

Brake Repair.
Ure om te voltooi: 6
Arbeidskoste: R420.00

Totale dae om al die werk te voltooi = 3
Totale arbeidskoste = R1 190.00

As die keuseblokkie gekies is (Koop Parte), dan is die afvoer soos hieronder:

Totale dae om al die werk te voltooi = 3
Totale arbeidskoste = R1 340.00

(11)

TOTAAL VRAAG 3.2: (22)

- Sleutel jou naam en van as kommentaar in die eerste reël van die programlêer. (In albei eenhede – klaseenheid en hoofprogram wat die klas gebruik)
- Stoor jou programme.
- 'n Drukstuk van jou kode vir albei eenhede mag vereis word.

[40]

VRAAG 4: PROBLEEMOPLOSSING PROGRAMMERING

'n Verkiesing vind plaas in 10 distrikte van 'n provinsie en daar is net twee partye, A en B. Jy moet die program voltooi om te bepaal watter party die verkiesing gewen het.

Doen die volgende:

- Maak die onvoltooide program in die **Vraag 4**-lêergids oop.
- Sleutel jou naam en van as kommentaar in die eerste reël van die **Vraag4_u.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie.

Voorbeeld van die tekslêer met die naam **stemme.txt**

1225
2141
1565
2174
3158
1396
1525
1841
2734
2556

Voltooi die kode vir elke vraag, VRAAG 4.1 en VRAAG 4.2.

LET WEL:

- Goeie programmeringstegnieke moet toegepas word in die ontwerp en kodering van jou oplossing.
- Jy mag GEEN gegewe kode verander NIE.

Gegee:

'n Globale skikking, **ar2Stemme**, is verklaar.

'n Tekslêer, **stemme.txt**, is in die Vraag 4-gidslêer en bevat 20 reëls van stemme vir kandidate A en B vir die 10 distrikte.

4.1 Knoppie V 4.1 Kry stemme

- Die twee-dimensionele skikking met die naam **ar2Stemme** bestaan uit 10 kolomme vir die 10 distrikte en 2 rye vir die twee kandidate, A en B.
- Ry 1 van die skikking is vir kandidaat A se stemme en ry 2 is kandidaat B se stemme.
- Die eerste reël in die tekslêer is die aantal stemme vir kandidaat A en die tweede reël is die aantal stemme vir kandidaat B, ens. (Al die ewegetal rye in die tekslêer is kandidaat B se stemme en al die ongelyke rye is kandidaat A se stemme.)

Skryf kode om die volgende te doen:

- Lees die reëls van die tekslêer, **stemme.txt** en stoor dit in die twee-dimensionele skikking, **ar2Stemme**.
- Daar is geen afvoer wat vertoon moet word nie.

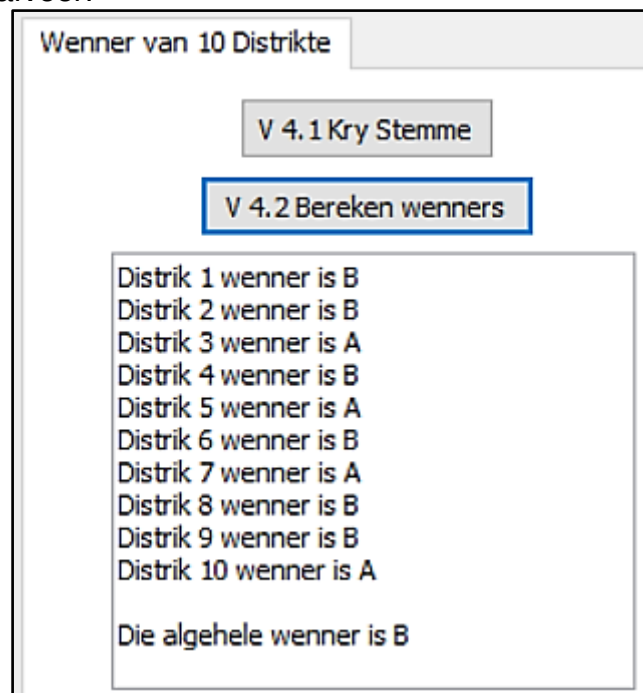
(12)

4.2 Knoppie V 4.2 Bereken wenners

Jy moet die twee-dimensionele skikking, **ar2Stemme**, gebruik om kode te skryf om die wenner vir elke distrik te bepaal en vertoon.

- Vertoon elke distriknaam en wenner vir daardie distrik in die 'richedit', **redV4** vir die 10 distrikte.
- Skryf kode om te tel watter kandidaat die meeste distrikte gewen en vertoon die algehele wenner in **redV4**.

Voorbeeld van afvoer:



(18)

- Sleutel jou naam en van as kommentaar in die eerste reël van die programlêer.
- Stoor jou program.
- 'n Drukstuk van jou kode mag vereis word.

[30]

TOTAAL: 150



Province of the
EASTERN CAPE
EDUCATION



NASIONALE SENIOR SERTIFIKAAT

GRAAD 12

SEPTEMBER 2022

**INLIGTINGSTEGNOLOGIE V1
NASIENRIGLYN**

PUNTE: 150

Hierdie nasienriglyn bestaan uit 16 bladsye.

NAAM VAN LEERDER:				
TOTAAL VRAAG 1:	TOTAAL VRAAG 2:	TOTAAL VRAAG 3:	TOTAAL VRAAG 4:	TOTAAL
/40	/40	/40	/30	/150

VRAAG 1: ALGEMENE PROGRAMMERINGSVAARDIGHEDE		MAKS. PUNTE	PUNTE BEHAAL
1.1	Knoppie [1.1 Vorm] shpToets.Shape := stcircle; ✓ shpToets.Brush.Color := clgreen; ✓ gbxVraag12.Visible := true; ✓ gbxVraag13.Enabled := true; ✓	4	
1.2	Knoppie [1.2 Ontsyfer] Kry toevoer van edtToevoer ✓ Inisialiseer 'n leë string ✓ Lus totdat toevoer string leeg is ✓ Kry ordinale getal en verwerk na integer ✓ Verwyder ordinale getal van toevoer string ✓ Verwerk ordinale getal na karakter deur CHR te gebruik ✓ Voeg karakter by nuwe string ✓ Vertoon nuwe string in edtAfvoer ✓	8	
1.3	Knoppie [1.3 Toets] Maak die list box skoon ✓ Kry toevoer van spinedit ✓ Inisialiseer die teller vir skikking ✓ <i>Kry die aantal priemfaktore van die toevoer en stoor in skikking – 7 punte:</i> Buitenste Lus van 1 tot toevoer ✓ Stel getal priemfaktore na 0 ✓ Binneste Lus van 1 na buitenste lus telle (geneste lus) ✓ As binneste lus teler 'n priemfaktor is ✓ Vermeerder dan die aantal priemfaktore met 1 ✓ As getal priemfaktore = 2 ✓ Voeg dan die buitenste lus teller by skikking ✓		

	Bereken en vertoon die pare van priemfaktore wat se som gelyk is aan die toevoer – 9 punte: Stel boolse vlag na vals ✓ Buitenste lus van 1 na skikking teller – 1 ✓ Stel veranderlike 1 na skikking[buitenste lus teller] ✓ Binneste lus van buitenste loop +1 na skikking teller ✓ Stel veranderlike 2 na skikking[binneste lus teller] ✓ As veranderlike 1 + veranderlike 2 = toevoer ✓ Vertoon dan albei getalle in listbox ✓ Omgeskakel na string ✓ Stel Boolse vlag na waar ✓ As Boolse vlag = vals is ✓ Vertoon gepaste boodskap in ShowMessage komponent ✓	21	
1.4	Knoppie [1.4 Area] Bereken area van vierkant ✓ Bereken area van sirkel: $\text{Pi} * (0.9/(2 * \text{Pi})) * (0.9/(2 * \text{Pi}))$ ✓ gebruik Pi of 3.1415 ✓ Trek area van vierkant af van area van sirkel ✓ Vermenigvuldig met 100 ✓ Voeg antwoord by huidige 'caption' van lblAfvoer ✓ Rond af na 3 desimale plekke ✓	7	
TOTAAL VRAAG 1		40	

VRAAG 2: DATABASISPROGRAMMERING		MAKS. PUNTE	PUNTE BEHAAL
2.1.1	Knoppie: [V 2.1.1]	3	
	'select Afleveringsadres, AfleveringDatum from tblBestellings order by AfleveringDatum Desc'		
	Konsepte: SELECT twee korrekte velde ✓ FROM korrekte tabel ✓ ORDER BY korrekte veld DESC ✓		
2.1.2	Knoppie: [V 2.1.2]	4	
	'select count(BestellingID) as Items20ofMeer from tblBestellings where items >= 20'		
	Konsepte: SELECT count(enige veld van tblBestellings-tabel) ✓ as Items20ofMeer ✓ FROM korrekte tabel ✓ WHERE items >=20 ✓		
2.1.3	Knoppie: [V 2.1.3]	4	
	'select * from TblBestellings where Afleveringsadres like ' + quotedstr('%' + sline + '%')		
	Konsepte: SELECT * (alle velde) ✓ FROM korrekte tabel ✓ WHERE Afleveringsadres LIKE ✓ quotedstr('%' + sline + '%') ✓		
2.1.4	Knoppie: [V 2.1.4]	3	
	'update tblBestellings set Kollekteer = true where AfleveringDatum is null'		
	Konsepte: Update tblBestellings ✓ Set Kollekteer = true ✓ Where AfleveringDatum is null ✓		
2.1.5	Knoppie: [V 2.1.5]	4	
	'select Afleveringsadres, AfleveringDatum - BestellingDatum as AflewerDae from TblBestellings where Kollekteer = false'		
	Konsepte: Select DeliveryAdress ✓ AfleveringDatum – BestellingDatum ✓ From TblBestellings ✓ Where Kollekteer = false ✓		

2.1.6	Knoppie: [V 2.1.6] 'select WinkelNaam, format(sum(BestellingBedrag,"currency")) as TOTALEVerkope from tblBestellings, tblWinkels where tblWinkels.Winkelsid = tblBestellings.Winkelsid group by WinkelNaam	7	
	Concepts: Select WinkelNaam, ✓ format(sum(BestellingBedrag ✓, "currency")) ✓ as TOTALEVerkope ✓ from TblBestellings, Winkels ✓ where TblWinkels.TblWinkelsID = TblBestellings.WinkelsID ✓ group by WinkelNaam ✓		
	2.1 Subtotaal: SQL	25	
2.2.1	Knoppie: [V 2.2.1] tblWinkels.First ✓ while not tblWinkels.eof do ✓ if tblWinkels['Aanlyn'] = true then ✓ redOutput.Lines.Add(tblWinkels['WinkelNaam']) ✓ tblWinkels.Next ✓	5	
2.2.2	Knoppie: [V 2.2.2] Gaan na eerste ry in tblWinkels en lus ✓ as WinkelNaam = combobox item ✓ stel dan integer veranderlike na tblWinkelsID ✓ Gaan na volgende ry in tblWinkels Stel reële som veranderlike na = 0 ✓ Gaan na eerste ry in tblBestellings en lus as TblWinkelsID = integer veranderlike van eerste lus ✓ Vertoon dan in redAfvoer: BestellingDatum omgeskakel na string ✓ en BestellingBedrag in geldeenheid ✓ Voeg BestellingBedrag by reële som veranderlike ✓ Gaan na volgende ry in TblBestellings ✓ Na die lus vertoon die reële som veranderlike in redAfvoer in geldeenheid ✓	10	
	2.2 Subtotaal: Kode Konstruke	15	
	TOTAAL VRAAG 2	40	

VRAAG 3: OBJEK-GEÏORIENTEERDE PROGRAMMERING		MAKS. PUNTE	PUNTE BEHAAL
3.1.1	Konstruktor Create: Korrekte naam ✓ met een string parameter en een integer parameter ✓ Ken korrekte parameter waardes toe aan korrekte attribute ✓ Stel fKoste = 70 * fUre ✓	4	
3.1.2	prosedure setTotaleKoste Korrekte metode – procedure ✓ Een integer parameter ✓ Stel korrekte attribuut (fTotaleKoste) ✓ = 70 * parameter ✓	4	
3.1.3	Funksie BerekenDae(ure : integer) : integer Korrekte metode – integer funksie ✓ een integer parameter ✓ As ure > 8 ✓ bereken en stuur korrekte dae terug (Ceil of ander metode) ✓ else ✓ Stuur 1 dag terug ✓	6	
3.1.4	funksie toString : string; korrekte string metode ✓ korrekte attribute teruggestuur as result ✓ gebruik #13, ✓ skakel ure en koste om na string ✓	4	
	2.1 Subtotaal: Objekklas	18	
3.2.1	Knoppie [V 3.2.1 Voeg by kwotasie] Kry toevoere van spinedit ✓ en list box ✓ Instansieer die objek Objeknaam = ✓ tHerstel.create ✓ een string, een integer parameter ✓ In korrekte volgorde ✓ Gebruik metode van klas ✓ en objeknaam om te vertoon ✓ Vertoon leë reël in redV3 ✓ Gebruik metode van klas en objeknaam ✓ om by totale ure globale veranderlike te voeg ✓	11	
3.2.2	Knoppie [V 3.2.2 Finaliseer Kwotasie] Gebruik objeknaam en metode van die klas ✓ setTotaleKoste met globale totale ure as parameter ✓ Gebruik objeknaam en metode van die klas ✓ getTotaleKoste en ken aan lokale veranderlike toe ✓ As cbxParte geselekteer is, ✓ voeg 150 by totale koste ✓ Vertoon dae in richedit deur objeknaam ✓ en korrekte metode BerekenDae met iTotaleUre as parameter ✓ as 'n string ✓ Vertoon totale koste in richedit ✓ omgeskakel na geldeenheid ✓	11	
	2.2 Subtotaal: Vormklas	22	
	TOTAAL VRAAG 3	40	

VRAAG 4: PROBLEEMOPLOSSING		MAKS. PUNTE	PUNTE BEHAAL
4.1	Assignfile stelling ✓ Reset teksleër vir lees ✓ Stel kolomteller na 0 ✓ Lus deur teksleër ✓ Lees 'n reël ✓ Inkrementeer kolom ✓ Stel ry = 1 ✓ Voeg reël na skikking, ✓ verander na integer ✓ Lees 'n reël ✓ Stel ry = 2 ✓ Voeg reël na skikking, verander na integer ✓	12	
4.2	Stel tellers van A en B na 0 ✓ Stel kolomteller na 0 ✓ Lus deur skikking ✓ Stel ry = 1 ✓ Voeg 1 by kolom teller ✓ Stel eerste syfer na skikking gebruik ry en kolom tellers [ry,kolom] ✓ Stel ry = 2 ✓ Stel tweede syfer na skikking gebruik ry en kolom-tellers [ry,kolom] ✓ as eerste syfer > tweede syfer ✓ vertoon in richedit korrekte bewoording – distriknommer en A ✓ inkrementeer A teller ✓ else ✓ vertoon in richedit korrekte bewoording – distriknommer en B ✓ inkrementeer B teller ✓ as A teller > B teller ✓ dan vertoon wennen A ✓ else ✓ vertoon wennen B ✓	18	
TOTAAL VRAAG 4		30	

VOORBEELDE VAN OPLOSSINGS**VRAAG 1**

```
procedure TfrmQuestion1.btnQ1_3Click(Sender: TObject);
var inum, i1,i2, k, x, m, ifactors : integer;
arrprime : array[1..100] of integer;
icount: integer;
bprime : boolean;
begin
  lstoutput.clear;

  icount := 0;
  inum := sednumber.Value;

  for k := 1 to inum do
    begin
      ifactors := 0;
      for m := 1 to k do
        begin
          if k mod m = 0 then
            inc(ifactors);
          end;
        if ifactors = 2 then
          begin
            inc(icount);
            arrprime[icount] := k;
          end;
        end;
      end;

  bprime := false;
  for k := 1 to icount - 1 do
    begin
      i1 := arrprime[k];
      for x := k + 1 to icount do
        begin
          i2 := arrprime[x];
          if i1 + i2 = inum then
            begin
              lstoutput.Items.Add(inttostr(i1) + ' + ' + inttostr(i2));
              bprime := true;
            end;
          end;
        end;
      end;

  if bprime = false then
    showmessage(inttostr(inum) + ' cannot be calculated using the sum of two prime
    numbers')
  end;
```

```
procedure TfrmQuestion1.btnQ1_4Click(Sender: TObject);
var rsq, rc : real;
rr : real;
begin
rsq := 0.2 * 0.2;
rc := Pi * (0.9/(2 * Pi)) * (0.9/(2 * Pi));
rr := (rc - rsq) * 100;
lbloutput.Caption := lbloutput.Caption + floattostrf(rr,ffixed,8,3) + ' square cm';
end;
```

```
procedure TfrmQuestion1.btnQ1_1Click(Sender: TObject);
begin
shpToets.Shape := stcircle;
shpToets.Brush.Color := clgreen;
gbquestion1_2.Visible := true;
gbquestion1_3.Enabled := true;
end;
```

```
procedure TfrmQuestion1.btnQ1_2Click(Sender: TObject);
var sline : string;
snew , sword : string;
k, inum : integer;
begin
// If text in edinput is erased use the following as input:
//      '73 84 32 105 115 32 101 97 115 121 33'
```

```
sline := edinput.Text;
snew := "";
while length(sline) <> 0 do
begin
    sword := copy(sline,1, pos(' ',sline) - 1);
    delete(sline,1, pos(' ',sline));
    inum := strtoint(sword);
    snew := snew + chr(inum);
end;
edtoutput.text := snew;
end;

end.
```

VRAAG 2

```
procedure TQuestion_2.btnQuestion2_1_1Click(Sender: TObject);
// Provided code - do not change
var
    sSQL1: String;
begin
    /// enter your code below//
    sSQL1 := 'select Afleveringsadres, AfleveringDatum from tblBestellings order by
AfleveringDatum desc';

    // Provided code - do not change
    dbCONN.runSQL(sSQL1);
    if length(ssql1) <> 0 then
        dbconn.setgridforsql1(dbgsq1);
end;

procedure TQuestion_2.btnQuestion2_1_2Click(Sender: TObject);
// Provided code - do not change
var
    sSQL2: String;
begin
    /// enter your code below//
    sSQL2 := 'select count(BestellingID) as TblBestellings20orMore from tblBestellings
where items >= 20';

    // Provided code - do not change
    dbCONN.runSQL(sSQL2);
end;

procedure TQuestion_2.btnQuestion2_1_3Click(Sender: TObject);
// Provided code - do not change
var
    sline : string;
    sSQL3: String;
begin
    sline := inputbox('Enter a street name','Nxolo Street');
    /// enter your code below//
    sSQL3 := 'select * from TblBestellings where Afleveringsadres like ' + quotedstr('%' +
sline + '%)';

    // Provided code - do not change
    dbCONN.runSQL(sSQL3);
    if length(ssql3) <> 0 then
        dbconn.setgridforsql3(dbgsq1);
end;

procedure TQuestion_2.btnQuestion2_1_4Click(Sender: TObject);
// Provided code - do not change
var
    sSQL4: String;
begin
    /// enter your code below//
    sSQL4 := 'update tblBestellings set Kollekteer = true where AfleveringDatum is null';
```

```
// Provided code - do not change
dbCONN.executeSQL(sSQL4,dbgtblWinkels,dbgorder,dbgsql);
if length(ssql4) <> 0 then
    dbconn.setgridforsql3(dbgsql);
end;

procedure TQuestion_2.btnQuestion2_1_5Click(Sender: TObject);
// Provided code - do not change
var
    sSQL5: String;
begin
    /// enter your code below//
    sSQL5 := 'select Afleveringsadres, AfleveringDatum - BestellingDatum as AfleverDae
from TblBestellings where Kollekteer = false';

    // Provided code - do not change
    dbCONN.runSQL(sSQL5);
    if length(ssql5) <> 0 then
        dbconn.setgridforsql5(dbgsql);
    end;

procedure TQuestion_2.btnQuestion2_1_6Click(Sender: TObject);
// Provided code - do not change
var
    sSQL6: String;
begin
    /// enter your code below//
    sSQL6 := 'select WinkelNaam, format(sum(BestellingBedrag),"currency") as
TOTALEVerkope from tblBestellings, tblWinkels where tblWinkels.tblWinkelsid =
tblBestellings.tblWinkelsid group by WinkelNaam';

    // Provided code - do not change
    dbCONN.runSQL(sSQL6);
    if length(ssql6) <> 0 then
        dbconn.setgridforsql6(dbgsql);
    end;

procedure TQuestion_2.btnQuestion2_2_1Click(Sender: TObject);
begin
    // Provided code - do not change
    redoutput.Clear;
    redoutput.Lines.Add('TblWinkelss that are available for Aanlyn tblWinkelsping');
    redoutput.Lines.Add('-----');
    /// enter your code below//
    tblWinkels.First;
    while not tblWinkels.eof do
        begin
            if tblWinkels['Aanlyn'] = true then
                redoutput.Lines.Add(tblWinkels['WinkelNaam']);
            tblWinkels.Next;
        end;
    end;
end;
```

```
procedure TQuestion_2.btnQuestion2_2_2Click(Sender: TObject);
var stblWinkels : string; // Provided code - do not change
bfound : boolean;
inum : integer;
rsum : real;
begin
// Provided code - do not change
redoutput.Clear;
redoutput.Lines.Add('Order Date' + #9 + 'Amount of order');
redoutput.Lines.Add('-----') ;
stblWinkels := cmbtblWinkels.Text;
/// enter your code below//
bfound := false;
tblWinkels.First;
while (not tblWinkels.Eof) and (bfound = false) do
begin
if tblWinkels['WinkelNaam'] = stblWinkels then
begin
inum := tblWinkels['tblWinkelsid'];
bfound := true;
end;
tblWinkels.Next;
end;
rsum := 0;
tblBestellings.first;
while not tblBestellings.eof do
begin
if tblBestellings['tblWinkelsid'] = inum then
begin
redoutput.Lines.Add(datetostr(tblBestellings['BestellingDatum']) + #9 +
floattostf(tblBestellings['BestellingBedrag'],ffcurrency,8,2));
rsum := rsum + tblBestellings['BestellingBedrag'];
end;
tblBestellings.Next;
end;
redoutput.Lines.Add("");
redoutput.Lines.Add('Total amount ordered: ' + floattostf(rsum,ffcurrency,8,2));
end;
```

VRAAG 3**Klaseenheid:**

unit Question3ClassDefinition;

interface

uses sysutils, math, dialogs;

type

TRepair = class

private

frepairname : string;

fhours : integer;

fcost : real;

ftotalcost : real;

public

constructor create(sname : string; ihrs : integer);

function calculatedays(ihrs : integer) : integer;

function gethours : integer;

function tostring : string;

procedure settotalcost(ihrs : integer);

function gettotalcost : real;

end;

implementation

{ TRepair }

constructor TRepair.create(sname: string; ihrs: integer);

begin

frepairname := sname;

fhours := ihrs;

fcost := 70 * fhours;

end;

// provided code do not delete//

function TRepair.gethours: integer;

begin

result := fhours;

end;

// provided code do not delete//

function TRepair.gettotalcost: real;

begin

result := ftotalcost;

end;

procedure TRepair.settotalcost(ihrs: integer);

begin

ftotalcost := ihrs * 70;

end;

```
function TRepair.calculatedays(ihrs: integer) : integer;
var idays : integer;
begin
  if ihrs > 8 then
    begin
      idays := ceil(ihrs/8);
    end
  else
    idays := 1;
  result := idays;
end;
```

```
function TRepair.tostring: string;
begin
  result := frepairname +
    #13 + 'Hours to complete: ' + inttostr(fhours) +
    #13 + 'Labour Cost: ' + floattostrf(fcost,ffcurrency,8,2)
end;

end.
```

Hoofteenheid:

```
procedure TForm1.btnQ3_2_1Click(Sender: TObject);
begin
  objrepair := trepair.create(lstrepairs.Items[lstrepairs.ItemIndex],sedhours.value);
  redq3.Lines.Add(objrepair.tostring);
  redq3.Lines.Add("");
  itotalhours := itotalhours + objrepair.gethours;
end;

procedure TForm1.cbxBuyPartsClick(Sender: TObject);
var rcost : real;
begin
  objrepair.settotalcost(itotalhours);
  rcost := objrepair.gettotalcost;

  if cbxparts.checked = true then
    rcost := rcost + 150;

  redq3.Lines.Add("");
  redq3.Lines.Add('Total days to complete all jobs = ' +
    inttostr(objrepair.calculatedays(itotalhours)));
  redq3.Lines.Add('Total labour cost = ' + floattostrf(rcost,ffcurrency,8,2));
end;
```

VRAAG 4

```
unit Question4_u;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, Grids, ComCtrls;

type
  TForm1 = class(TForm)
    pgc: TPageControl;
    tbsvoting: TTabSheet;
    btnQ4_1: TButton;
    btnQ4_2: TButton;
    redQ4: TRichEdit;
    procedure btnQ4_1Click(Sender: TObject);
    procedure btnQ4_2Click(Sender: TObject);

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

var
  Form1: TForm1;
  ar2Stemme : array[1..2,1..10] of integer;

implementation
{$R *.dfm}

procedure TForm1.btnQ4_1Click(Sender: TObject);
var myfile : textfile;
    irow, icol, inum: integer;
    sline : string;
begin
  assignfile(myfile, 'stemme.txt');
  reset(myfile);
  icol := 0;
  while not eof(myfile) do
    begin
      readln(myfile, sline);
      inc(icol);
      irow := 1;
      ar2Stemme[irow, icol] := strtoint(sline);
      readln(myfile, sline);
      irow := 2;
      ar2Stemme[irow, icol] := strtoint(sline);
    end;
  end;
end;
```

```
procedure TForm1.btnQ4_2Click(Sender: TObject);
var
irow, icol, inum1, inum2, k, ia, ib : integer;
begin
    icol := 0;
    ia := 0;
    ib := 0;
    while icol < 10 do
        begin
            irow := 1 ;
            inc(icol);
            inum1 := ar2Stemme[irow,icol] ;
            irow := 2;
            inum2 := ar2Stemme[irow,icol];
            if inum1 > inum2 then
                begin
                    redq4.Lines.Add('Distrik '+ inttostr(icol) + ' wenner is A');
                    inc(ia);
                end
            else
                begin
                    redq4.Lines.Add('Distrik '+ inttostr(icol) + ' wenner is B');
                    inc(ib);
                end;
            end;
            redq4.Lines.Add("");
        if ia > ib then
            redq4.Lines.Add('Die algehele wenner is A')
        else
            redq4.Lines.Add('Die algehele wenner is B')
        end;

    end.
end.
```