

Vertroulik



# basic education

Department:  
Basic Education  
**REPUBLIC OF SOUTH AFRICA**

## **SENIORSERTIFIKAAT-EKSAMEN/ NASIONALE SENIORSERTIFIKAAT-EKSAMEN**

**INLIGTINGSTEGNOLOGIE V1**

**MEI/JUNIE 2025**

**PUNTE: 150**

**TYD: 3 uur**

**Hierdie vraestel bestaan uit 24 bladsye, 2 bladsye met data,  
2 bladsye vir beplanning en 'n aparte inligtingsblad.**

**INSTRUKSIES EN INLIGTING .**

1. Hierdie vraestel is in VIER afdelings ingedeel. Kandidate moet AL die vrae in AL VIER afdelings beantwoord.
2. Twee leë bladsye wat vir beplanningsdoeleindes gebruik kan word, is aan die einde van die vraestel ingesluit.
3. 'n Inligtingsblad is voorsien wat jy aan die einde van die eksamensessie moet voltooi. Maak seker dat AL die inligting wat jy verskaf het, korrek is en lewer die inligtingsblad in voordat jy die eksamensentrum verlaat.
4. 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.
5. Hierdie vraestel is opgestel met programmeringsterme wat kenmerkend vir Delphi as programmeringstaal is. Die Delphi-programmeringstaal moet gebruik word om die vrae te beantwoord.
6. Maak seker dat jy die vrae beantwoord volgens die spesifikasies wat in elke vraag gegee word. Punte sal volgens die voorgeskrewe vereistes toegeken word.
7. Beantwoord slegs wat in elke vraag gevra word. Byvoorbeeld, indien die vraag nie vir datavalidering vra nie, sal geen punte vir datavalidering toegeken word nie.
8. 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.
9. Roetines, soos soek, sorteer en seleksie, moet vanuit eerste beginsels ontwikkel word. Jy mag NIE die ingeboude funksies van die Delphi-programmeringstaal vir enige van hierdie roetines gebruik NIE.
10. Alle datastrukture moet deur jou, die programmeerder, verklaar word, tensy die datastrukture verskaf word.
11. 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.
12. Maak seker dat jou eksamennommer as kommentaar verskyn in elke program wat jy kodeer, asook op elke gebeurtenis wat aangedui word.
13. Indien dit vereis word, druk die programmeringskode van al die programme/klasse wat jy voltooi het. Jou eksamennommer moet op alle drukstukke voorkom. Jy sal ná die eksamensessie 'n halfuur tyd vir drukwerk gegee word.
14. 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....

15. Die lêers wat jy nodig 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.

Doen die volgende:

- Dubbelklik op die volgende uitvoerbare lêer wat met 'n wagwoord beskerm is: **DataJUN2025.exe**
- Klik op die 'Extract'-knoppie.
- Sleutel die volgende wagwoord in: **GAMES@J2025**

Nadat dit onttrek ('extracted') is, sal die volgende lys lêers in die lêergids ('folder') **DataJUN2025** beskikbaar wees:

**Question 1:**

DataQ1\_3.txt  
Question1\_P.dpr  
Question1\_P.dproj  
Question1\_P.res  
Question1\_U.dfm  
Question1\_U.pas

**Question 2:**

CompGamesDB - Copy.mdb  
CompGamesDB.mdb  
ConnectDB\_U.pas  
Question2\_P.dpr  
Question2\_P.dproj  
Question2\_P.res  
Question2\_U.dfm  
Question2\_U.pas

**Question 3:**

Game\_U.pas  
Question3\_P.dpr  
Question3\_P.dproj  
Question3\_P.res  
Question3\_U.dfm  
Question3\_U.pas

**Question 4:**

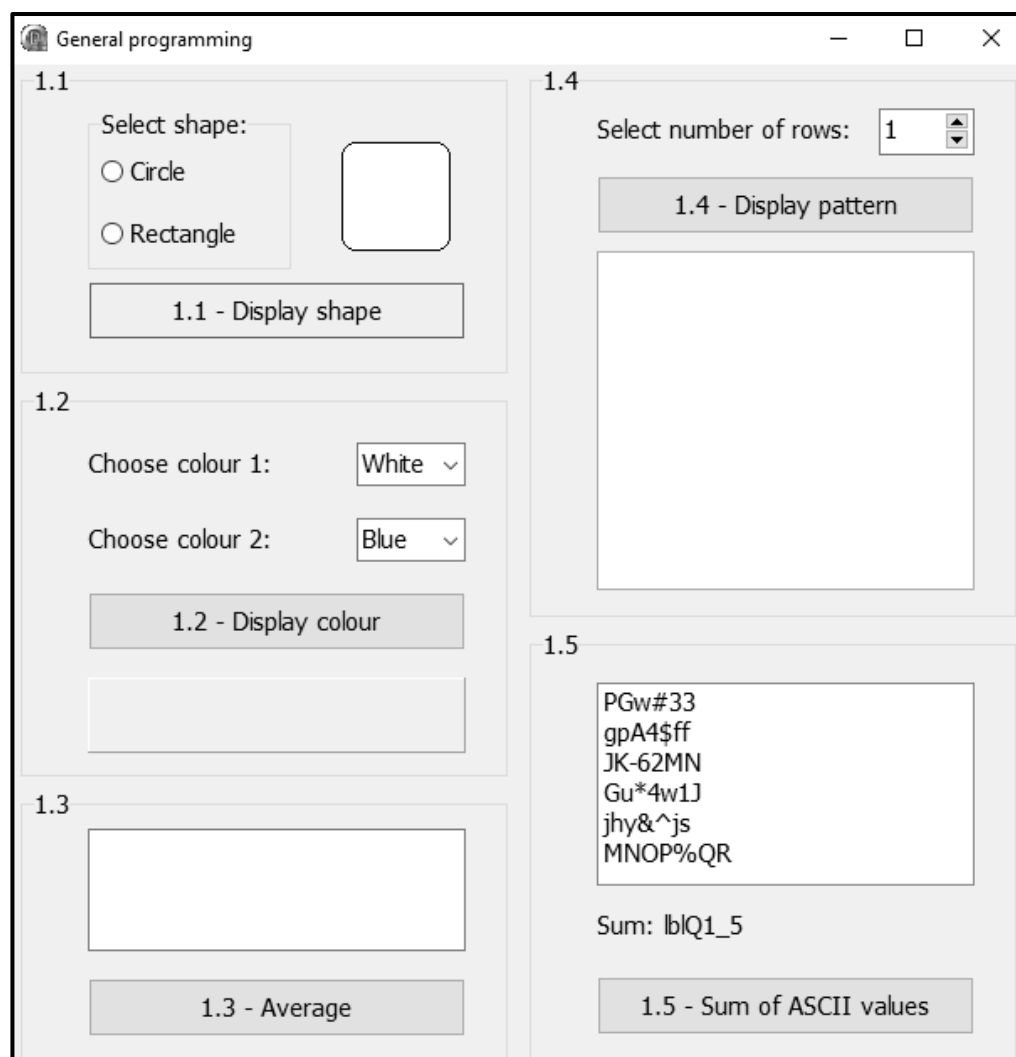
Question4\_P.dpr  
Question4\_P.dproj  
Question4\_P.res  
Question4\_U.dfm  
Question4\_U.pas

**AFDELING A****VRAAG 1: ALGEMENE PROGRAMMERINGSVAARDIGHEDE**

Doen die volgende:

- Maak die onvolledige program in die **Question 1**-lêergids oop.
- Sleutel jou eksamenommer as kommentaar in die eerste reël van die **Question1\_U.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie.

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



- Voltooi die kode vir elke afdeling van VRAAG 1, soos wat in VRAAG 1.1 tot VRAAG 1.5 beskryf is.

### 1.1 Knoppie [1.1 - Display shape]

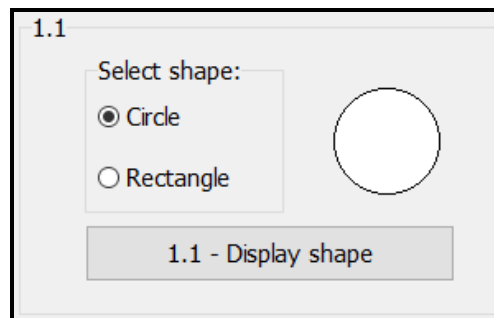
Die gebruiker moet 'n vorm uit die radiogroep ('radio group') **rgpQ1\_1** selekteer.

Kode is in die **OnCreate**-gebeurtenishanteerder ('event handler') voorsien om die sigbaarheidsienskap ('visibility property') van die **shpQ1\_1**-komponent op *true* te stel.

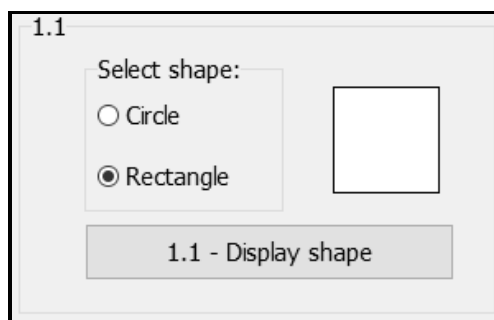
Skryf kode om die volgende te doen:

- Onttrek ('Extract') die geselekteerde vorm-opsie uit die radiogroep **rgpQ1\_1**.
- Vertoon die geselekteerde vorm in die vormkomponent **shpQ1\_1**.

Voorbeeld van afvoer as die 'Circle'-opsie in **rgpQ1\_1** geselekteer is:



Voorbeeld van afvoer as die 'Rectangle'-opsie in **rgpQ1\_1** geselekteer is:



(5)

### 1.2 Knoppie [1.2 - Display colour]

Die gebruiker moet 'n kleur uit elk van die kombinasieblokkies ('combo boxes'), **cmbColour1** en **cmbColour2**, wat voorsien is, onderskeidelik selekteer.

Kode is voorsien om die volgende te doen:

- Stel die kleur van die paneel **pnIQ1\_2** op `c1BtnFace`
- Maak die byskrif ('caption') van die paneel **pnIQ1\_2** skoon.

Skryf kode om die volgende te doen:

- Onttrek ('Extract') en stoor die kleure wat uit die kombinasieblokkies, **cmbColour1** en **cmbColour2**, geselekteer is.
- Bepaal die kleurkombinasie wat op die volgende voorwaardes gebaseer is:
  - As die kombinasie van kleure beide 'Red' en 'Blue' insluit, stel die kleur van die paneel **pnlQ1\_2** op pers.
  - Of anders, stel die byskrif ('caption') van **pnlQ1\_2** op 'No colour'.

Voorbeeld van afvoer as die kleure 'Red' en 'Blue' geselekteer is. Die kleur van die paneel verander na pers:

Voorbeeld van afvoer as enige kombinasie van kleure, wat beide 'Red' en 'Blue' uitsluit, geselekteer word:

(8)

### 1.3 Knoppie [1.3 - Average]

'n Tekslêer met die naam **DataQ1\_3.txt** is voorsien. Die tekslêer bevat 'n onbekende aantal heelgetalwaardes.

Voorbeeld van die eerste vyf reëls data in die tekslêer **DataQ1\_3.txt**:

22  
21  
23  
42  
124

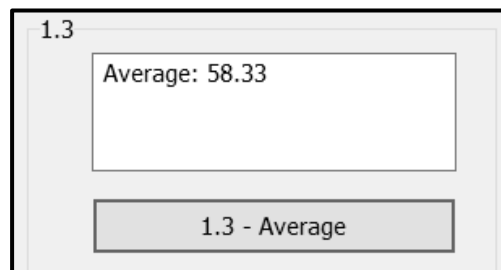
Die gemiddeld van die getalle in die tekslêer moet bereken en vertoon word.

Skryf kode om die volgende te doen:

Toets of die tekslêer **DataQ1\_3.txt** bestaan of nie.

- As die lêer NIE bestaan NIE, vertoon 'n geskikte boodskap in die memo-komponent **memQ1\_3** en maak die program toe.
- As die lêer bestaan:
  - Gebruik 'n voorwaardelike lus om die getalle uit die tekslêer te lees.
  - Bepaal die gemiddeld van die getalle wat uit die tekslêer gelees is.
  - Vertoon die gemiddeld, afgerond tot TWEE desimale plekke, in die memo-komponent **memQ1\_3**.

Voorbeeld van afvoer:



(10)

#### 1.4 Knoppie [1.4 - Display pattern]

'n Patroon, wat bestaan uit syfers met die waarde 1 in ry 1, die waarde 2 in ry 2, die waarde 3 in ry 3 en so aan, moet vertoon word. Die getal rye en die getal syfers per ry sal afhang van die waarde wat deur die gebruiker uit die spin-redigeerblokkie ('spin edit') **spnQ1\_4** geselekteer is.

Voorbeeld van die patroon as die gebruiker die waarde 4 selekteer:

```
1
22
333
4444
```

Kode is voorsien om die 'rich edit'-komponent **redQ1\_4** skoon te maak.

Skryf kode om die volgende te doen:

- Onttrek ('Extract') die getal rye wat uit die spin-redigeerblokkie **spnQ1\_4** geselekteer is.
- Gebruik geneste lusse en die geselekteerde getal rye om die patroon wat hierbo beskryf is, in die **redQ1\_4**-komponent saam te stel en te vertoon.

Voorbeeld van toevoer en afvoer as die getal rye wat geselekteer is, drie is:

Voorbeeld van toevoer en afvoer as die getal rye wat geselekteer is, ses is:

(8)

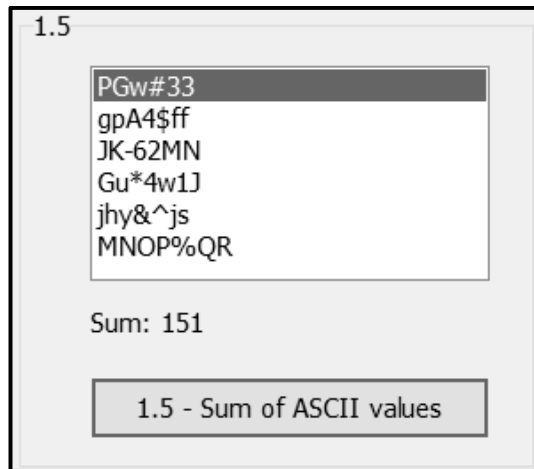
### 1.5 Knoppie [1.5 - Sum of ASCII values]

'n Gebruiker moet 'n string uit die lysblokkie ('list box') **IstQ1\_5** wat voorsien is, selekteer. Die som van die ASCII-waardes wat slegs met die hoofletters in die string geassosieer word, moet bereken en vertoon word.

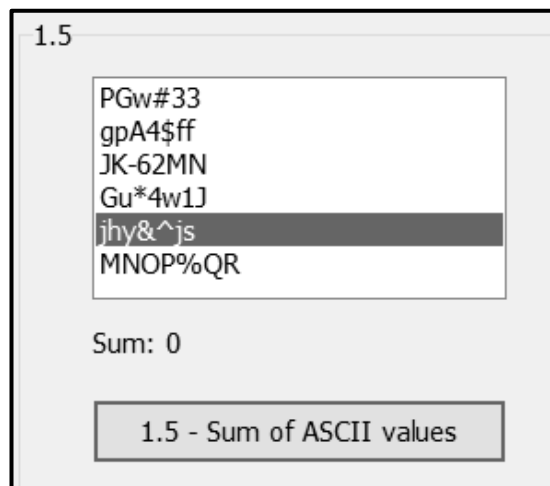
Skryf kode om die volgende te doen:

- Verklaar 'n veranderlike om die som van die waardes te stoor.
- Onttrek ('Extract') en stoor die string wat uit die lysblokkie **IstQ1\_5** geselekteer is.
- Gebruik 'n lus om deur die karakters in die string te stap.  
As 'n karakter 'n hoofletter is:
  - Bepaal die ASCII-waarde van die hoofletter
  - Tel die waarde by die som-veranderlike
- Vertoon die som van die ASCII-waardes van die hoofletters in die etiket ('label') **IblQ1\_5**.

Voorbeeld van afvoer as die string 'PGw#33' uit die lysblokkie geselekteer is:



Voorbeeld van afvoer as die string 'jhy&^js' uit die lysblokkie geselekteer is:



(9)

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

**TOTAAL AFDELING A: 40**

**AFDELING B**

**VRAAG 2: DATABASISPROGRAMMERING**

'n Rekenaarspeletjiesmaatskappy het hulp nodig om navrae te genereer deur 'n omvattende databasis wat gewilde videospelletjies bevat, te gebruik.

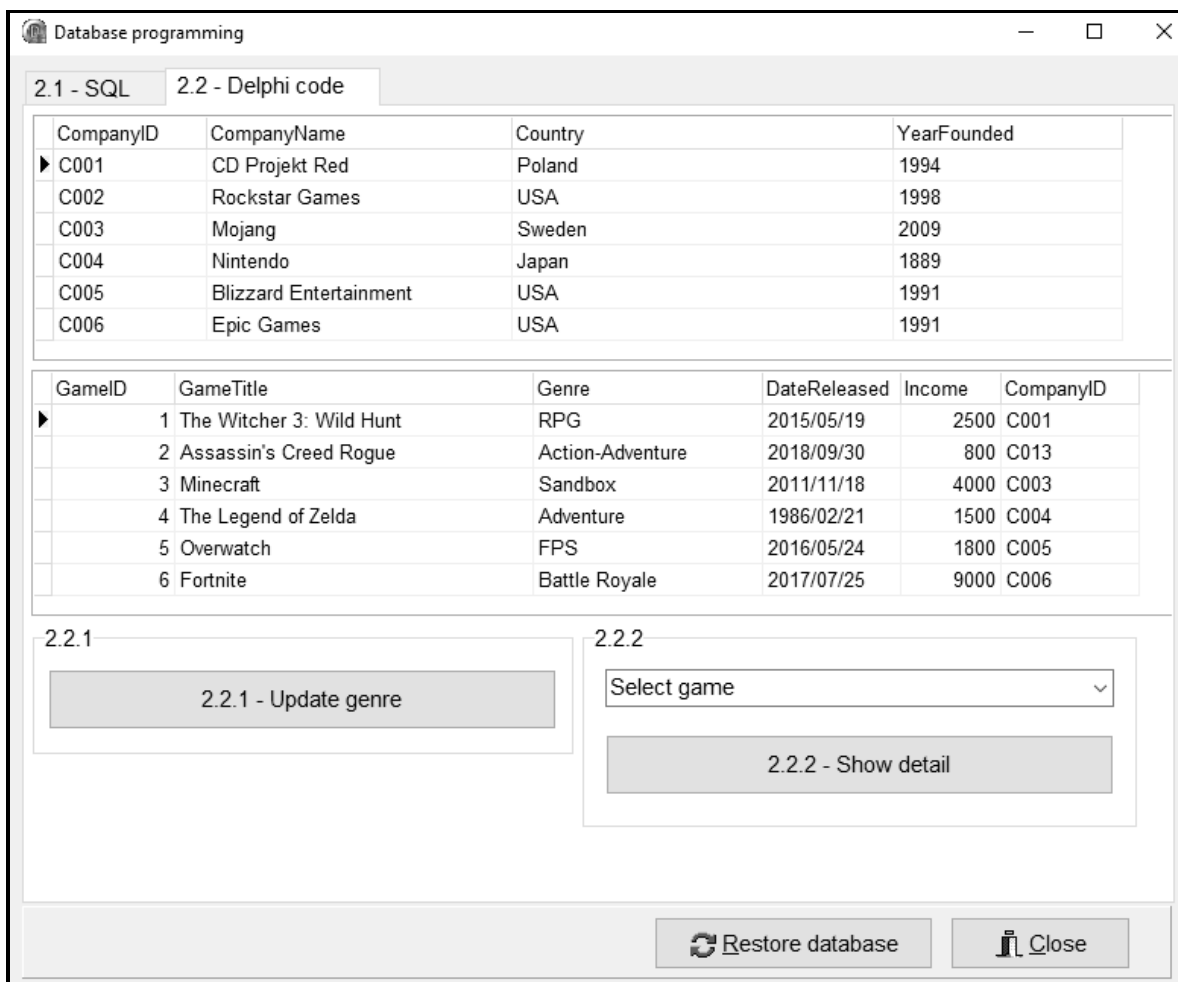
'n Databasis met die naam **CompGamesDB.mdb**, wat inligting bevat oor gewilde videospelletjies wat voor die jaar 2020 vrygestel is, is geskep.

Die databasis bevat twee tabelle met die name **tblCompanies** en **tblGames**.

Die bladsye met data wat aan die einde van die vraestel aangeheg is, verskaf inligting oor die ontwerp en die inhoud van die **CompGamesDB.mdb**-databasis.

Doen die volgende:

- Maak die onvolledige projeklêer met die naam **Question2\_P.dpr** in die **Question 2**-lêergids oop.
- Sleutel jou eksamennummer as kommentaar in die eerste reël van die **Question2\_U.pas**-eenheidlêer in.
- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie. Die inhoud van die tabelle word vertoon, soos wat hieronder op die seleksie van die oortjieblad ('tab sheet') **2.2 - Delphi code** getoon word.



- Volg die instruksies hieronder om die kode vir elke afdeling wat in VRAAG 2.1 en VRAAG 2.2 beskryf word, te voltooi.
- Gebruik SQL-stellings om VRAAG 2.1 te beantwoord en Delphi-kode om VRAAG 2.2 te beantwoord.

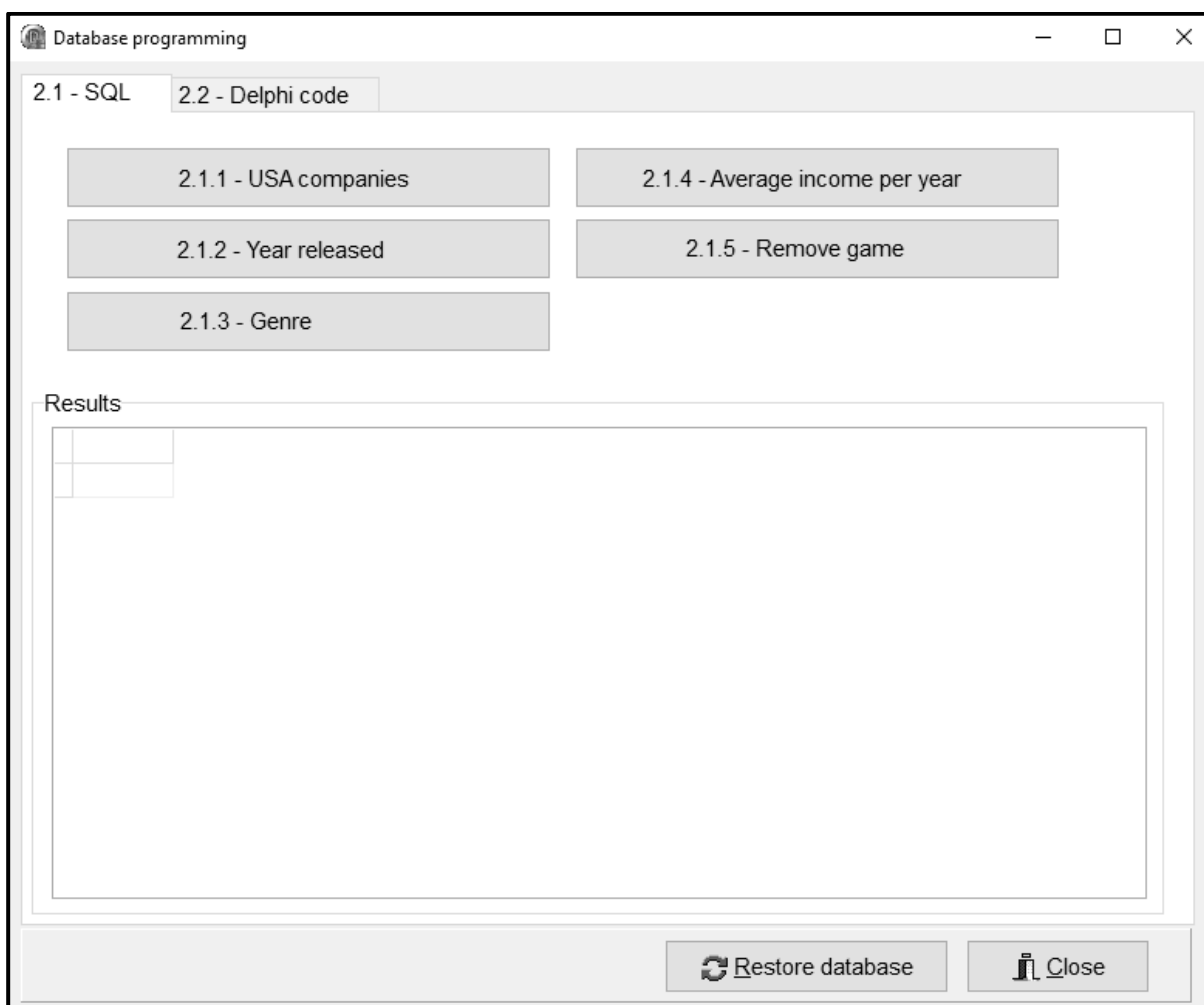
**LET WEL:**

- Die 'Restore database'-knoppie word voorsien om die data wat in die databasis voorkom, na die oorspronklike inhoud te herstel.
- Die inhoud van die databasis is met 'n wagwoord beskerm, m.a.w. jy sal NIE toegang tot die inhoud van die databasis kan verkry deur Microsoft Access te gebruik NIE.
- Kode word voorsien om die GGK('GUI')-komponente met die databasis te koppel. MOENIE enige van die kode wat voorsien is, verander NIE.
- TWEE veranderlikes is as publieke veranderlikes verklaar, soos in die tabel hieronder beskryf word.

Veranderlike	Datatype	Beskrywing
tblCompanies	TADOTable	Verwys na die tabel <b>tblCompanies</b>
tblGames	TADOTable	Verwys na die tabel <b>tblGames</b>

**2.1 Oortjieblad ('Tab sheet') [2.1 - SQL]**

Voorbeeld van die grafiese gebruikerskoppelvlak (GGK) vir VRAAG 2.1:



**LET WEL:**

- Gebruik SLEGS SQL-stellings om VRAAG 2.1.1 tot VRAAG 2.1.5 te beantwoord.
- Kode om die SQL-stellings uit te voer en die resultate van die navrae te vertoon, is voorsien. Die SQL-stellings wat aan die veranderlikes **sSQL1**, **sSQL2**, **sSQL3**, **sSQL4** en **sSQL5**-veranderlikes toegeken is, is onvolledig.

Voltooi die SQL-stellings om die take wat in VRAAG 2.1.1 tot VRAAG 2.1.5 hieronder beskryf word, uit te voer.

**2.1.1 Knoppie [2.1.1 - USA companies]**

Vertoon AL die velde van AL die maatskappye in die **tblCompanies**-tabel wat in die VSA ('USA') gestig is.

Voorbeeld van afvoer van die eerste vier rekords:

CompanyID	CompanyName	Country	YearFounded
C002	Rockstar Games	USA	1998
C005	Blizzard Entertainment	USA	1991
C006	Epic Games	USA	1991
C007	InnerSloth	USA	2015

(3)

**2.1.2 Knoppie [2.1.2 - Year released]**

Vertoon die **GameTitle**, **Genre** en die **DateReleased** van al die speletjies in die **tblGames**-tabel wat sedert die jaar 2019 vrygestel is.

Voorbeeld van afvoer van die eerste vyf rekords:

GameTitle	Genre	DateReleased
Call of Duty: Warzone	Battle Royale	2020/03/10
FIFA 21	Sports	2020/10/09
Assassin's Creed Valhalla	Action-Adventure	2020/11/10
Cyberpunk 2077	RPG	2020/12/10
Apex Legends	Battle Royale	2019/02/04

(4)

**2.1.3 Knoppie [2.1.3 - Genre]**

Die gebruiker moet die genre van 'n speletjie waarvoor daar gesoek word, insleutel deur 'n toevoerdialoogblokkie ('input dialogue box') te gebruik.

Kode is voorsien om die genre wat die gebruiker ingesleutel het, te onttrek en in 'n veranderlike met die naam **sGenre** te stoor.

Vertoon die **GameTitle** en **DateReleased** van al die speletjies wat ooreenstem met die genre wat deur die gebruiker ingesleutel is, gesorteer volgens die datum wat dit vrygestel is, van die nuutste tot die oudste speletjie.

Voorbeeld van afvoer as die genre 'Sports' ingesleutel is:

GameTitle	DateReleased
FIFA 21	2020/10/09
Rocket League	2015/07/07

(4)

#### 2.1.4 Knoppie [2.1.4 - Average Income per year]

Bepaal en vertoon die gemiddelde inkomste per jaar, met inagneming van die jaar waarin die maatskappy gestig is.

**WENK:** Deel die totale inkomste van al die speletjies wat deur die maatskappy ontwikkel is, deur die aantal jare sedert die maatskappy gestig is.

**LET WEL:** Die waardes in die **Income**-veld in die **tblGames**-tabel is in miljoene. Voorbeeld: Die inkomste uit die speletjie 'Among Us' word as 500 voorgestel, wat R500 000 000 beteken.

Vertoon die naam van die maatskappy en die gemiddelde inkomste per jaar in miljoene. Die berekende gemiddelde inkomste per jaar moet in geldeenheidsformaat ('currency') in 'n nuwe veld met die naam **AverageIncomePerYear** vertoon word.

Voorbeeld van afvoer van die eerste vier maatskappye:

CompanyName	AverageIncomePerYear
Behavior Interactive	R15 625 000.00
Bethesda Game Studios	R86 956 521.74
BioWare	R20 689 655.17
Blizzard Entertainment	R212 121 212.12

(8)

#### 2.1.5 Knoppie [2.1.5 - Remove game]

Skryf kode om die speletjie met die naam **Apex Legends** uit die **tblGames**-tabel te verwyder.

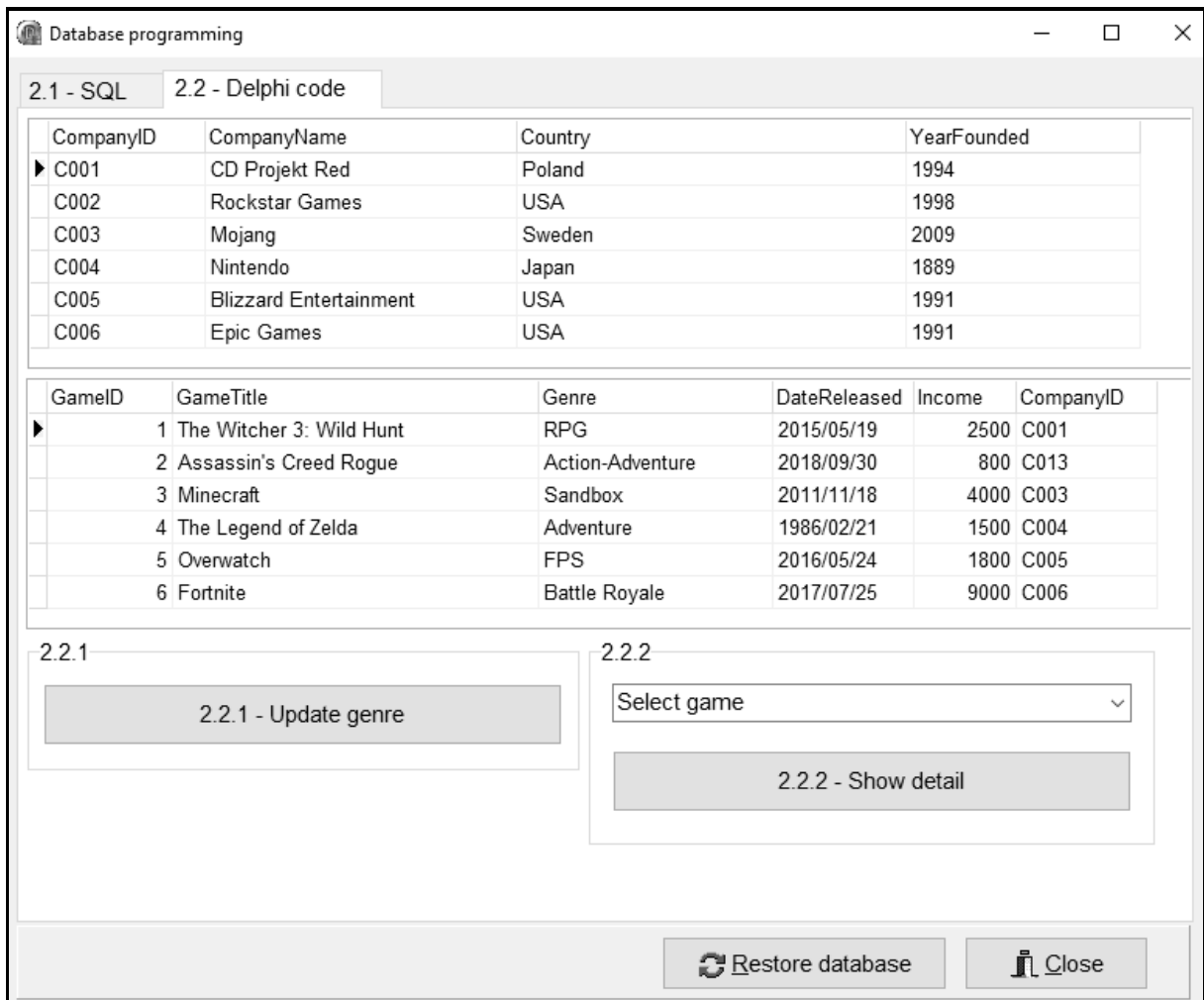
**WENK:** Kliek op **Knoppie 2.1.2** om te sien of die rekord verwyder is of nie.

**LET WEL:** Herstel ('Restore') die databasis om die oorspronklike data te kan gebruik as jy jou kode toets.

(2)

## 2.2 Oortjieblad ('Tabs heet') [2.2 - Delphi code]

Voorbeeld van die grafiese gebruikerskoppelvlak (GGK ('GUI')) vir VRAAG 2.2:



### LET WEL:

- Gebruik SLEGS Delphi-programmeringskode om VRAAG 2.2.1 en VRAAG 2.2.2 te beantwoord.
- GEEN punte sal vir SQL-stellings in VRAAG 2.2 toegeken word nie.

### 2.2.1 Knoppie [2.2.1 - Update genre]

Daar is verskeie *Tetris*-speletjies beskikbaar, soos *Tetris 1*, *Tetris 2*, *Tetris 99* ens. Die **Genre** van al die *Tetris*-speletjies is verkeerd gestoor en sal verander moet word.

Skryf kode om die genre van al die speletjies wat 'Tetris' as deel van die **GameTitle**-veld bevat, na 'Puzzle' te verander.

(7)

### 2.2.2 Knoppie [2.2.2 - Show detail]

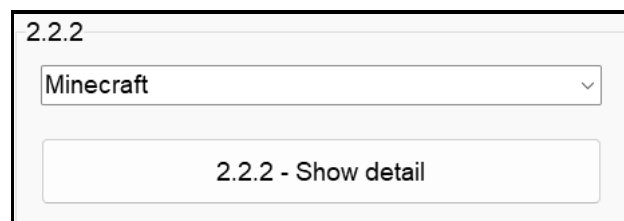
Die gebruiker moet 'n titel van 'n speletjie uit die kombinasieblokkie ('combo box') **cmbQ2\_2\_2** selekteer.

Kode is voorsien om die titel van die speletjie wat deur die gebruiker geselekteer is, te onttrek en die titel in 'n veranderlike met die naam **sGame** te stoor.

Vertoon die titel van die geselekteerde speletjie, die genre, die maatskappy wat die speletjie ontwikkel het en land waarvandaan die maatskappy kom in 'n **ShowMessage**-dialoogblokkie ('dialogue box').

Die **GameTitle** en **Genre** van die speletjie moet in die eerste reël, en die **CompanyName** en **Country** in die tweede reël in die ShowMessage-dialoogblokkie vertoon word, soos in die voorbeeld hieronder getoon word.

Voorbeeld van toevoer:



A screenshot of a form titled "2.2.2". It features a dropdown menu with "Minecraft" selected. Below the dropdown is a button labeled "2.2.2 - Show detail".

Voorbeeld van afvoer:



A screenshot of a dialog box titled "Question2\_p". It displays the text "Minecraft Sandbox" and "Developed by Mojang, Sweden". There is an "OK" button at the bottom right.

(12)

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

**TOTAAL AFDELING B: 40**

**AFDELING C****VRAAG 3: OBJEK-GEORIËNTEERDE PROGRAMMERING**

Persone wat rekenaarspeletjies speel ('gamers') gebruik 'n toepassing wat hulle voorsien van inligting oor eksklusiewe aanlyn/aflyn speletjies wat slegs vir 'n enkele platform beskikbaar is.

Doen die volgende:

- Maak die onvolledige program in die **Question 3**-lêergids oop.
- Maak die onvolledige objekklas **Game\_U.pas** oop.
- Sleutel jou eksamennummer as kommentaar in die eerste reël van die **Question3\_U.pas**-lêer en die **Game\_U.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans beperkte funksionaliteit.

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

The screenshot shows a window titled "Object-oriented programming" with a standard Windows title bar. The main content area has a black header with the text "Exclusive Online/Offline Games" in white. Below the header, there are four distinct panels, each with a title and a button:

- 3.2.1**: Contains a text input field with "Elden Ring", a checked "Online:" checkbox, a "Number of downloads:" spinner set to "0", a "Platform:" dropdown menu showing "Xbox", and a button labeled "3.2.1 - Instantiate object".
- 3.2.2**: Contains an unchecked "Internet:" checkbox, a "Platform:" dropdown menu showing "Xbox", and a button labeled "3.2.2 - Game compatibility".
- 3.2.3**: Contains a button labeled "3.2.3 - Update downloads".
- 3.2.4**: Contains a button labeled "3.2.4 - Popularity".

- Voltooi die kode soos gespesifiseer in VRAAG 3.1 en VRAAG 3.2.

**LET WEL:** Jy mag NIE enige addisionele attribute of gebruikergedefinieerde metodes byvoeg NIE, tensy die opdrag uitdruklik in die vraag genoem word.

- 3.1 Die onvolledige objekklas (**TGame**) wat voorsien is, bevat die verklaring van vier attribute wat 'n **Game**-objek beskryf.

Die UML-klasdiagram vir die **TGame**-klas word hieronder gegee.

<b>TGame-objek</b>	
- fName: String	// Naam van speletjie
- fOnline: Boolean	// True as internettoegang vereis word om te speel en false indien nie
- fDownloadCount: Integer	// Getal kere wat die speletjie afgelaai is
- fPlatform: String	// Op watter platform die speletjie uitgevoer word (Xbox/PlayStation/PC)
+ constructor create(sName: String, bOnline: Boolean, iDownloadCount: Integer, sPlatform: String)	
+ onlineStatus: String	
+ updateDownloadCount(iNumberDownloads: Integer)	
+ determinePopularity: String	
+ getName: String (is voorsien)	
+ getOnline: Boolean (is voorsien)	
+ getPlatform: String (is voorsien)	
+ toString : String (is voorsien - onvolledig)	

Voltooi die kode in die objekklas, soos wat in VRAAG 3.1.1 tot VRAAG 3.1.5 beskryf word.

Gebruik die UML-diagram wat voorsien is om VRAAG 3.1.1 tot VRAAG 3.1.5 te beantwoord.

- 3.1.1 Skryf kode vir die **constructor create**-metode soos gedefinieer in die klasdiagram wat voorsien is. Die attribute moet geïnisialiseer word deur die parameterwaardes wat ontvang is, te gebruik. (5)
- 3.1.2 Skryf kode vir die **onlineStatus**-metode wat 'n String-waarde 'Yes' of 'No' sal terugstuur om aan te dui of internettoegang vir die speletjie nodig is of nie. (4)
- 3.1.3 Skryf kode vir die **updateDownloadCount**-metode, wat die waarde van die **fDownloadCount**-attribuut sal vergroot deur die waarde wat as 'n parameter ontvang word, te gebruik. (3)

- 3.1.4 Skryf kode vir 'n metode met die naam **determinePopularity** wat die inligting hieronder gebruik om die gewildheid van 'n speletjie te bepaal, slegs as dit 'n **aanlyn** speletjie is. Stuur die resultaat ('popularity') as 'n String-waarde terug, op die inligting in die tabel hieronder gebaseer. Stuur die string 'Not an online game' terug as die speletjie nie 'n aanlyn speletjie is nie.

Die gewildheid van 'n **aanlyn** speletjie word bepaal deur die aflaai-telling, bv. 'n speletjie met 500 000 of meer aflaai-tellings word as 'Very popular' beskou.

Aflaai-telling ( <b>'Download count'</b> )	Gewildheid ( <b>'Popularity'</b> )
Minder as 100 000	'Not popular'
Groter as of gelyk aan 100 000 en kleiner as 500 000	'Popular'
Groter as of gelyk aan 500 000	'Very popular'

(10)

- 3.1.5 'n Onvoltooide **toString**-metode is voorsien. Voltooi die kode vir die **toString**-metode om aan te dui of internettoegang vir die speletjie vereis word of nie ('Yes'/'No').

Formaat van die **toString**-metode:

Name: <Naam van die speletjie>  
 Internet required: <Yes/No>  
 Number of downloads: <Aflaai-telling>  
 Platform: <Platform>

(2)

- 3.2 'n Onvolledige program is in die **Question 3**- lêergids ('folder') voorsien. Die program bevat kode vir die objekklas om toeganklik te wees en verklaar 'n objekveranderlike met die naam **objGame**.

Skryf kode om die take wat in VRAAG 3.2.1 tot VRAAG 3.2.4 beskryf word, uit te voer.

### 3.2.1 **Knoppie [3.2.1 - Instantiate object]**

Kode is voorsien om die naam, aanlyn status, getal aflaai ('downloads') en die platform van 'n speletjie uit die komponente af te laai en in veranderlikes te stoor.

Skryf kode om die volgende te doen:

- Gebruik die veranderlikes **sName**, **bOnline**, **iNumberOfDownloads** en **sPlatform** om 'n nuwe **objGame**-objek te instansieer.
- Roep die **toString**-metode om die inligting van die **Game**-objek in die 'rich edit' **redQ3** te vertoon.

Voorbeeld van toevoer:

3.2.1

Name: Elden Ring

Online:

Number of downloads: 650000

Platform: Xbox

3.2.1 - Instantiate object

Voorbeeld van afvoer:

Name: Elden Ring  
Internet required: Yes  
Number of downloads: 650000  
Platform: Xbox

(5)

### 3.2.2 Knoppie [3.2.2 - Game compatibility]

Die merkblokkie ('check box') **cbxQ3\_2\_2** moet afgemerkt word as die gebruiker toegang tot die internet het, en die platform wat gebruik word, moet uit die kombinasieblokkie ('combo box') **cmbQ3\_2\_2** geselekteer word.

Kode is voorsien om die waardes wat deur die gebruiker geselekteer is, te onttrek en in veranderlikes **bOnline** en **sPlatform** te stoor.

Skryf kode om die volgende te doen:

- Roep die **getOnline**- en **getPlatform**-metodes om na te gaan of die gebruiker die speletjie sal kan speel of nie.
- Gebruik 'n boodskapdialoogblokkie ('message dialogue box') om 'n geskikte boodskap, wat aandui of die gebruiker die speletjie kan speel of nie, te vertoon.

Voorbeeld van toevoer:

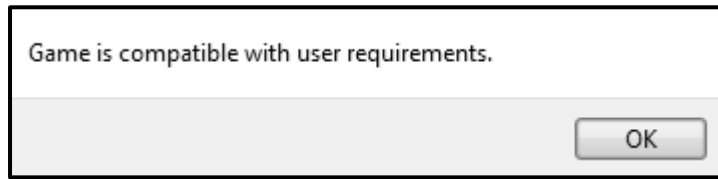
3.2.2

Internet:

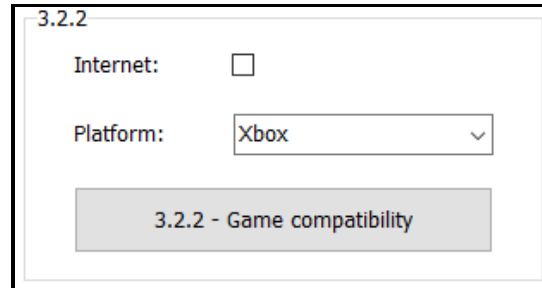
Platform: Xbox

3.2.2 - Game compatibility

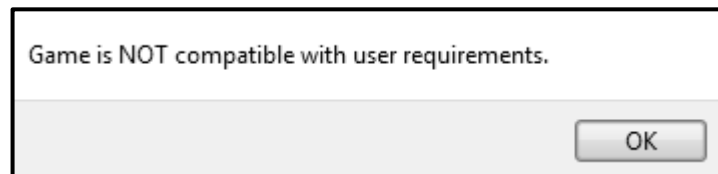
Voorbeeld van afvoer:



Voorbeeld van toevoer:



Voorbeeld van afvoer:



(4)

### 3.2.3 Knoppie [3.2.3 - Update downloads]

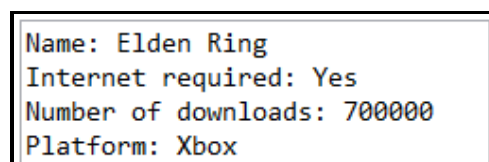
Wanneer 'n speletjie afgelaai word, moet die relevante attribuut van die **Game**-objek ooreenkomstig opgedateer word.

Die gebruiker moet die getal nuwe aflaai ('downloads') van die speletjie insleutel deur 'n toevoerdialoogblokkie ('input dialogue box') te gebruik.

Skryf kode om die volgende te doen:

- Roep die **updateDownloadCount**-metode deur die waarde wat uit die toevoerdialoogblokkie onttrek is, as 'n argument te gebruik.
- Roep die **toString**-metode om die opgedateerde inligting van die **Game**-objek in die 'rich edit' **redQ3** te vertoon.

Voorbeeld van afvoer vir die speletjie *Elden Ring* wat 650 000 aflaai gehad het, en nog 50 000 aflaai is bygetel:



(5)

**3.2.4 Knoppie [3.2.4 - Popularity]**

Skryf kode om die relevante metodes te roep om die naam en vlak van gewildheid van die speletjie in die 'rich edit' **redQ3** te vertoon.

Voorbeeld van afvoer vir die speletjie *Elden Ring* wat 700 000 aflaai het:

```
Elden Ring: Very popular
```

Voorbeeld van afvoer vir die speletjie *Orange Box* wat 50 000 aflaai het:

```
Orange Box: Not popular
```

(2)

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

**TOTAAL AFDELING C: 40**

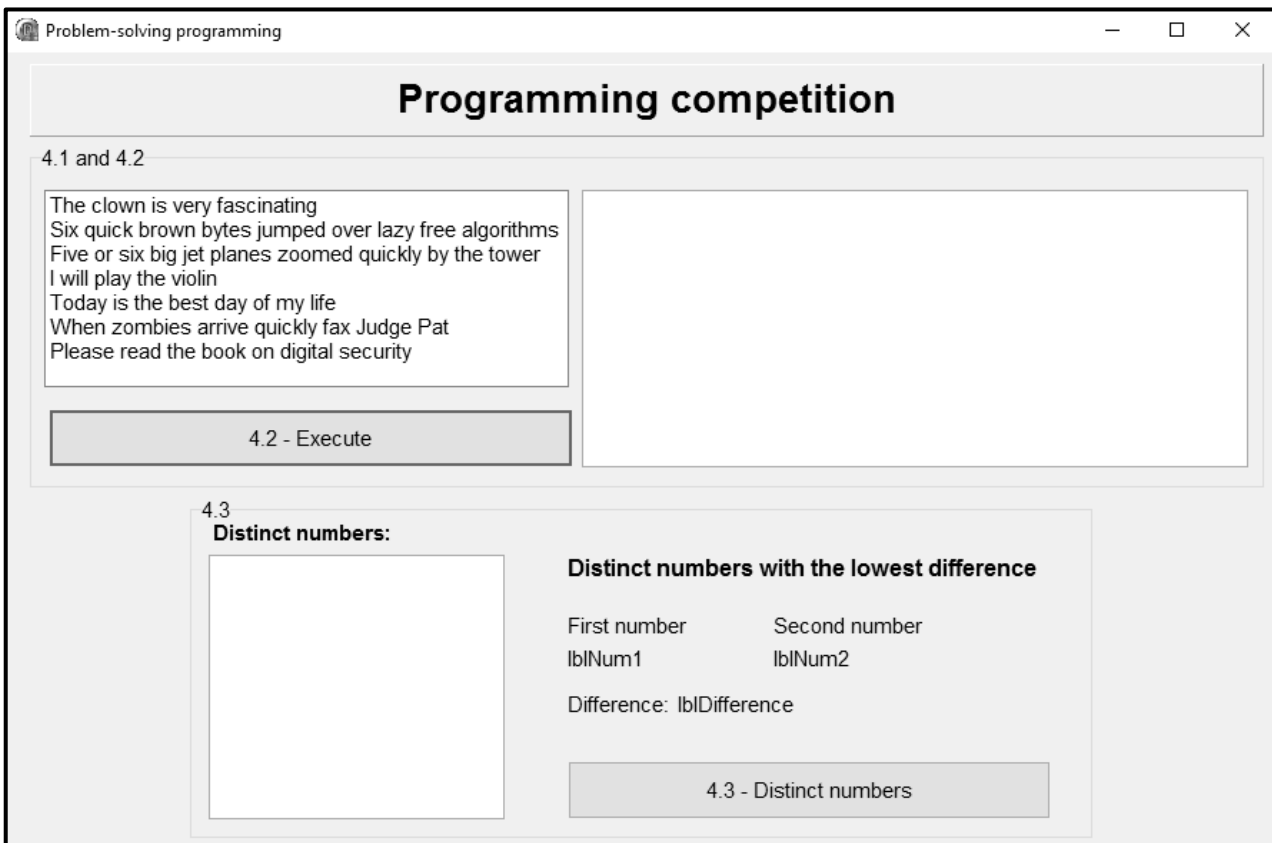
**AFDELING D****VRAAG 4: PROBLEEMOPLOSSINGPROGRAMMERING**

Doen die volgende:

- Maak die onvolledige program in die **Question 4**-lêergids oop.
- Sleutel jou eksamenommer as kommentaar in die eerste reël van die **Question4\_U.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie.

Voltooi die kode vir elke afdeling van VRAAG 4, soos wat in VRAAG 4.1, VRAAG 4.2 en VRAAG 4.3 beskryf word.

Voorbeeld van die grafiese gebruikerskoppelvlak (GGK ('GUI')) vir VRAAG 4:



'n Lysblokkie ('List box'), **IstQ4\_2**, wat verskeie sinne bevat, is in die program voorsien:

Voorbeeld van die sinne in lysblokkie **IstQ4\_2**:

- The clown is very fascinating
- Six quick brown bytes jumped over lazy free algorithms
- Five or six big jet planes zoomed quickly by the tower

#### 4.1 Funksie isPangram

'n Pangram is 'n string wat al die letters van die alfabet ten minste een keer bevat.

'n Voorbeeld van 'n pangram: The quick brown fox jumped over the lazy dog.

Skep 'n funksie met die naam **isPangram** wat 'n sin as 'n parameter sal ontvang en bepaal of die sin 'n pangram is of nie. Stuur die waarde TRUE terug as die sin 'n pangram is, of andersins, stuur die waarde FALSE terug.

(8)

#### 4.2 Knoppie [4.2 - Execute]

Elk van die sinne wat in die lysblokkie **IstQ4\_2** voorsien is, moet ge lees en geëvalueer word om vas te stel of elke sin 'n pangram is of nie.

Skryf kode om die sinne en die uitslae van die pangramme in die 'rich edit' **redQ4\_2** te vertoon, soos in die voorbeeld hieronder getoon.

**LET WEL:** Jou kode moet vir enige stel sinne werk, nie slegs vir die sinne wat in die lysblokkie voorsien is nie.

Voorbeeld van afvoer:

Original sentence	Pangram
The clown is very fascinating	No
Six quick brown bytes jumped over lazy free algorithms	Yes
Five or six big jet planes zoomed quickly by the tower	Yes
I will play the violin	No
Today is the best day of my life	No
When zombies arrive quickly fax Judge Pat	Yes
Please read the book on digital security	No

(8)

#### 4.3 Knoppie [4.3 - Distinct numbers]

Jy is van die volgende globale skikking ('array') voorsien:

```
arrNumbers: array [1 .. 20] of Real = (10.39, 5.33, 5.48,
    9.53, 5.82, 4.21, 5.33, 2.26, 4.21, 8.48, 4.82,
    9.53, 2.17, 5.33, 9.53, 8.46, 9.53, 10.26, 5.33,
    9.21);
```

Die skikking **arrNumbers** bevat twintig reële getalle.

Sommige van die getalle in die skikking verskyn meer as een keer. 'n Onderskeidings-/Unieke ('distinct/unique') element in 'n skikking is 'n element wat nie herhaal word nie.

Skryf kode om die volgende te doen:

- Identifiseer die onderskeidings-/unieke getalle in die **arrNumbers**-skikking en vertoon hierdie getalle in die 'rich edit' **redQ4\_3**, soos in die voorbeeld van afvoer hieronder getoon.
- Bepaal watter twee onderskeidings-/unieke getalle het die kleinste verskil tussen hulle, en vertoon die twee getalle in die etikette ('labels') **lbINum1** en **lbINum2** en die verskil tussen hulle in die etiket **lbDifference**, tot TWEE desimale plekke geformatteer.

Voorbeeld van afvoer:

4.3  
**Distinct numbers:**

10.39  
5.48  
5.82  
2.26  
8.48  
4.82  
2.17  
8.46  
10.26  
9.21

**Distinct numbers with the lowest difference**

First number      Second number  
8.48                8.46

Difference: 0.02

4.3 - Distinct numbers

(14)

- Sleutel jou eksamennummer as kommentaar in die eerste reël van die programlêer in.
- Stoor jou program.
- Druk die kode indien dit vereis word.

**TOTAAL AFDELING D: 30**  
**GROOTTOTAAL: 150**

**INLIGTINGSTEGNOLOGIE V1****DATABASIS-INLIGTING VIR VRAAG 2:**

Die ontwerp van die databasistabelle soos volg:

Tabel: **tblCompanies**

Hierdie tabel bevat die besonderhede van elke maatskappy:

Veldnaam	Datatype	Beskrywing
CompanyID	Text(10)	Unieke ID van die maatskappy
CompanyName	Text(30)	Naam van die maatskappy
Country	Text(15)	Land waarvandaan die maatskappy kom
YearFounded	Number	Jaar waarin die maatskappy gestig is

Voorbeeld van die eerste tien rekords in die **tblCompanies**-tabel:

CompanyID ▾	CompanyName ▾	Country ▾	YearFounded ▾
C001	CD Projekt Red	Poland	1994
C002	Rockstar Games	USA	1998
C003	Mojang	Sweden	2009
C004	Nintendo	Japan	1889
C005	Blizzard Entertainment	USA	1991
C006	Epic Games	USA	1991
C007	InnerSloth	USA	2015
C008	Infinity Ward	USA	2002
C009	Riot Games	USA	2006
C010	Valve Corporation	USA	1996

Tabel: **tblGames**

Hierdie tabel bevat inligting oor die speletjies wat deur elke maatskappy ontwikkel is:

Veldnaam	Datatype	Beskrywing
GameID	Number	Unieke ID vir die speletjie
GameTitle	Text(40)	Titel van die speletjie
Genre	Text(20)	Genre van die speletjie
DateReleased	DateTime	Datum waarop die speletjie vrygestel is
Income	Number	Totale inkomste sedert die vrystellingsdatum in miljoene
CompanyID	Text(10)	Die ID van die maatskappy wat die speletjie ontwikkel het

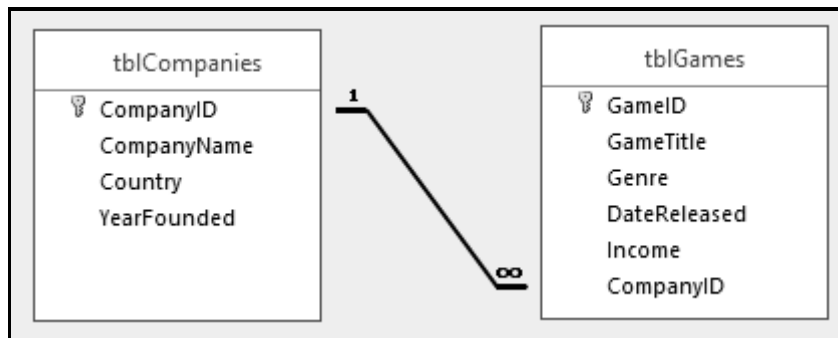
Voorbeeld van die eerste tien rekords van die **tblGames**-tabel:

GameID	GameTitle	Genre	DateReleased	Income	CompanyID
1	The Witcher 3: Wild Hunt	RPG	2015/05/19	2500	C001
2	Assassin's Creed Rogue	Action-Adventure	2018/09/30	800	C013
3	Minecraft	Sandbox	2011/11/18	4000	C003
4	The Legend of Zelda	Adventure	1986/02/21	1500	C004
5	Overwatch	FPS	2016/05/24	1800	C005
6	Fortnite	Battle Royale	2017/07/25	9000	C006
7	Red Dead Redemption 2	Action-Adventure	2018/10/26	3500	C002
8	Among Us	Social Deduction	2018/06/15	500	C007
9	Call of Duty: Warzone	Battle Royale	2020/03/10	2000	C008
10	League of Legends	MOBA	2009/10/27	3000	C009

**LET WEL:**

- Konneksiekode is voorsien.
- Die databasis word met 'n wagwoord beskerm, dus sal jy nie direkte toegang tot die databasis kan verkry nie.

Die volgende een-tot-baie-verwantskap met referensiële integriteit ('referential integrity') bestaan tussen die twee tabelle in die databasis:



**BEPLANNINGSBLADSY 1**

**BEPLANNINGSBLADSY 2**

Eksamenplakker

**150****INLIGTINGSTEGNOLOGIE V1 – MEI/JUNIE 2025****INLIGTINGSBLAD** (moet NÁ die 3 uur-eksamensessie deur die kandidaat ingevul word)

SENTRUMNOMMER: \_\_\_\_\_

EKSAMENNOMMER: \_\_\_\_\_

NOMMER VAN WERKSTASIE: \_\_\_\_\_

**Weergawe van Delphi wat tydens die INLIGTINGSTEGNOLOGIE SS/NSS MEI/JUNIE 2025-eksamen gebruik is:**

Merk die relevante blokkie met 'n kruis (X)	Delphi 10	Delphi XE	Delphi 10.3	Delphi Community	Delphi 11	Delphi 12	Ander: (Spesifiseer) _____
---	-----------	-----------	-------------	------------------	-----------	-----------	-------------------------------

NAAM VAN LÊERGIDS ('FOLDER') \_\_\_\_\_

*Kandidaat moet afmerk as die lêernaam (lêername) wat vir elke antwoord gebruik is, gestoor is en/of 'n poging aangewend is.*

Vraag-nommer	Lêernaam	Gestoor (✓)	Gepoog (✓)	Maksimum Punt	Punt Toegeken	Nasiener se Kode
1	Question1_P.dproj			<b>40</b>		
2	Question2_P.dproj			<b>40</b>		
3	MGame_U.pas			<b>24</b>		
	Question3_P.dproj			<b>16</b>		
4	Question4_P.dproj			<b>30</b>		
<b>TOTAAL</b>				<b>150</b>		

Kommentaar: (slegs vir kantoor/nasiener se gebruik)

---



---



# basic education

---

Department:  
Basic Education  
**REPUBLIC OF SOUTH AFRICA**

## **SENIOR CERTIFICATE EXAMINATIONS/ NATIONAL SENIOR CERTIFICATE EXAMINATIONS**

**INFORMATION TECHNOLOGY P1**

**MAY/JUNE 2025**

**MARKING GUIDELINES**

**MARKS: 150**

**These marking guidelines consist of 27 pages.**

**GENERAL INFORMATION:**

- These marking guidelines are to be used as the basis for the marking session. They were prepared for use by markers. All markers are required to attend a rigorous standardisation meeting to ensure that the guidelines are consistently interpreted and applied in the marking of candidates' work.
- Note that learners who provide an alternate correct solution to that given as example of a solution in the marking guidelines will be given full credit for the relevant solution, unless the specific instructions in the paper was not followed or the requirements of the question was not met
- **Annexures A, B, C and D** (pages 3 to 12) include the marking grid for each question.
- **Annexures E, F, G and H** (pages 13 to 27) contain examples of solutions for QUESTIONS 1 to 4 in programming code.
- Copies of **Annexures A, B, C, D** and **the summary for the marks of the learner** (pages 3 to 11) should be made for each learner and completed during the marking session.

## ANNEXURE A

## QUESTION 1: MARKING GRID – GENERAL PROGRAMMING SKILLS

CENTRE NUMBER:		EXAMINATION NUMBER:	
QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
1.1	<p><b>Button [1.1 - Display shape]</b></p> <p>Check if ✓ <code>itemIndex = 0</code> / item is circle ✓  Change shape property of <code>shpQ1_1</code> ✓ to <code>stCircle</code>  Else ✓  Change shape property of <code>shpQ1_1</code> to <code>stRectangle</code> ✓</p> <p><b>Also accept:</b></p> <ul style="list-style-type: none"> <li>• Case can be used instead of if statement</li> </ul> <p><b>Concepts:</b></p> <ul style="list-style-type: none"> <li>• Test using if / case (1)</li> <li>• Using <code>itemIndex</code> / item (1)</li> <li>• Changing the shape property (1)</li> <li>• Else (1)</li> <li>• Correctly assigning both shapes (1)</li> </ul>	5	
1.2	<p><b>Button [1.2 - Display colour]</b></p> <p>Correct use of the colours selected from the combo boxes ✓</p> <p>Check if the colours selected in either of the combo boxes is red and blue:</p> <p>if <code>((sColor1 = 'Red') ✓ AND (sColor2 = 'Blue')) ✓</code>  OR ✓ <code>((sColor1 = 'Blue') AND (sColor2 = 'Red')) ✓</code> then  Set the colour of <code>pnlQ1_2</code> to <code>clPurple</code> ✓  Else ✓  Set the caption of <code>pnlQ1_2</code> to 'No colour' ✓</p> <p><b>Concepts:</b></p> <ul style="list-style-type: none"> <li>• Extract colours selected (1)</li> <li>• Use of if-statement / case statement to test for the first colour (1)</li> <li>• Test for second colour in combination (1)</li> <li>• Correct use of AND (1), correct use of OR in both combinations (1)</li> <li>• Set colour to <code>clPurple</code> (1)</li> <li>• Else (1)</li> <li>• Set caption to No colour (1)</li> </ul>	8	

1.3	<p><b>Button [1.3 - Average]</b></p> <p>Initialise variables (sum and counter)  Test if the text file does not exist (Try/FileExists) ✓      Display error message and exit program ✓  AssignFile (tFile, 'DataQ1_3.txt') ✓  Reset file ✓  Loop through the text file ✓      Read value from file into variable ✓      Add value to sum ✓      Increase counter ✓  Calculate the average ✓ and display formatted to two decimal places in memQ1_3 ✓</p>	10	
1.4	<p><b>Button [1.4 - Display pattern]</b></p> <p>Extract the number of rows selected from spnQ1_4 ✓  Outer loop from 1 to the number of rows ✓      Initialise string ✓      Nested ✓ inner loop from 1 to outer loop ✓          Add outer loop counter value ✓ to string ✓      // End of inner loop      Display output string in redQ1_4 ✓  // End of outer loop</p>	8	
1.5	<p><b>Button [1.5 - Sum of ASCII values]</b></p> <p>Extract word from list box ✓  Initialise sum ✓  Loop ✓ from 1 to length of word ✓      Test if character ✓ is uppercase alphabet character ✓      Add the ASCII value of the character ✓ to sum ✓  Display sum ✓</p>	9	
<b>TOTAL SECTION A:</b>		<b>40</b>	

## ANNEXURE B

## QUESTION 2: MARKING GRID – DATABASE PROGRAMMING

CENTRE NUMBER:		EXAMINATION NUMBER:	
QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
2.1	<b>SQL statements</b>		
2.1.1	<b>Button [2.1.1 - USA companies]</b> SELECT * ✓ FROM tblCompanies ✓ WHERE Country = "USA" ✓	3	
2.1.2	<b>Button [2.1.2 - Year released]</b> SELECT GameTitle, Genre, DateReleased ✓ FROM tblGames ✓ WHERE YEAR ✓ (DateReleased) >= 2019 ✓  Also accept: Left (DateReleased,4) >= 2019 Mid (DateReleased,1,4) >= 2019 Year (DateReleased) > 2018 DateReleased >= #2019/01/01#	4	
2.1.3	<b>Button [2.1.3 - Genre]</b> SELECT GameTitle, DateReleased FROM tblGames ✓ WHERE Genre = ✓ "' + sGenre + '" ✓ ORDER BY DateReleased DESC ✓  Also accept: WHERE Genre = ' + QuotedStr(sGenre) + '	4	
2.1.4	<b>Button [2.1.4 – Average income per year]</b> SELECT CompanyName, Format((Sum(Income)*1000000) ✓/ (Year(Date()) ✓ - YearFounded - 1) ✓, "Currency" ✓) AS AverageIncomePerYear ✓ FROM tblCompanies, tblGames ✓ WHERE tblCompanies.CompanyID = tblGames.CompanyID ✓ GROUP BY CompanyName, YearFounded ✓  Also accept: Year(Now)	8	
2.1.5	<b>Button [2.1.5 - Remove game]</b> DELETE FROM tblGames ✓ WHERE GameTitle = "Apex Legends" ✓	2	
	<b>Subtotal:</b>	<b>21</b>	

**QUESTION 2: MARKING GRID (CONT.)**

2.2	<b>Database Manipulation</b>		
2.2.1	<p><b>Button [2.2.1 - Update genre]</b></p> <p>Go to the first record in tblGames ✓  Loop through tblGames ✓  Test if Tetris ✓ is part of the field tblGames['GameTitle'] ✓  tblGames.Edit; ✓  tblGames['Genre'] := 'Puzzle' ✓  tblGames.Post;  tblGames.Next; ✓</p>	7	
2.2.2	<p><b>Button [2.2.2 - Show detail]</b></p> <p>Go to the first record in tblGames ✓  Loop through tblGames ✓</p> <p>Test if tblGames['GameTitle'] = sGame ✓  Go to the first record in tblCompanies ✓</p> <p>Loop through tblCompanies ✓  Test if (tblGames ['CompanyID'] = ✓  tblCompanies ['CompanyID']) ✓</p> <p>Use a ShowMessage dialog box ✓ to display:  tblGames['GameTitle'] + ' ' + tblGames['Genre'] ✓  + #13 + 'Developed by ' +  tblCompanies ['CompanyName'] + ', ' +  tblCompanies ['Country'] ✓</p> <p>tblCompanies.Next ✓  End loop (tblCompanies)</p> <p>tblGames.Next ✓  End loop (tblGames)</p>	12	
	<b>Subtotal:</b>	<b>19</b>	
	<b>TOTAL SECTION B:</b>	<b>40</b>	

## ANNEXURE C

## QUESTION 3: MARKING GRID – OBJECT-ORIENTED PROGRAMMING

CENTRE NUMBER:		EXAMINATION NUMBER:	
QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
3.1.1	<b>Constructor create method:</b>  Heading with four parameter values (String, Boolean, Integer, String) ✓  Set <code>fName</code> to String parameter ✓ Set <code>fOnline</code> to Boolean parameter ✓ Set <code>fDownloadCount</code> to integer parameter ✓ Set <code>fPlatform</code> to String parameter ✓	5	
3.1.2	<b>onlineStatus function:</b>  function heading with String return type ✓  Check if <code>fOnline</code> ✓ = True Result = 'Yes' ✓ Else Result = 'No' ✓	4	
3.1.3	<b>updateDownloadCount procedure:</b>  procedure heading with integer parameter ✓  <code>fDownloadCount = fDownloadCount</code> ✓ + parameter ✓	3	
3.1.4	<b>determinePopularity function:</b>  function heading with String return type ✓  Check if <code>fOnline</code> ✓ = True if <code>fDownloadCount</code> < 100000 ✓ Result := 'Not popular' ✓ Else ✓ if <code>fDownloadCount</code> < 500000 ✓ Result := 'Popular' ✓ Else ✓ Result := 'Very popular' ✓ Else Result := 'Not an online game'; ✓	10	

QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
3.1.5	<b>toString function</b>  Result := 'Name: ' + fName + #13 + 'Internet required: ' + <b>onlineStatus</b> + #13 + 'Number of downloads: ' + IntToStr(fDownloadCount) + #13 + 'Platform: ' + fPlatform;  Calling onlineStatus method ✓ at the correct position. ✓	2	
	<b>Subtotal: Object class</b>	<b>24</b>	

**QUESTION 3: MARKING GRID (CONT.)**

QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
3.2.1	<b>Button [3.2.1 - Instantiate object]</b>  objGame := ✓ TGame.create ✓ (sName, bOnline, iNumberOfDownloads, sPlatform) ✓  Display objGame information in redQ3 ✓ using toString method. ✓	<b>5</b>	
3.2.2	<b>Button [3.2.2 - Game compatibility]</b>  Check if (bOnline = objGame.getOnline) ✓ AND (sPlatform = objGame.getPlatform) ✓ ShowMessage('Game is compatible with user requirements.') ✓ Else ShowMessage('Game is NOT compatible with user requirements.');	<b>4</b>	
3.2.3	<b>Button [3.2.3 – Update downloads]</b>  Extract number entered using a dialog box ✓ and typecast to integer ✓ Call the updateDownloadCount method ✓ with the extracted number as argument ✓ Display objGame information in redQ3 using toString method. ✓	<b>5</b>	
3.2.4	<b>Button [3.2.4 - Popularity]</b>  Call the getName and the determinePopularity methods ✓ Display the name of the game as well as the popularity in redQ3 ✓ in the correct format	<b>2</b>	
	<b>Subtotal Form class:</b>	<b>16</b>	
	<b>TOTAL SECTION C:</b>	<b>40</b>	

## ANNEXURE D

## QUESTION 4: MARKING GRID – PROBLEM-SOLVING PROGRAMMING

CENTRE NUMBER:		EXAMINATION NUMBER:	
QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
4.1	<p><b>Function isPangram</b></p> <p>Function heading with string parameter ✓ with Boolean return type ✓</p> <p>Loop ✓ or mechanism to check each letter of the alphabet OR build frequency/count structure</p> <p>Mechanism to check presence of all 26 letters: (3)</p> <ul style="list-style-type: none"> <li>• Alphabet letters are individually tracked or counted ✓</li> <li>• A correct uppercase / lowercase check is applied to each letter ✓</li> <li>• To ensure that the letter appears in the string ✓</li> </ul> <p>Checking counter / Boolean ✓ Return TRUE if all alphabet letters are found Return FALSE if any alphabet letter is missing ✓</p>	8	
4.2	<p><b>Button [4.2 - Execute]</b></p> <p>Loop ✓ through sentences ✓ Extract the sentence from IstQ4_2 ✓ If statement – call isPangram ✓ with argument ✓ Display sentence with 'Yes' ✓ separated by #9 ✓ in redQ4_2 Else Display sentence with 'No' separated by #9 in redQ4_2 ✓</p>	8	

<p>4.3</p>	<p><b>Button [4.3 - Distinct numbers]</b></p> <p><b>Concepts:</b></p> <p>Create String / temporary array with unique numbers:                  Outer loop ✓ Inner loop ✓                  Determine if number is unique ✓✓✓                  Add unique number to string / temporary array ✓✓                  Display unique number ✓</p> <p>Determine lowest difference:                  Nested looping ✓                  Calculate difference as a positive number ✓                  Determine if current difference is the lowest ✓                  Store two numbers and the lowest difference ✓✓</p> <p>Display numbers with lowest difference in labels and the lowest difference correctly formatted ✓</p> <p><b>Possible solution:</b></p> <pre>                 Initialise index of unique array to 0                 Loop outer from 1 to length of arrNumbers                 Initialise counter to 0                 Nested loop inner from 1 to length of arrNumbers                     Check if arrNumbers[outer] = arrNumbers[inner]                     Increment counter                 //End inner loop                  //Check if duplicate elements not found                 Test if counter = 1                 Increment index of unique array                 arrUnique[index] = arrNumbers[outer]                 Display unique number/arrUnique[index]                 //End outer loop                  Initialise lowest difference variable                 Loop I from 1 to length of temp array - 1                 Nested loop J from I + 1 to length of temp array                 Current difference =                     Abs(arrTemp[I] - arrTemp[J])                 Check if (rCurrDiff &lt; rLowestDiff) AND                 (rCurrDiff &lt;&gt; 0)                     rNum1 := arrTemp[I];                     rNum2 := arrTemp[J];                     rLowestDiff:= rCurrDiff;             </pre> <p>Display numbers with lowest difference and the lowest difference on labels correctly formatted</p>	<p>14</p>	
	<p><b>TOTAL SECTION D:</b> <b>GRAND TOTAL:</b></p>	<p><b>30</b> <b>150</b></p>	

**SUMMARY OF LEARNER'S MARKS:**

<b>CENTER NUMBER:</b>		<b>LEARNER'S EXAMINATION NUMBER:</b>			
	<b>SECTION A</b>	<b>SECTION B</b>	<b>SECTION C</b>	<b>SECTION D</b>	
	<b>QUESTION 1</b>	<b>QUESTION 2</b>	<b>QUESTION 3</b>	<b>QUESTION 4</b>	<b>GRAND TOTAL</b>
<b>MAX. MARKS</b>	<b>40</b>	<b>40</b>	<b>40</b>	<b>30</b>	<b>150</b>
<b>LEARNER'S MARKS</b>					

**ANNEXURE E: SOLUTION FOR QUESTION 1**

```
unit Question1_U;

interface

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

type
  TfrmQuestion1 = class(TForm)
    grpQ1_1: TGroupBox;
    grpQ1_2: TGroupBox;
    grpQ1_3: TGroupBox;
    grpQ1_4: TGroupBox;
    grpQ1_5: TGroupBox;
    rgpQ1_1: TRadioGroup;
    shpQ1_1: TShape;
    btnQ1_1: TButton;
    Label1: TLabel;
    Label2: TLabel;
    cmbColour1: TComboBox;
    cmbColour2: TComboBox;
    btnQ1_2: TButton;
    pnlQ1_2: TPanel;
    Label3: TLabel;
    spnQ1_3: TSpinEdit;
    btnQ1_3: TButton;
    redQ1_3: TRichEdit;
    btnQ1_4: TButton;
    pnlQ1_4: TPanel;
    btnQ1_5: TButton;
    procedure btnQ1_1Click(Sender: TObject);
    procedure btnQ1_2Click(Sender: TObject);
    procedure btnQ1_3Click(Sender: TObject);
    procedure btnQ1_4Click(Sender: TObject);
    procedure btnQ1_5Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmQuestion1: TfrmQuestion1;

implementation

{$R *.dfm}
```

```
//=====
// Question 1.1                                     5 marks
// =====
```

```
procedure TfrmQuestion1.btnQ1_1Click(Sender: TObject);
begin
    // Question 1.1

    if rgpQ1_1.ItemIndex = 0 then
        shpQ1_1.Shape := stCircle
    else
        shpQ1_1.Shape := stRectangle;
end;
```

```
//=====
// Question 1.2                                     8 marks
// =====
```

```
procedure TfrmQuestion1.btnQ1_2Click(Sender: TObject);
var
    sColour1, sColour2: String;
begin
    // Provided code
    pnlQ1_2.Color := clBtnFace;
    pnlQ1_2.Caption := '';

    // Question 1.2
    sColour1 := cmbColour1.Text;
    sColour2 := cmbColour2.Text;

    if ((sColour1 = 'Red') or (sColour2 = 'Red')) AND
        ((sColour1 = 'Blue') or (sColour2 = 'Blue')) then
        pnlQ1_2.Color := clPurple
    else
        pnlQ1_2.Caption := 'No colour';

end;
```

```
//=====
// Question 1.3 10 marks
// =====
```

```
procedure TfrmQuestion1.btnQ1_3Click(Sender: TObject);
var
  tFile: textFile;
  iNum, iSum, iCnt: Integer;
  rAverage: real;
begin
  AssignFile(tFile, 'DataQ1_3.txt');
  try
    reset(tFile);
  except
    ShowMessage('File not found');
    Application.Terminate;
  end;
  iSum := 0;
  iCnt := 0;
  while not eof(tFile) do
  begin
    readln(tFile, iNum);
    iSum := iSum + iNum;
    Inc(iCnt);
  end;
  rAverage := iSum / iCnt;
  memQ1_3.Lines.Add('Average: ' + FloatToStrF(rAverage, ffFixed, 8, 2));
  CloseFile(tFile);
end;
```

```
//=====
// Question 1.4 8 marks
// =====
```

```
procedure TfrmQuestion1.btnQ1_4Click(Sender: TObject);
var
  iRows, I, J, K: Integer;
  sConcat: String;
begin
  // Provided code
  redQ1_4.Clear;

  // Question 1.4
  iRows := spnQ1_4.Value;

  for I := 1 to iRows do
  begin
    sConcat := '';
    for J := 1 to I do
      sConcat := sConcat + IntToStr(I);
    redQ1_4.Lines.Add(sConcat);
  end;
end;
```

```
//=====
// Question 1.5 9 marks
// =====

procedure TfrmQuestion1.btnQ1_5Click(Sender: TObject);
var
    sWord: String;
    iCnt, iTot: Integer;

begin
    iTot := 0;
    sWord := lstQ1_4.Items[lstQ1_4.ItemIndex];
    for iCnt := 1 to length(sWord) do
        begin
            if sWord[iCnt] IN ['A' .. 'Z'] then
                begin
                    iTot := iTot + Ord(sWord[iCnt])
                end;
            end;
        lblQ1_4.Caption := IntToStr(iTot);
    end;

procedure TfrmQuestion1.FormCreate(Sender: TObject);
begin
    shpQ1_1.Visible := False;
end;

end.
```

**ANNEXURE F: SOLUTION FOR QUESTION 2**

```
//=====
// Question 2.1 - Section: SQL statements
//=====

//=====
// Question 2.1.1                                     3 marks
//=====
    sSQL1 := 'SELECT * ' +
            'FROM tblCompanies ' +
            'WHERE Country = "USA"';

//=====
// Question 2.1.2                                     4 marks
//=====
    sSQL2 := 'SELECT GameTitle, Genre, DateReleased ' +
            'FROM tblGames ' +
            'WHERE YEAR (DateReleased) >= 2019';

//=====
// Question 2.1.3                                     4 marks
//=====
    sSQL3 := 'SELECT GameTitle, DateReleased ' +
            'FROM tblGames ' +
            'WHERE Genre = "' + sGenre + '" ' +
            'ORDER BY DateReleased DESC';

//=====
// Question 2.1.4                                     8 marks
//=====
    sSQL4 := 'SELECT CompanyName, format((sum(Income)*1000000)/
            (year(date()) - YearFounded - 1),"Currency") ' +
            'AS IncomePerYear ' +
            'FROM tblCompanies, tblGames ' +
            'WHERE tblCompanies.CompanyID = tblGames.CompanyID ' +
            'GROUP BY CompanyName, YearFounded';

//=====
// Question 2.1.5                                     2 marks
//=====
    sSQL5 := 'DELETE FROM tblGames ' +
            'WHERE GameTitle = "Apex Legends";
```

```
//=====
// Question 2.2 - Section Delphi code
//=====

//=====
// Question 2.2.1 7 marks
//=====
procedure TfrmQuestion2.btnQ2_2_1Click(Sender: TObject);
begin
  tblGames.First;
  while (NOT tblGames.Eof) do
  begin
    if pos('Tetris', tblGames['GameTitle']) > 0 then
    begin
      tblGames.Edit;
      tblGames['Genre'] := 'Puzzle';
      tblGames.Post;
    end;
    tblGames.Next;
  end;
end;

//=====
// Question 2.2.2 12 marks
//=====
procedure TfrmQuestion2.btnQ2_2_2Click(Sender: TObject);
var
  sGame,sDetail : String;
begin
  // Provided code
  sGame := cmbQ2_2_2.Text;

  // 2.2.2 - Show detail
  tblGames.First;
  while (NOT tblGames.Eof) do
  begin
    if (sGame = tblGames['GameTitle']) then
    begin
      tblCompanies.First;
      while (NOT tblCompanies.EOF) do
      begin
        if (tblGames['CompanyID'] =
            tblCompanies['CompanyID']) then
        begin
          ShowMessage(tblGames['GameTitle'] + ' ' +
            tblGames['Genre'] + #13 +
            'Developed by ' +
            tblCompanies['CompanyName'] + ' ' +
            tblCompanies['Country']);
        end;
        tblCompanies.Next;
      end;
    end;
    tblGames.Next;
  end;
end;
```

## SC/NSC – Marking Guidelines

```
// =====  
// {$ENDREGION}  
// =====  
// {$REGION 'Provided code: Setup DB connections - DO NOT CHANGE!'}  
// =====  
  
procedure TfrmQuestion2.bmbRestoreDBClick(Sender: TObject);  
begin  
    // Restores the Database  
    dbCONN.RestoreDatabase;  
    dbCONN.SetupGrids(dbgDevelopers, dbgGames, dbgrdSQL);  
end;  
  
procedure TfrmQuestion2.FormClose(Sender: TObject; var Action:  
TCloseAction);  
begin  
    // Disconnects from database and closes all open connections  
    dbCONN.dbDisconnect;  
end;  
  
procedure TfrmQuestion2.FormShow(Sender: TObject);  
begin  
    // Sets up the connection to database and opens the tables.  
    dbCONN := TConnection.Create;  
    dbCONN.dbConnect;  
    tblCompanies := dbCONN.tblOne;  
    tblGames := dbCONN.tblMany;  
    dbCONN.setupGrids(dbgDevelopers, dbgGames, dbgrdSQL);  
    pgcDBAdmin.ActivePageIndex := 0;  
end;  
// =====  
// {$ENDREGION}  
  
end.
```

**ANNEXURE G: SOLUTION FOR QUESTION 3****Object class**

```
unit Game_U;

interface

Uses SysUtils;

Type
  TGame = class(TObject)
  private
    fName: String;
    fOnline: Boolean;
    fDownloadCount: Integer;
    fPlatform: String;
  public
    constructor create(sName: String; bOnline: Boolean;
                      iDownloadCount: Integer; sPlatform: String);
    function onlineStatus: String;
    procedure updateDownloadCount(iNumberDownloads: Integer);
    function determinePopularity: String;
    // Provided code
    function getName: String;
    function getPlatform : String;
    function getOnline: Boolean;
    function toString : String;
  end;

implementation

{ TGame }

// =====
// Question 3.1.1 5 marks
// =====
constructor TGame.create(sName: String; bOnline: Boolean;
                        iDownloadCount: Integer; sPlatform: String);
begin
  // Question 3.1.1
  fName := sName;
  fOnline := bOnline;
  fDownloadCount := iDownloadCount;
  fPlatform := sPlatform;
end;
```

```
// =====  
// Question 3.1.2                                     4 marks  
// =====  
function TGame.onlineStatus: String;  
begin  
    // Question 3.1.2  
    if fOnline then  
        Result := 'Yes'  
    else Result := 'No';  
end;  
  
// =====  
// Question 3.1.3                                     3 marks  
// =====  
procedure TGame.updateDownloadCount(iNumberDownloads: Integer);  
begin  
    // Question 3.1.3  
    fDownloadCount := fDownloadCount + iNumberDownloads;  
end;  
  
// =====  
// Question 3.1.4                                     10 marks  
// =====  
function TGame.determinePopularity: String;  
begin  
    // Question 3.1.4  
    if fOnline then  
        begin  
            If fDownloadCount < 100000 then  
                Result := 'Not popular'  
            else if fDownloadCount < 500000 then  
                Result := 'Popular'  
            else  
                Result := 'Very popular';  
        end  
    else  
        Result := 'Not an online game';  
end;  
  
// =====  
// Question 3.1.5                                     2 marks  
// =====  
function TGame.toString: String;  
begin  
    // Question 3.1.5  
    Result := 'Name: ' + fName + #13 + 'Internet required: ' + onlineStatus +  
        #13 + 'Number of downloads: ' + IntToStr(fDownloadCount) +  
        #13 + 'Platform: ' + fPlatform;  
end;
```

```
// =====  
// Provided code - do not change  
// =====  
function TGame.getName: String;  
begin  
    Result := fName;  
end;  
  
function TGame.getOnline: Boolean;  
begin  
    Result := fOnline;  
end;  
  
function TGame.getPlatform: String;  
begin  
    Result := fPlatform;  
end;{$ENDREGION}  
  
// =====  
end.
```

**Main Form Unit**

```
// =====  
// Question 3.2.1 5 marks  
// =====  
procedure TTfrmQuestion3.btnQ3_2_1Click(Sender: TObject);  
var  
    sName, sPlatform : String;  
    bOnline: Boolean;  
    iNumberOfDownloads: integer;  
begin  
    // Provided code  
    redQ3.Lines.Clear;  
  
    sName := edtQ3_2_1.Text;  
    bOnline := cbxQ3_2_1.Checked;  
    iNumberOfDownloads := spnQ3_2_1.Value;  
    sPlatform := cmbQ3_2_1.Text;  
  
    // End of provided code  
  
    // 3.2.1 Instantiate object  
    objGame := TGame.create(sName, bOnline,  
                            iNumberOfDownloads, sPlatform);  
    redQ3.Lines.Add(objGame.toString);  
end;  
  
// =====  
// Question 3.2.2 4 marks  
// =====  
procedure TTfrmQuestion3.btnQ3_2_2Click(Sender: TObject);  
var  
    bOnline : Boolean;  
    sPlatform : String;  
begin  
    // Provided code  
    redQ3.Lines.Clear;  
  
    bOnline := cbxQ3_2_2.Checked;  
    sPlatform := cmbQ3_2_2.Text;  
  
    // 3.2.2 - Game compatibility  
    if (bOnline = objGame.getOnline) AND (sPlatform = objGame.getPlatform)  
then  
    ShowMessage('Game is compatible with user requirements.')  
    else  
    ShowMessage('Game is NOT compatible with user requirements.');
```

```
// =====  
// Question 3.2.3 5 marks  
// =====  
procedure TTfrmQuestion3.btnQ3_2_3Click(Sender: TObject);  
var  
    iNumberOfDownloads: integer;  
begin  
    // Provided code  
    redQ3.Lines.Clear;  
  
    // 3.2.3 - Update number of downloads  
    iNumberOfDownloads := StrToInt(TextBox('Number of downloads', 'Please  
enter number of downloads', ''));  
    objGame.updateDownloadCount(iNumberOfDownloads);  
    redQ3.Lines.Add(objGame.toString);  
end;  
  
// =====  
// Question 3.2.4 2 marks  
// =====  
procedure TTfrmQuestion3.btnQ3_2_4Click(Sender: TObject);  
begin  
    // Provided code  
    redQ3.Lines.Clear;  
  
    // 3.2.4 - Determine popularity of a game  
    redQ3.Lines.Add(objGame.getName + ': ' + objGame.determinePopularity);  
end;  
  
end.
```

**ANNEXURE H: SOLUTION FOR QUESTION 4**

```
unit Question4_U;

interface

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

type
  TfrmQuestion4 = class(TForm)
    Panel1: TPanel;
    GroupBox1: TGroupBox;
    btnQ4_2: TButton;
    redQ4_2: TRichEdit;
    lstQ4_2: TListBox;
    grpQ4_3: TGroupBox;
    redQ4_3: TRichEdit;
    btnQ4_3: TButton;
    lblNum1 : TLabel;
    lblNum2 : TLabel;
    lblDifference : TLabel;
    procedure btnQ4_2Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure btnQ4_3Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
    function isPangram(sSentence: String): boolean;
  end;

var
  frmQuestion4: TfrmQuestion4;
  arrNumbers: array [1..20] of real;

implementation

{$R *.dfm}

// =====
// Question 4.1 8 marks
// =====

function TfrmQuestion4.isPangram(sSentence: String): boolean;
var
  iCount: integer;
  cChar: char;
begin
  iCount := 0;
  for cChar := 'A' to 'Z' do
    begin
      if pos(cChar, UpperCase(sSentence)) > 0 then
        Inc(iCount);
    end;
  Result := iCount = 26;
end;
```

```
// =====
// Question 4.2 8 marks
// =====
```

```
procedure TfrmQuestion4.btnQ4_2Click(Sender: TObject);
var
  sSentence, sTemp, sWord, sOut: String;
  I : integer;
  bFlag : boolean;
begin
  // Provided code
  redQ4_2.Lines.Add('Original sentence' + #9 + 'Pangram' + #13);

  // Question 4.2
  I := 0;
  lstQ4_2.Items.Add('');

  repeat
    sSentence := lstQ4_2.Items[I];
    inc(I);
    if sSentence <> '' then
      begin
        if isPangram(sSentence) then
          sOut := sSentence + #9 + 'Yes'
        else if (isPangram(sSentence) = False) then
          sOut := sSentence + #9 + 'No';
        redQ4_2.Lines.Add(sOut);
      end;
  until (sSentence = '');
end;
```

```
// =====
// Question 4.3 14 marks
// =====
```

```
procedure TfrmQuestion4.btnQ4_3Click(Sender: TObject);
var
  iLoop1, iLoop2: integer;
  rNum, rLowest, rDiff, rNum1, rNum2: real;
  iCount, iUnique: integer;
  arrUnique: array [1..20] of real;
begin
  iCount := 0;
  for iLoop1 := 1 to 20 do
    begin
      iUnique := 0;
      for iLoop2 := 1 to 20 do
        if arrNumbers[iLoop1] = arrNumbers[iLoop2] then
          inc(iUnique);

      if iUnique = 1 then
        begin
          inc(iCount);
          arrUnique [iCount] := arrNumbers[iLoop1];
          redQ4_3.Lines.Add(FloatToStr(arrUnique [iCount]));
        end;
    end;
end;
```

## SC/NSC – Marking Guidelines

```
rLowest := ABS(arrUnique[1] - arrUnique[2]);

for iLoop1 := 1 to iCount - 1 do
begin
  for iLoop2 := iLoop1+1 to iCount do
  begin
    rDiff := ABS(arrUnique[iLoop1] - arrUnique[iLoop2]);
    if rDiff < rLowest then
    begin
      rLowest := rDiff;
      rNum1 := arrUnique[iLoop1];
      rNum2 := arrUnique[iLoop2];
    end;
  end;
end;
lblNum1.Caption := FloatToStr(rNum1);
lblNum2.Caption := FloatToStr(rNum2);
lblDifference.Caption := FloatToStrF(rLowest, ffFixed,8,2);
end;

// =====
// Provided code
// =====
procedure TfrmQuestion4.FormCreate(Sender: TObject);
begin
  arrNumbers[1] := 10.39;
  arrNumbers[2] := 5.33;
  arrNumbers[3] := 5.48;
  arrNumbers[4] := 9.53;
  arrNumbers[5] := 5.82;
  arrNumbers[6] := 4.21;
  arrNumbers[7] := 5.33;
  arrNumbers[8] := 2.26;
  arrNumbers[9] := 4.21;
  arrNumbers[10] := 8.48;
  arrNumbers[11] := 4.82;
  arrNumbers[12] := 9.53;
  arrNumbers[13] := 2.17;
  arrNumbers[14] := 5.33;
  arrNumbers[15] := 9.53;
  arrNumbers[16] := 8.46;
  arrNumbers[17] := 9.53;
  arrNumbers[18] := 10.26;
  arrNumbers[19] := 5.33;
  arrNumbers[20] := 9.21;
  redQ4_2.Paragraph.TabCount := 2;
  redQ4_2.Paragraph.Tab[0] := 290;
  redQ4_2.Paragraph.Tab[1] := 390;
end;

// =====
// End of provided code
// =====

end.
```