



Province of the
EASTERN CAPE
EDUCATION



**NATIONAL
SENIOR CERTIFICATE**

GRADE 12

SEPTEMBER 2022

INFORMATION TECHNOLOGY P1

MARKS: 150

TIME: 3 hours

This question paper consists of 18 pages.

INSTRUCTIONS AND INFORMATION

1. This question paper is divided into FOUR sections. Candidates must answer ALL the questions in ALL FOUR sections.
2. The duration of this examination is three hours. Because of the nature of this examination, it is important to note that you will not be permitted to leave the examination room before the end of the examination session.
3. This question paper is set with programming terms that are specific to Delphi programming language.
4. Make sure that you answer the questions according to the specifications that are given in each question. Marks will be awarded according to the set requirements.
5. Answer only what is asked in each question. For example, if the question does not ask for data validation, then no marks will be awarded for data validation.
6. Your programs must be coded in such a way that they will work with any data and not just the sample data supplied, or any data extracts that appear in the question paper.
7. Routines, such as locate, search, sort and selection, must be developed from first principles. You may NOT use the built-in features of Delphi for any of these routines.
8. All data structures must be defined by you, the programmer, unless the data structures are supplied.
9. You must save your work regularly on the disk/CD/DVD/flash disk you have been given, or on the disk space allocated to you for this examination session.
10. Make sure that your name appears as a comment in every program that you code, as well as on every event indicated.
11. If required, print the programming code of all the programs/classes that you completed. You will be given half an hour printing time after the examination session.
12. At the end of this examination session, you must hand in a disk/CD/DVD/flash disk with all your work saved on it OR you must make sure that all your work has been saved on the disk space allocated to you for this examination session. Make sure that all files can be read.

13. The files that you need to complete this question paper have been given to you on the disk/CD/DVD/flash disk or on the disk space allocated to you. The files are provided in the form of password-protected executable files.

Do the following:

- Double click on the password-protected executable file.
- Click on the extract button.
- Enter the following password: **Oh%t22**

Once extracted, the following list of files will be available in the folder **DataSept2022**:

Question 1:

Question1_u.pas
Question1_u.dfm
Question1_p.dpr
Question1_p.res

Question 2:

dbConnection_u.pas
Delivery.mdb
DeliveryBackup.mdb
Question2_u.pas
Question2_u.dfm
Question2_p.dpr
Question2_p.res

Question 3:

Question3ClassDefinition.pas
Question3_u.pas
Question3_u.dfm
Question3_p.dpr
Question3_p.res

Question 4:

votes.txt
Question4_u.pas
Question4_u.dfm
Question4_p.dpr
Question4_p.res

QUESTION 1: GENERAL PROGRAMMING SKILLS

Do the following:

- Open the incomplete program in the **Question 1** folder.
- Enter your full name as a comment in the first line of the **Question1_u.pas** file.
- Compile and execute the program. The program has no functionality currently.
- Follow the instructions below to complete the code for each section of QUESTION 1, as described in QUESTION 1.1, QUESTION 1.2, QUESTION 1.3 and QUESTION 1.4.

1.1 Button [1.1 Shape]

Write code to do the following:

- Change the shape of the shape component named **shpTest** to a circle.
- Change the colour of the shape named **shpTest** to green.
- Change the group box named **gbxQuestion12** so that it can be visible.
- Enable the group box named **gbxQuestion13**.

NOTE: Changes made in the Object Inspector will not be marked. (4)

1.2 Button [1.2 Decode]

The following code has been entered into the edit box named **edtInput**:

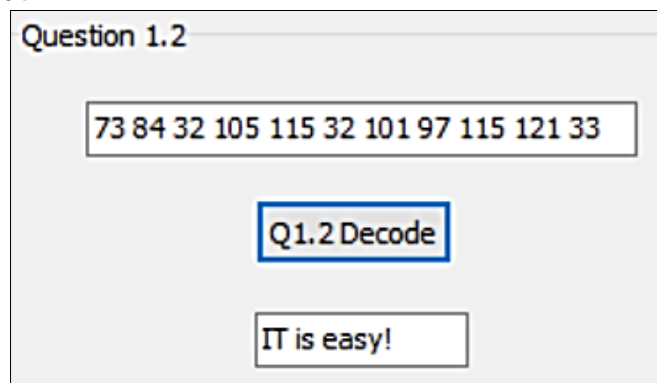
```
73 84 32 105 115 32 101 97 115 121 33
```

Each number in the string is the ordinal value of a character or symbol and each ordinal value is separated by a space.

Write code to calculate and display the sentence after converting the ordinal values to the corresponding characters and symbols, using the function CHR.

NOTE: A space has the ordinal value of 32.

Example of output:



(8)

1.3 Button [1.3 Test]

A prime number is a positive integer with exactly two positive divisors. The number 1 is not a prime number.

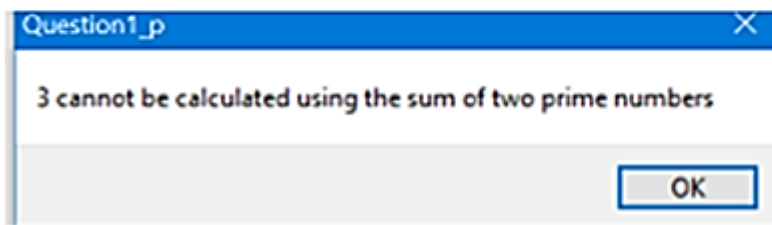
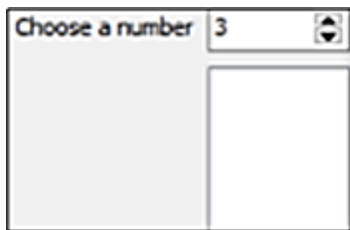
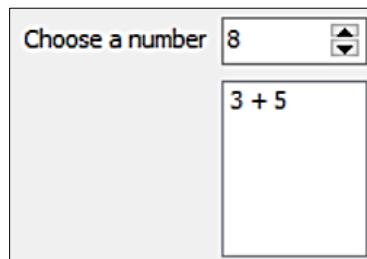
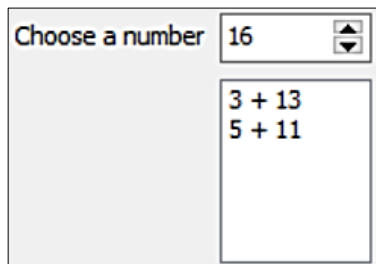
The number 16 can be expressed as the sum of two prime numbers, namely $3 + 13$ and $5 + 11$. No other additions of two prime numbers are possible to obtain the number 16.

Given a positive integer, you are required to determine the number of ways it can be expressed as the sum of two prime numbers. The sums, $x + y$ and $y + x$, are considered to be the same.

Do the following:

- Clear the listbox named **lstOutput**.
- Get input from the spinedit named **sedNumber**.
- Find all prime factors of the input and store them in an array.
- Calculate and display, in the listbox named **lstOutput**, all groups of two prime numbers that can be added together to obtain the input.
- Display a suitable message in a message component if the input cannot be obtained by calculating the sum of two prime numbers.

Examples of output:



(21)

1.4 Button [1.4 Area]

A circle has a circumference of 0.9 m and a square with a side length of 0.2 m is cut out of the circle.

Area of a circle = $\pi \times \text{radius} \times \text{radius}$.

Circumference of a circle = $2 \times \pi \times \text{radius}$

Radius of a circle = $\text{circumference} / (2 \times \pi)$

Use the following formulae:

Area of a circle = $\pi \times (0.9 / (2 \times \pi)) \times (0.9 / (2 \times \pi)) \text{ m}^2$

Area of a square = $0.2 \times 0.2 \text{ m}^2$

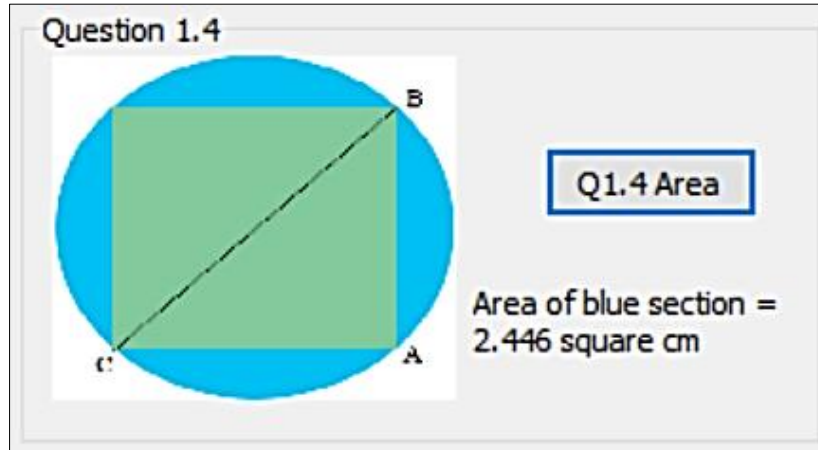
Remaining area of the circle = area of circle - area of square

You need to write code to calculate the remaining area of the circle, converted to cm^2 . ($1 \text{ m}^2 = 100 \text{ cm}^2$)

Display the answer rounded to three decimal places in the label named **lblOutput**.

NOTE: Add the answer to the existing text displayed in the label.

Example of output:



(7)

- Enter your name and surname as a comment in the first line of the program file.
- Save your program.
- A printout of the code may be required.

[40]

QUESTION 2: SQL AND DATABASE PROGRAMMING

The database **Delivery.mdb** contains the details of shops and orders that have been placed by customers and delivered by a delivery company. The database contains two tables, namely **tblShop** and **tblOrders**.

Table: **tblShop**

This table contains the descriptions of shops that are using the delivery company.

Field name	Data type	Description
ShopID	AutoNumber	A unique number assigned to each shop
Shopname	Text (50)	The name of the shop
Online	Yes/No	Indicates if the shop can sell goods online

Example of data in the **tblShop** table:

ShopID	ShopName	Online
1	Grocery Checkout	<input checked="" type="checkbox"/>
2	Pay Less Purchasing	<input type="checkbox"/>
3	Economy All Foods	<input checked="" type="checkbox"/>
4	WellWorthIt Suppliers	<input checked="" type="checkbox"/>
5	Faster Fashions	<input type="checkbox"/>
6	Edjoy Retailers	<input type="checkbox"/>
7	Get Stores	<input type="checkbox"/>
8	Span Stores	<input checked="" type="checkbox"/>
9	Discover Pharmacy	<input type="checkbox"/>
10	Collect Pharmacy	<input checked="" type="checkbox"/>

Table: **tblOrders**

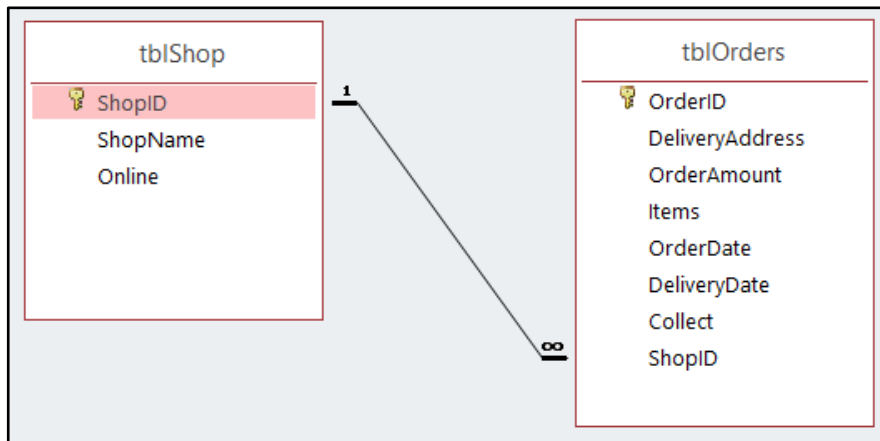
This table contains information of orders placed and delivered to addresses

Field name	Data type	Description
OrderID	Autonumber	A unique number assigned to an order
DeliveryAddress	Text (50)	The address of the buyer to which an order can be delivered.
OrderAmount	Currency	Monetary value of an order
Items	Number	Number of items ordered per order
OrderDate	Date (short date)	Date that the order was placed
DeliveryDate	Date (short date)	Date that the order was delivered
Collect	Yes/No	Indicates if a buyer will collect the order instead of delivery
ShopID	Number	Foreign key to connect to the Shop table

Example of data of the first seven records of the **tblOrders** table:

OrderID	DeliveryAddress	OrderAmount	Items	OrderDate	DeliveryDate	Collect	ShopID
1	26 Noel Street	R435.54	8	2022/01/24		<input checked="" type="checkbox"/>	1
2	6 Busby Avenue	R1 323.98	18	2022/02/12	2022/02/15	<input type="checkbox"/>	5
3	2 Nxolo Street	R304.71	8	2022/02/08	2022/02/10	<input type="checkbox"/>	3
4	46 Ekasi Avenue	R1 230.60	15	2022/08/01	2022/08/02	<input type="checkbox"/>	6
5	18 Hope Street	R672.75	5	2022/04/03	2022/04/07	<input type="checkbox"/>	3
6	12 Bains Broadway	R348.00	7	2022/04/16	2022/04/18	<input type="checkbox"/>	1
7	23 Nxolo Street	R145.00	6	2022/02/08	2022/02/10	<input type="checkbox"/>	2

The relationship between the two tables is shown below:



Do the following:

- Open the incomplete project file called **Question2_p.dpr** in the Question 2 folder.
- Enter your name as a comment in the first line of the **Question2_u.pas** unit file.
- Compile and execute the program. The program has no functionality currently.
- Follow instructions to complete the code for each question, as described in QUESTION 2.1 and QUESTION 2.2.
- **Use SQL code to answer QUESTION 2.1 and Delphi code to answer QUESTION 2.2.**

NOTE:

- The [**Restore Database**] button is provided to restore your data contained in the database to the original content. If you need to test your code on the original data, you may click this button to restore data.
- Do NOT change any of the code provided.
- Code is provided to link the GUI components to the database.
- **TWO** variables are declared as global variables, as described in the table below.
- **Use tblshop and tblorders components in QUESTION 2.2 only.**

Variable	Data type	Description
tblshop	TADOTable	Refers to the table named tblShop
tblorders	TADOTable	Refers to the table named tblOrders

2.1 In this section you may ONLY use **SQL statements** to answer QUESTION 2.1.1 to QUESTION 2.1.6.

Code to execute the SQL statements and display the results of the queries is provided. The SQL statements are incomplete.

Do the following to complete the incomplete **SQL statements** assigned to the variables sSQL1, sSQL2, sSQL3, sSQL4, sSQL5 and sSQL6 per question respectively.

2.1.1 Button [Q 2.1.1]

Write SQL code to display the **DeliveryAddress** and **DeliveryDate** of all orders, sorted to show the most recent **DeliveryDate** first.

Example of output:

DeliveryAddress	DeliveryDate
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 Button [Q 2.1.2]

Write SQL code to calculate and display the number of orders where the quantity of **Items** is greater than or equal to 20. Display the result as **Orders20orMore**.

Example of output:

Orders20orMore
4

(4)

2.1.3 Button [Q 2.1.3]

The user must enter a street name. Code has been provided for the street name "**Nxolo Street**" to be entered in an input box and saved in a variable named **sline**.

Write SQL code to display all fields of all orders that contain the street name, as stored in **sline**, in the **DeliveryAddress**.

Example of output:

OrderID	DeliveryAddress	OrderAmount	Items	OrderDate	DeliveryDate	Collect	ShopID
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 Button [Q 2.1.4]

Write an SQL statement to change the **Collect** field of all orders to True if the **DeliveryDate** field is empty.

(Code has been written to display the successfully updated table after editing.)

Example of output of the first 7 records:

OrderID	DeliveryAddress	OrderAmount	Items	OrderDate	DeliveryDate	Collect	ShopID
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 Button [Q 2.1.5]

Write SQL code to display, for all orders, the **DeliveryAddress** and the number of days it took to deliver each order in a calculated field named **DaysToDeliver**.

NOTE: All orders took place in 2022 and the **OrderDate** and **DeliveryDate** for all orders took place in the same month.

Example of output of the first 10 records:

DeliveryAddress	DaysToDeliver
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 **Button [Q 2.1.6]**

Write SQL code to display the **ShopName** and the total sales, in currency format, for each shop for all orders using the description '**TOTALSales**'.

Example of output:

ShopName	TOTALSales
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)

TOTAL QUESTION 2.1: (25)

2.2 In this section, only Delphi programming code may be used to answer QUESTION 2.2.1 and QUESTION 2.2.2.

Use the global variables, **tblshop** and **tblOrders**, provided.

NO marks will be awarded for SQL statements in QUESTION 2.2.

2.2.1 **Button [Q 2.2.1]**

Write code to display all **ShopNames** which have created online shopping websites. Code has been provided for the heading.

Example of output:

Shops that are available for online shopping
Grocery Checkout
Economy All Foods
WellWorthIt Suppliers
Span Stores
Collect Pharmacy

(5)

2.2.2 Button [Q 2.2.2]

- Code has been written to obtain a **ShopName** from the combobox named **cmbShop** and stored into a variable named **sShop**.
- Code has been written to display the headings in the richedit named **redOutput**.
- Write code to find all orders for the shop name chosen from the combobox and display, in the richedit named **redOutput**, the **OrderDate** and **OrderAmount** for each order from the chosen shop.
- You must also calculate the sum of the order amounts and display this at the end of the list of details for the shop.

Examples of output:

Shop Name: Discover Pharmacy	
Q 2.2.1	Q 2.2.2
Order Date	Amount of order

2022/11/17	R183.74
2022/04/16	R845.90
Total amount ordered: R1,029.64	

Shop Name: Get Stores	
Q 2.2.1	Q 2.2.2
Order Date	Amount of order

2022/05/07	R563.61
2022/04/03	R774.30
2022/02/08	R2,398.87
Total amount ordered: R3,736.78	

Shop Name: Grocery Checkout	
Q 2.2.1	Q 2.2.2
Order Date	Amount of order

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
Total amount ordered: R1,884.80	

(10)

TOTAL QUESTION 2.2: (15)

- | |
|--|
| <ul style="list-style-type: none"> • Enter your name and surname as a comment in the first line of the program file. • Save your program. • A printout of the code may be required. |
|--|

[40]

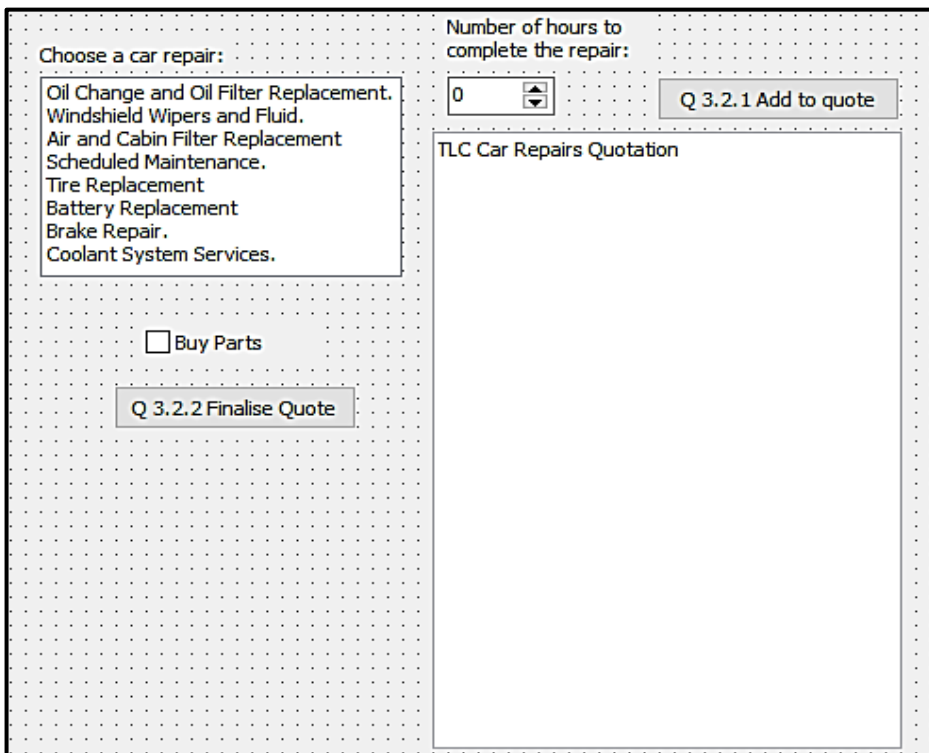
QUESTION 3: OBJECT-ORIENTED PROGRAMMING

TLC Car Repair company needs a program to record quotations for customer repairs to vehicles. The cost of labour for repairs is R70 per hour.

Do the following:

- Open the incomplete program in the Question 3 folder.
- Open the incomplete object class **Question3ClassDefinition.pas**.
- Enter your name as a comment in both **Question3ClassDefinition.pas**. and **Question3_u.pas**.
- Compile and execute the program. Currently the program has no functionality.
- Do NOT remove or change any provided code.

The following user interface is displayed:



Complete the code for this program, as specified in QUESTION 3.1 and QUESTION 3.2.

3.1 The incomplete class (**TRepair**) contains the declaration of five attributes that describe the **objRepair** object. Two methods named **getHours** and **getTotalCost** have been written for you.

NAMES OF ATTRIBUTES	DESCRIPTION
fRepairName	The description of the repair to be made
fHours	The number of hours for a repair to be completed
fCost	The labour cost of ONE repair
fTotalCost	The total cost of all repairs

Complete the code in the object class, as described in QUESTION 3.1.1 to QUESTION 3.1.4 below.

- 3.1.1 Write code for a **constructor** method named **Create** that will receive the name of a repair (description of what must be repaired on a car) and the number of hours for the repair as a whole number. Assign these parameter values to the correct attributes and set the attribute for the cost of the repair to $70 * \text{hours for the repair}$ (R70 labour cost per hour). (4)
- 3.1.2 Write code for a method named **setTotalCost** which must receive an integer value representing hours and set the attribute named **fTotalCost** by multiplying the parameter by 70. (4)
- 3.1.3 Write code for a method named **CalculateDays** that will receive one integer parameter and return an integer value using the following rules:
- Divide the parameter by 8 and if the result contains a fraction, then the number must be rounded up to the next whole number.
 - Any repair that takes less than or equal to 8 hours is considered to last one day.

Examples:

HOURS	DAYS
10	2
9	2
23	3
2	1
1	1

(6)

- 3.1.4 Write code for a method named **toString** which must return a string. The string must contain the attributes of the class joined together in the following format:

<repair name>

Hours to complete: <hours>

Labour cost: <unit cost in currency format>

Example of output:

<i>Air and Cabin Filter Replacement</i> <i>Hours to complete: 1</i> <i>Labour Cost: R70.00</i>
--

(4)

TOTAL QUESTION 3.1: (18)

3.2 An incomplete unit **Question3_u.pas** has been provided. It contains code for the object class to be accessible and has a global object variable, **objRepair**, already declared.

Do NOT delete or change any provided code.

Follow the instructions below to code the solution:

3.2.1 **Button Q 3.2.1 Add to quote**

The user will choose a repair and the number of hours for that repair using the provided input components named **lstRepairs** and **sedHours**.

Write code to do the following:

- Instantiate the object, **objRepair**, using the repair name and the number of hours for the repair.
- Use a **method of the class** to display the repair name, the number of hours to complete and the labour cost.
- A global variable named **iTotalHours** has been created and you must use a **method of the class** to add hours to this global variable.

Example of output:

```
TLC Car Repairs Quotation
Oil Change and Oil Filter Replacement.
Hours to complete: 4
Labour Cost: R.280.00

Air and Cabin Filter Replacement
Hours to complete: 6
Labour Cost: R.420.00

Battery Replacement
Hours to complete: 1
Labour Cost: R.70.00

Brake Repair.
Hours to complete: 6
Labour Cost: R.420.00
```

(11)

3.2.2 **Button Q 3.2.2 Finalise Quote**

Do the following:

- Use the global variable **iTotalHours** and **method of the class** to set the total cost of repairs.
- Get the total cost of repairs using a **method of the class** and store it in a variable.
- Using the checkbox named **cbxParts**, add R150 for transport/delivery cost to the total cost if parts have to be purchased.
- Display the total days to complete all jobs in the richedit named **redOutput** using a **method of the class** and the global variable **iTotalHours**.
- Display the total labour cost in the richedit named **redOutput**.

Example of output:

```
TLC Car Repairs Quotation

Oil Change and Oil Filter Replacement.
Hours to complete: 4
Labour Cost: R280.00

Air and Cabin Filter Replacement
Hours to complete: 6
Labour Cost: R420.00

Battery Replacement
Hours to complete: 1
Labour Cost: R70.00

Brake Repair.
Hours to complete: 6
Labour Cost: R420.00

Total days to complete all jobs = 3
Total labour cost = R1,190.00
```

If Check box is checked (Buy Parts), then example of output below:

```
Total days to complete all jobs = 3
Total labour cost = R1,340.00
```

(11)

TOTAL QUESTION 3.2: (22)

- Enter your name and surname as a comment in the first line of the program file. (In both the class and the main program that uses the class)
- Save your programs.
- A printout of the code of both units may be required.

[40]

QUESTION 4: PROBLEM-SOLVING PROGRAMMING

Voting takes place in 10 districts of a province and two parties, A and B, are the only contestants. You must complete a program to determine which party wins the election.

Do the following:

- Open the incomplete program in the **Question 4** folder.
- Enter your name and surname as a comment in the first line of the **Question4_u.pas** file.
- Compile and execute the program. Currently the program has no functionality.

Example of the text file named **votes.txt**

```
1225
2141
1565
2174
3158
1396
1525
1841
2734
2556
```

Complete the code for each question, QUESTION 4.1 and QUESTION 4.2.

NOTE:

- Good programming techniques must be applied in the design and coding of your solution.
- You may NOT change the code provided.

Given:

A global array named **ar2Votes** has been declared.

A text file named **votes.txt** is included in the Question 4 data folder and it contains 20 lines of votes of contestants A and B for the 10 districts.

4.1 Button Q 4.1 Read votes

- The two-dimensional array named **ar2Votes** has 10 columns for the 10 districts and 2 rows for the two contestants, A and B.
- Row 1 of the array is for contestant A's votes and row 2 is for contestant B's votes.
- The first line of the text file is the number of votes for contestant A and the second line is the number of votes for contestant B, etc. (All of the even rows of the text file are contestant B's votes and all odd rows are contestant A's votes).

Write code to do the following:

- Read lines from the text file named **votes.txt** into the two-dimensional array named **ar2Votes**.
- There is no output to display.

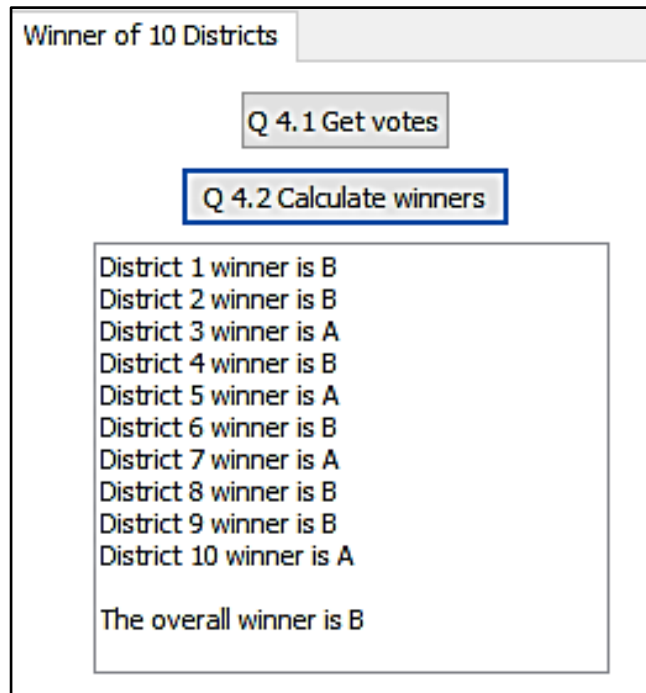
(12)

4.2 Button Q 4.2 Calculate winners

You must use the two-dimensional array named **ar2Votes** to write code to calculate and display the winner for each district.

- Display each district name and the winner for that district in the richedit named **redQ4** for all 10 districts.
- Write code to count which contestant won in the most number of districts and display the overall winner in **redQ4**.

Example of output:



(18)

- Enter your name and surname as a comment in the first line of the program file.
- Save your program.
- A printout of the code may be required.

[30]

TOTAL: 150