education

Department:
Education
PROVINCE OF KWAZULU-NATAL

JUNE EXAMINATION 2016
INFORMATION TECHNOLOGY
GRADE 11
PAPER 1

EXAMINER: V. B. RAMKILAWAN (*Stanger Secondary School*)
MODERATOR: R. PILLAY (*Mountview Secondary School*)

MARKS: 150
DURATION: 3 HOURS

INSTRUCTIONS AND INFORMATION:

1. This question paper is divided into THREE questions. Answer ALL THREE questions.
2. This paper is set in programming terms that are specific to the Delphi Programming Language (utilizing the IDE *Embarcadero Delphi 2010* or later).
3. Make sure that you answer the questions according to the specifications that are given in each question. Marks will only be awarded according to the set requirements.
4. 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.
5. 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.
6. Routines such as search, average and selection must be developed from first principles. You may not use the built-in features of a programming language for any of these routines.
7. You must save your work regularly (at least once every 5 minutes) on the disk you have been given, or the disk space allocated to you for this examination.
8. Make sure that your name appears as a comment in every program that you code as well as on every event indicated.
9. 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. Ensure that all files can be read.

DATA FILES

10. Three GUI Forms have been provided. You are not allowed to modify the GUI, text-files or any pre-defined procedures / declarations in any way.
11. Ensure that the data folder named "**June_Exam2016**" is on your Desktop.
12. Inside this folder are THREE subfolders: **Question1, Question2** and **Question3**.
13. Inside each folder is a Delphi Project File (5KB in size). The files are:
q1_p.dproj (for Question 1 – found in the **Question1** folder)
q2_p.dproj (for Question 2 – found in the **Question2** folder)
q3_p.dproj (for Question 3 – found in the **Question3** folder)
14. Question 1 has a text file named "**INGREDIENTS.TXT**" (in the **QUESTION1** folder).
15. Ensure that ALL data files are present before beginning the examination.

THIS QUESTION PAPER CONSISTS OF 8 PRINTED PAGES.

SCENARIO

The Cooler Cola Company is one of the world's largest producers of soft drinks. These refreshing fizzy drinks are immensely popular, especially during summer.
Cooler's range of drinks are very diverse, ranging from its traditional Cola to the fruit-flavoured drink called "Santa" and the lemonade-flavoured drink "Frite".

As part of the software development team at Cooler Cola you have been tasked with developing "portals" which will assist them in the daily running of the company.

QUESTION 1

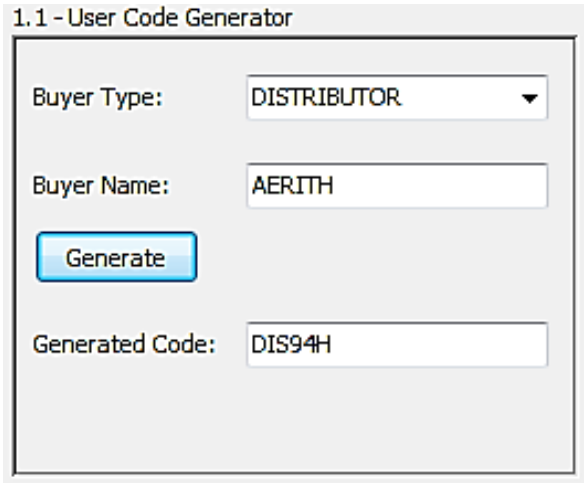
Customer Portal

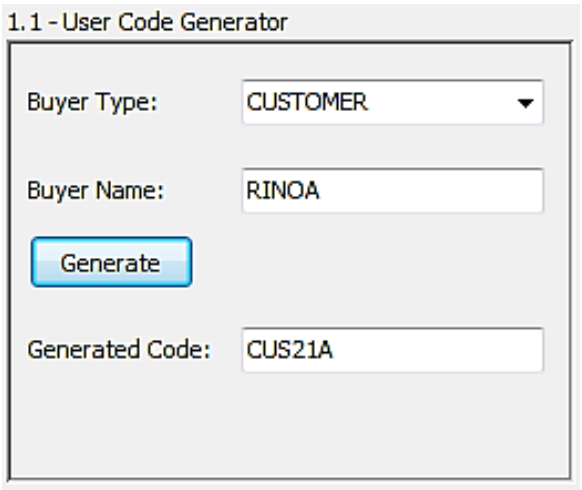
Customers are a very important component in the success of a business. Cooler Cola requires a "Customer Portal" which will help manage the most common tasks involving customers.

1.1 User Code Generator

Write code for *btnQ1_1* which will generate a unique user code.

- Obtain the Buyer Type from the ComboBox (*cmbBuyerType*).
  Buyers at Cooler Cola can fall into 1 of 3 categories:
  - Distributor
  - Wholesaler
  - Customer
- Obtain the Buyer's Name from the Edit Box (*edtBuyerName*).
- Generate and store a random positive two-digit number.
- Generate a unique code for the buyer by combining:
  - the first three characters of the buyer type with
  - the random two-digit number and
  - the last letter of the buyer's name.





- Display the generated code in the Edit Box *edtGenCode*.

[15]

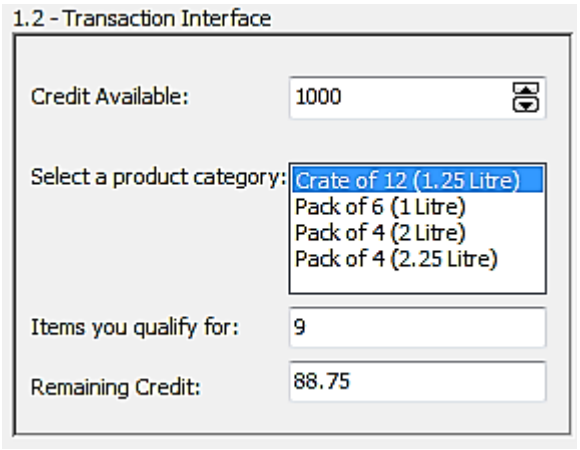
1.2 Transaction Interface

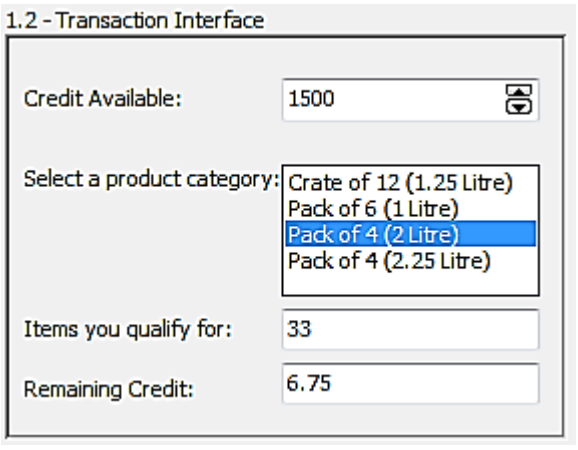
Each user purchases "credits" in order to buy products from Cooler Cola. These credits can be redeemed for different products.

Products are priced as follows:

PRODUCT	PRICE
Crate of 12 (1.25 Litre)	101.25
Pack of 6 (1 Litre)	35.88
Pack of 4 (2 Litre)	45.25
Pack of 4 (2.25 Litre)	53.81

- Obtain the amount of credit available as an Integer from spinner *sedCredit*.
- Obtain the product which the user wishes to buy from the List Box (*lstProductsQ1_2*).
- Determine the quantity of the selected product that the user qualifies for by using the credit available and its respective price.
- Calculate the user's remaining credit.
  For example:
  If the user has 1000 credits, and wishes to purchase "Crate of 12 (1.25 Litre)", he will qualify for 9 crates (Total Value: 101.25 x 9 = 911.25) and his Remaining Credit would be 88.75 (1000 – 911.25 = 88.75).





- Display the number of items that the buyer qualifies for in the Edit Box (*edtItemsQualify*).
- Display the amount of credit remaining in the Edit Box *edtCredRemaining* after the buyer has completed the transaction.

[15]

1.3 Social Responsibility Discount Calculator

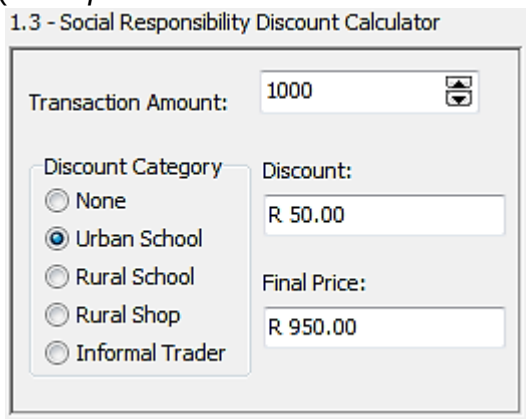
The Cooler Cola company's Social Responsibility Policy promotes assistance to sellers who service poorer areas.

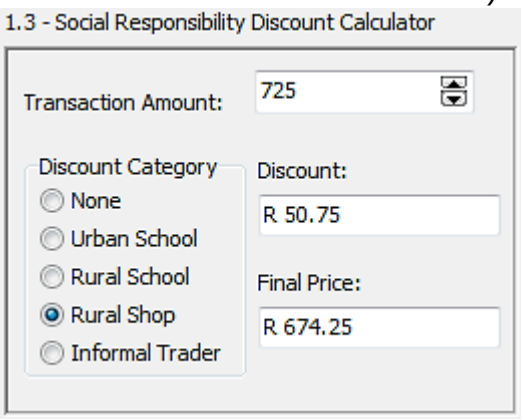
Discounts are calculated as follows:

CUSTOMER	DISCOUNT %
None	0
Urban School	5
Rural School	10
Rural Shop	7
Informal Trader	6

- Obtain the Transaction Amount entered by the user from spinner *sedTransAmount*.
- Obtain the Discount Category from the Radio Group (*rgpDiscQ1_3*).
- **Write code to:**
  - determine the Discount Amount.
  - determine the Final Amount.

(*Example of Discount and Final Amount shown in the screenshots below*.)





- Display the discount and the final price, in currency format, with TWO decimal places, in the respective edit boxes *edtDiscount* and *edtFinalPrice*

[15]

1.4 Ingredient Description

One of the most commonly asked questions by customers is what the various ingredients used in making Cooler Cola products do.

The company's Food Technology department has captured the most important ingredients in a text file named "ingredients.txt" (found in the *Question1* folder).

It is formatted as follows:

NAME OF INGREDIENT#DESCRIPTION OF INGREDIENT

***EXAMPLE**:*

Saccharin#Saccharin is a calorie-free sweetener.

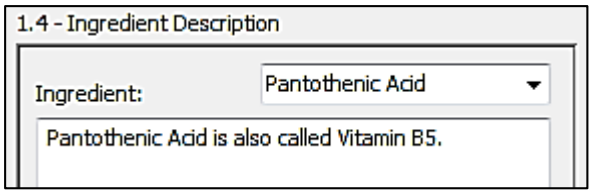
('#' acts as a delimiter)

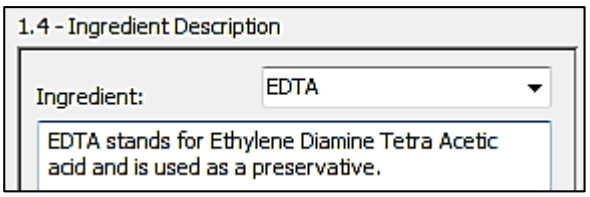
Write code to do the following:

- Check whether the text file "***ingredients.txt***" exists.
- Display a suitable message (using a message box) if the text file does not exist and close the program.

Do the following if the text file exists:

- Obtain the name of the ingredient from the Combo Box (*cmbIngredQ1_4*) provided.
- Use a conditional loop and search the text file for the ingredient obtained from the Combo Box.
- When the name of the ingredient is found, display the description of the ingredient in the memo box (*memOutput*) provided.





[15]

SUB-TOTAL: 60

QUESTION 2
Sales Portal

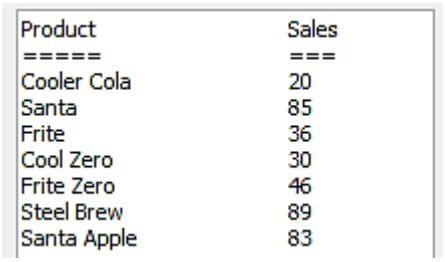
The Sales Portal is used by the company's internal planning department to project possible sales of their various products in the month ahead. Sales are expressed in 100000s sold.
An array (*arrProducts*) has been pre-declared with the company's 7 best-selling products: *'Cooler Cola', 'Santa', 'Frite', 'Cool Zero', 'Frite Zero', 'Steel Brew', 'Santa Apple'.*

A second array (*arrSales*) has also been declared; although no values have been assigned to it.
IMPORTANT: *arrProducts* and *arrSales* are parallel arrays.

2.1.1 **Write code for the Generate button (*btnGenerate2_1_1*) which will:**
Fill Array *arrSales* with random values in the range 0 to 99. (3)

2.1.2 **Write code for the Display button (*btnDisplay2_1_2*) which will:**
Display (in neat columns and with suitable headings), the name of each product from *arrProducts* and the corresponding value from *arrSales* in the Rich Edit Box (*redOutput*). (8)

N.B.: As sales are randomly generated, your output will not necessarily match the screenshot below.

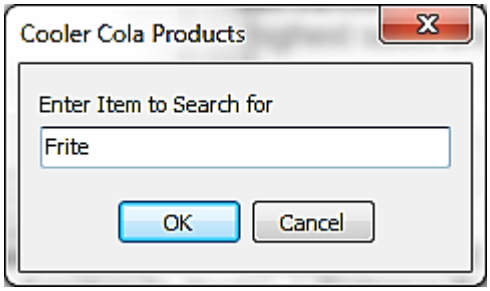


Product	Sales
=====	===
Cooler Cola	20
Santa	85
Frite	36
Cool Zero	30
Frite Zero	46
Steel Brew	89
Santa Apple	83

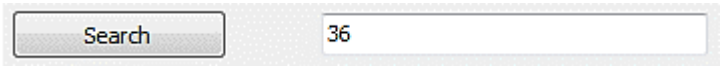
2.2 Once sales have been randomly generated, various processing tasks are required.
N.B.: As sales are randomly generated, your output will not necessarily match the screenshots below.

2.2.1 When button Highest Sales (***btnMaxSales2_2_1***) is clicked:
Determine which product had the highest sales and display the product name in Edit Box *edtHighest*. (10)

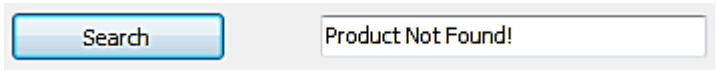
2.2.2 When button Search (***btnSearch2_2_2***) is clicked:
Use an InputBox to prompt the user to enter a name.



Write code to find the sales for the product entered by the user and display the sales value in the Edit Box *edtSearch*.

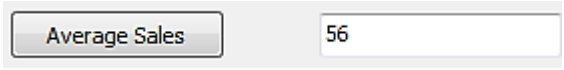


If the user types in a search query for a product which does not exist, display "Product Not Found!" in the Edit Box *edtSearch*.

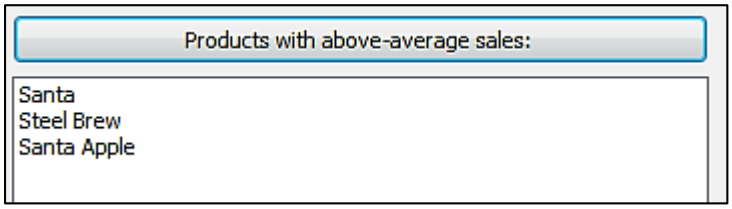


(12)

2.2.3 When button Average Sales (***btnAveSales2_2_3***) is clicked:
Write code to first determine the total sales; and then the average sales. Round the average and display the value in the Edit Box *edtAverage*. (10)



2.2.4 When button 'Products with Above Average Sales' (***btnAboveAve2_2_4***) is clicked:
Determine which products have sales above the average sales (calculated in 2.2.3) and add the product name(s) to the Memo Box (*memAbvAve*). (7)

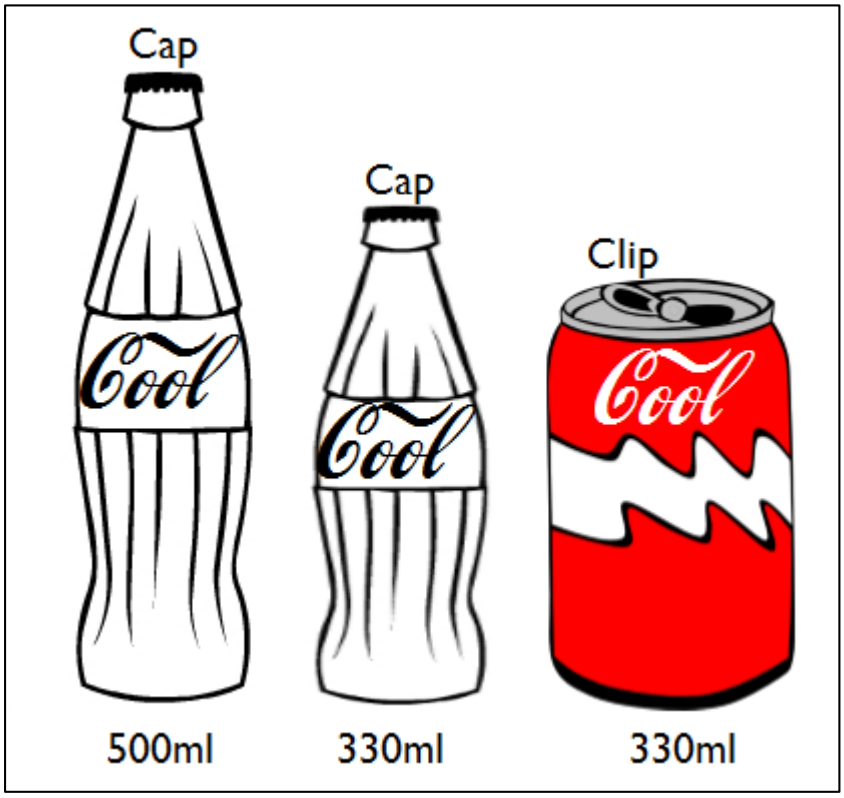


SUB-TOTAL: 50

QUESTION 3
Production Portal

Cooler Cola sells its products in smaller quantities for people "on the go". These are produced in the following quantities:

- 330ml cans
- 330ml plastic bottles
- 500ml plastic bottles.



- Each can requires one clip (used to seal the can).
- Each bottle requires one cap (used to close the bottle).
- The cap size for the 330ml bottles and the 500ml bottles are the same.

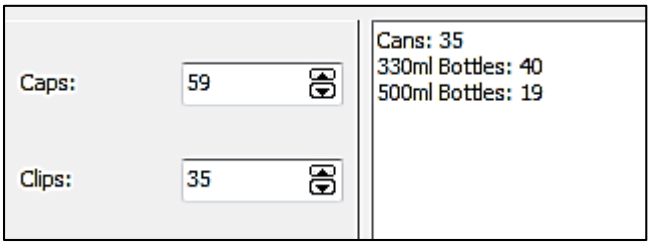
NOTE: Code for 3.1 to 3.4 is written for the PROCESS button (*btnProcess*).

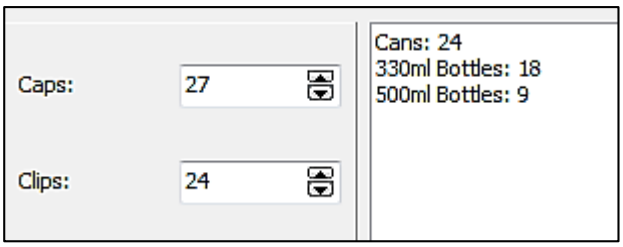
3.1 Extract the data from the spinners (*sedCaps* and *sedClips*) in order to determine how many Caps and Clips are available. (2)

3.2 **Based on demand, Cooler Cola produces twice as many 330ml plastic bottles compared to 500ml bottles.**
For example:
If there are 30 caps available, they will produce:
20 x 330ml bottles
10 x 500ml bottles
NOTE: Since 330ml bottles are more popular, if the available caps cannot be exactly distributed between the different bottle sizes in the given ratio, then the extra bottle(s) produced MUST BE of the 330ml capacity.

Write code to determine and display how many:

- cans will be produced (based on number of clips available). (2)
- 330ml bottles will be produced (based on number of caps available). (7)
- 500ml bottles will be produced (based on the number of caps available). (7)



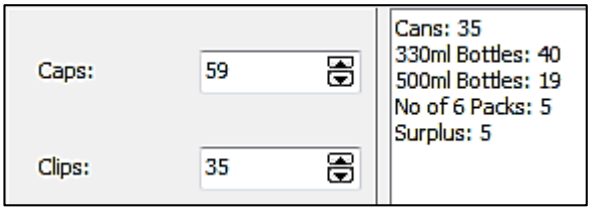


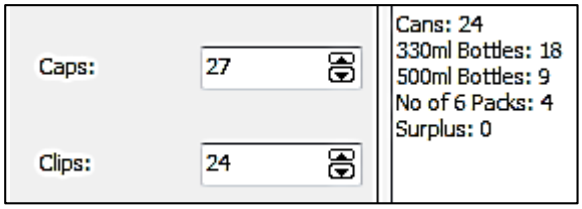
3.3 **Cooler Cola sells cans in packs of 6's**.

- Write code to determine how many complete packs of 6 will be produced using the result of cans calculated in 3.2 above. (6)

- Determine and display the number of surplus cans. (5)

  NOTE:
  Surplus cans are those left over after full packs of 6's have been bundled.
  For example if there are 35 clips, the surplus will be 5.
  30 cans will be placed into 5 packs of 6's; leaving 5 behind as a surplus.

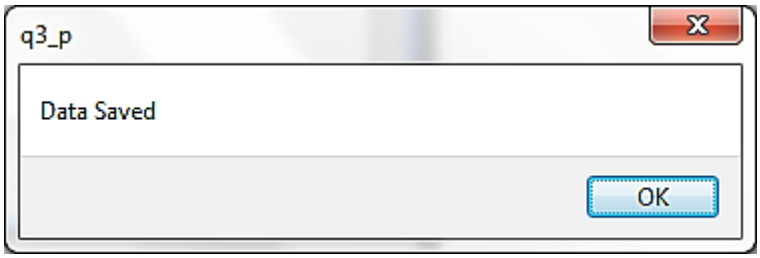




3.4 Write code to enable the Save button (***btnSaveQ3_4***). (1)

3.5 Write code for the Save button (*btnSaveQ3*_4) to:

- Create a new Text File named "ORDER.TXT". (3)
- Write the information from the Rich Edit Box (*redOutput*) to the Text File. (6)
- Display the message "Data saved" using a Message Dialogue Box. (1)



SUB-TOTAL: 40

GRAND TOTAL: 150