Province of the
EASTERN CAPE
EDUCATION

NATIONAL SENIOR CERTIFICATE

GRADE 11

NOVEMBER 2020

INFORMATION TECHNOLOGY P1

MARKS: **150**

TIME: **3 hours**

This question paper consists of 15 pages.

INSTRUCTIONS AND INFORMATION

1. This question paper consists of FOUR questions. Candidates must answer ALL four questions.
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. 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.
4. 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 this question paper.
5. Routines such as search, sort 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.
6. Save your work regularly. Ensure that all files can be read.
7. The files that you need to complete this question paper have been given 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: **Nov20C#4d**

   Once extracted, the following list of files will be available in the folder **DataNov2020**:

Question 1:

Question1_u.pas
Question1_u.dfm
Question1_p.dpr
Question1_p.res
SAFlag.txt

Question 2:

Question2_u.pas
Question2_u.dfm
Question2_p.dpr
Question2_p.res
numbers.txt

Question 3:

Question3_u.pas
Question3_u.dfm
Question3_p.dpr
Question3_p.res
dbConnection_u.pas
CovidResearch.mdb
CovidResearchBackUp.mdb

Question 4:

Question4_u.pas
Question4_u.dfm
Question4_p.dpr
Question4_p.res

Copyright reserved

QUESTION 1: GENERAL PROGRAMMING SKILLS

SCENARIO:

Complete a program to create codes for patients when they are admitted to a hospital and display the total number of patients admitted to hospitals in the Eastern Cape and the Western Cape.

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 currently has no functionality.
- Follow the instructions to complete the code for each Question 1.1, 1.2 and 1.3.

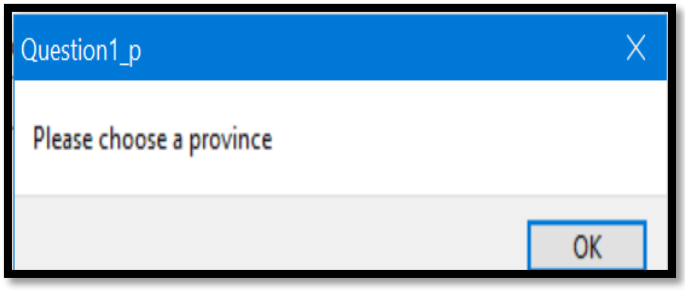
1.1 Button [Q1.1 Get Patient Code] on Question 1.1 tabsheet

Patients will enter their first name and surname into the edit box named **edtfullname**, choose a province (either Eastern Cape or Western Cape) from the combobox named **cmbprovince** and type in their year of birth (in the format YYYY) into the edit box named **edtyear**.

Write code to do the following:

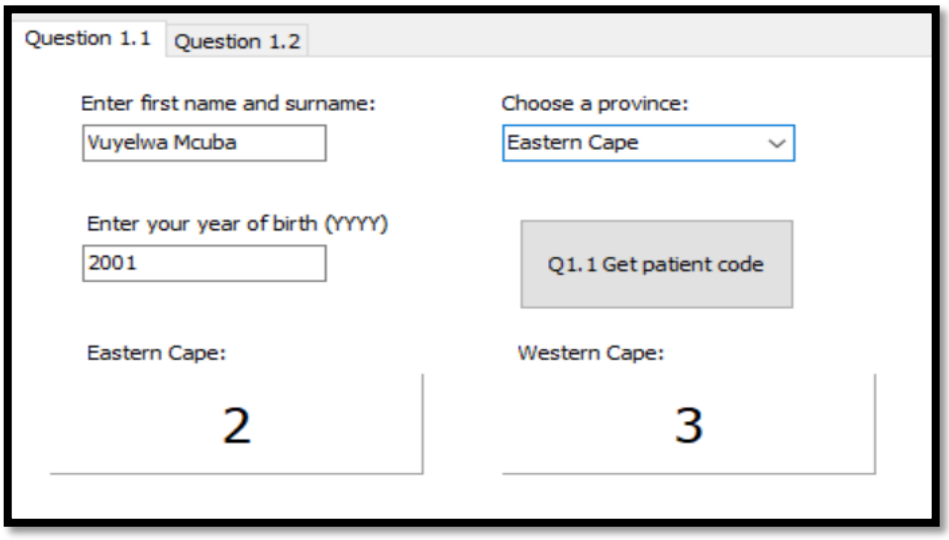
- An error message must be displayed and the procedure must exit if a patient did not choose a province.

  Example of output:



- Every time the **Get Patient Code** button is clicked, the number of patients for the selected province must be increased by one. The total number of patients for each province must be displayed on the panels named **pnlEC** and **pnlWC.**

Example of output:



¶ The patient code must be compiled and displayed in a message component using the following algorithm:

The 1st character of the first name and the 1st character of the surname must be joined to the 3rd and 4th characters of the year of birth and the first four characters of the province. A random number from 1000 to 60 000 (both included) must be added to the end of the code.

Example: Vuyelwa Mcuba was born in 2001 and has been admitted to a hospital in the Eastern Cape. She will have a patient code of VM01East26099. (The random number is 26099).

Example of output:

(26)

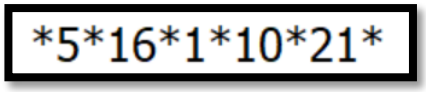
1.2 **Button [Q1.2 Encryption] on Question 1.2 tabsheet**

The user will enter a sentence in the edit box named **edtinput** which must then be encrypted and displayed in a label named **lbldisplay**.

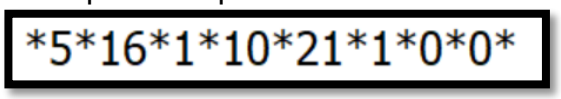
Write code to do the following:

- The input from the edit component named **edtinput** must be converted to all capital letters and then the sentence must be encrypted and displayed in the label named **lbldisplay** using the following cypher:
  - A space is represented as the number 1.
  - The characters from A to Z are represented as numbers from 2 to 27.
  - All symbols and numbers in the original sentence are represented as the number 0.
  - A star (*) must be added to the beginning and end of the encrypted string.
  - Every number of the encrypted string must be separated by a star (*).

Example of output if **ó D o** is entered:



Example of output if **ó D o 1** is entered:



(13)

1.3 **The rgppicture onclick event:**

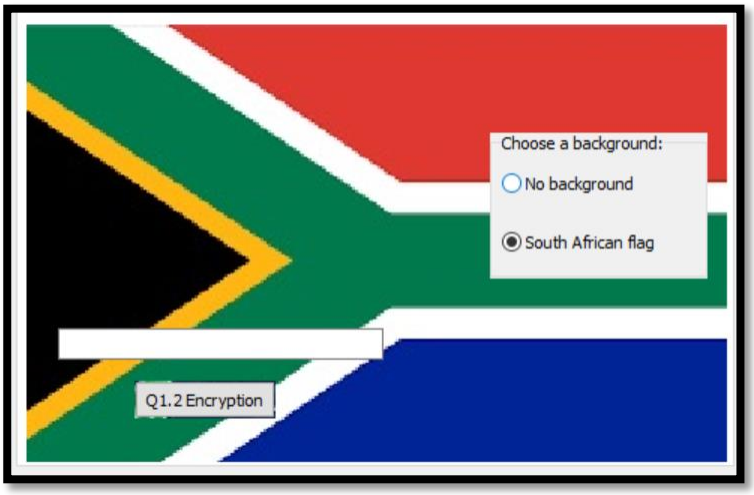
The user will choose an option from the radio group named **rgppicture.**

Write code to do the following:

If the user clicks on the radio button for ó No backgroundô, then the picture of the image component named **imgdisplay** must be set to invisible.

If the user clicks on the óSouth African Flagôradio button, then the picture named ***SAFlag.jpg*** must be displayed as the background and the picture in the image component must be set to visible.

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.	**[46]**

QUESTION 2: ARRAYS AND TEXT FILES

Do the following:

- Compile and execute the program in the Question 2 folder. The program currently has limited functionality.
- Enter your full name as a comment in the first line of the **Question2_u.pas** file.
- Complete the code for each question as described in Question 2.

2.1 Button [Q2.1 Display]

A global array named **arrqty** has been declared which contains 10 integers. The numbers in this array represent the total number of positive cases of the first 10 days of April.

Another global array named **arrdays** has been declared which contains strings representing the first 10 days of April and this array is parallel to **arrqty**.

Below is a diagrammatical representation of the array named **arrqty:**

1	2	3	4	5	6	7	8	9	10
1505	1934	1380	2003	1686	1655	1845	1462	1749	1585

Below is a diagrammatical representation of the array named **arrdays:**

1	2	3	4	5	6	7	8	9	10
3 April	9 April	1 April	10 April	6 April	5 April	8 April	2 April	7 April	4 April

Write code to do the following:

- Sort the array named **arrqty** (as well as the array named **arrdays**) in descending order from the highest value to the lowest value of the array named **arrqty.**

- Display the contents of the two global arrays in the richedit named **redout** using neat spacing. *(Code has been provided for headings and a tab stop.)*

- Calculate and display the average of all infections in the richedit, rounded to a whole number.

Copyright reserved

Example of output:

(21)

2.2 **Button [Q2.2 Multiples]**

You are provided with the text file named **numbers.txt** which contains FIVE lines representing the following data:

You are also provided with a constant array named **arradd.**

```
CONST  arradd : array [1..5] of integer = (8,6,4,2,0);
```

You are going to calculate the multiples of 18 by using the diagram below:

Numbers from the text file:	1	3	5	7	9
Numbers to append:	**8**	**6**	**4**	**2**	**0**
Multiples of 18:	18	36	54	72	90

Do the following:

- Write code to display a message and exit the procedure if the text file does not exist.
- Read the lines from the text file into an array and append the numbers 8,6,4,2 and 0 using the constant array named **arradd** to each of the corresponding numbers 1,3,5,7 and 9 from the text file, as shown in the diagram above, using a looping structure.
- Create a text file named **18.txt** ô
- Write to this text file, the five multiples of 18 (18,36,54,72,90) which you compiled using the method above. Use a looping structure.
- Display a message indicating that the file was successfully written.