Province of the
EASTERN CAPE
EDUCATION

NATIONAL
SENIOR CERTIFICATE

GRADE 11

NOVEMBER 2015

INFORMATION TECHNOLOGY P1

MARKS: 150

TIME: 3 hours

I I N F T D M 1

This question paper consists of 11 pages.

INSTRUCTIONS AND INFORMATION

1.   This is a three-hour examination. 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.

2.   You require the files listed below in order to answer the questions. They are either on a CD issued to you, or the invigilator/educator will tell you where to find them on the hard drive of the workstation you are using or in a network folder.

   Question1 Folder
   Question1_p.dpr
   Question1_u.dfm
   Question1_u.pas

   Question2 Folder
   Question2.dpr
   Question2.dfm
   Question2.pas
   Field.txt

   Question3 Folder
   Question3.dpr
   Question3.dfm
   Question3.pas

   Question4 Folder
   Question4_p.dpr
   Question4_u.dfm
   dmResultsData_u.dfm
   dmResultsData_u.pas
   Question4_u.pas
   Medals.mdb

3.   If a CD containing the above files was issued to you, write your name and surname on the label.

4.   Save your work at regular intervals as a precaution against power failures.

5.   Rename the *P1 Data folder* as your *Name and Surname*.

6.   Type in your name and surname as a comment in the first line of each program.

7.   Read ALL the questions carefully. Do only what is required.

8.   At the end of this examination session you will be required to hand in the CD with all the files with the work you have done or you must make sure that ALL the files with your work have been saved on the network as explained to you by the invigilator/educator.

9.   Ensure that ALL files can be read.

10.   During the examination you may use the HELP functions of the software. You may NOT make use of any other resource material.

> **SCENARIO**:
>
> Athletes from three different houses compete in an athletics competition. An athlete may be sponsored by six different sponsors.

QUESTION 1: DELPHI PROGRAMMING

Open the file Question1_p.dpr.

1.1 Complete the code for Question 1.1 tabsheet to generate a random number between one and three to determine the lanes that each house must use for track events.

Six lanes are available. In the case that two athletes from the same house are present, then the house in lane 1 will also be in lane 4, lane 2 will also be in lane 5, and lane 3 will also be in lane 6.

Ensure that no number is repeated.

Example of output (output will be different due to random numbers):
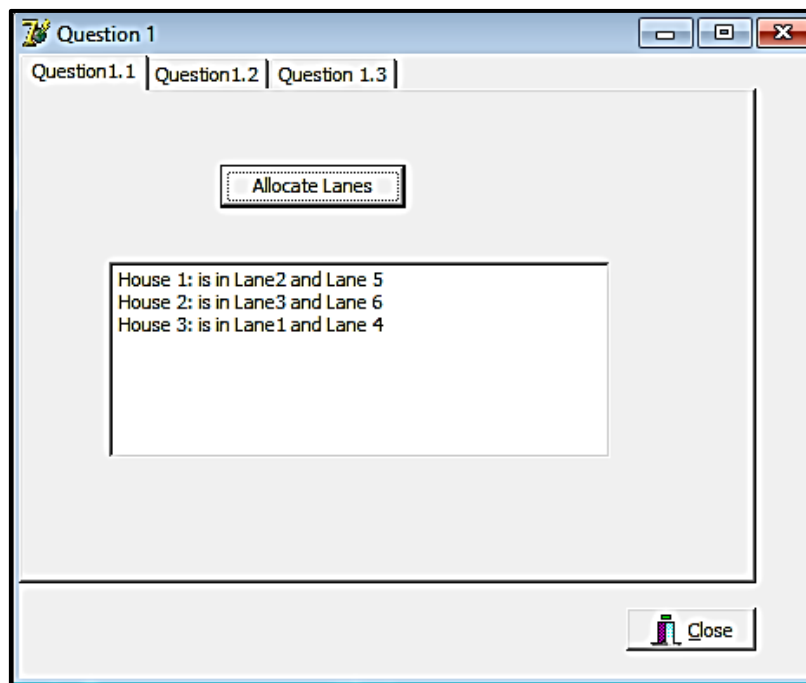
(10)

1.2 Question 1.2 tabsheet:

1.2.1 Enter the distances that the athlete has thrown when the 'Add some distances' button is clicked. Make use of an Inputbox.

Keep track of the longest distance thrown, as well as the number of throws done by the athlete. (7)

1.2.2   When 'Display the Results' button is clicked, calculate the average distance thrown. Display the total number of throws, the average, and the longest distance thrown (both rounded off to two decimal places).

(6)

1.3   With the Question 1.3 Tabsheet, each athlete has a chance to test his/her ID number to see whether it is even or odd.

The 13 digit ID number is even if the sum of the digits is even or odd if the sum of the digits is odd. If the number is even, then the athlete will be allowed one extra throw.

E.g. for the ID number 6903121156648 the sum
6+9+0+3+1+2+1+1+5+6+6+4+8=52, which is even.

The ID number must first be checked to make sure it consists of 13 characters and that it contains only numerical digits. Suitable error messages must be displayed to show what the error is.

If the ID number is valid then the sum of all the digits must be calculated and a message must be displayed informing the athlete whether he/she can have another throw.

(24)

• Enter your name and surname as a comment line in the first line of the file named Question1_u.pas.
• Save the program.
• Make a printout of the code of the Question1_u.pas file.   **[47]**

QUESTION 2: DELPHI PROGRAMMING

You are asked to examine the contents of a text file named *field.txt* saved in the folder named *Question2*. This file contains the data referring to the description of athletes who have entered for events.

During registration of athletes, a code is generated for each athlete. The codes are stored in field.txt and has the following format.

DHJ14412SAS
* Pos 1,2,3: indicate the three possible events J=Javelin, S=Shot put, D=Discus, H=High jump, L=Long jump
* Pos 4,5: indicates the age group
* Pos 6: Indicates the gender 0-4 female, 5-9 male
* Pos 7,8: indicates the athlete number between 10 and 99
* Pos 9, 10, 11: indicates the athlete's sponsor. Sponsor codes: SAS = Sasol, CAL = Caltex, SPA = Spar, BID=Bidvest, FNB=FNB, TOT=Total

2.1   Declare an array with class scope, named *arrAthletes*, which can store a maximum of 20 elements. The learner codes will be stored in *arrAthletes,* use a suitable variable to keep track of the number of athletes with valid codes. (3)

2.2   Write a method named **ValidCode** which will receive a string containing the athlete's code and it will return whether the code is valid or not.

The following rules apply:
* The code must be 11 characters long.
* The first three characters must be uppercase alphabetical characters.
* The next five characters must be numerical digits.
* The last three characters must be uppercase alphabetical characters. (15)

2.3   Complete code for the **Read Valid Text** Event Handler which must read
       information from the text file, **field.txt**.

       This method must test if the text file exists. If the file does not exist, a
       suitable error message must be displayed and the program must terminate.

       If the file exists the file must be opened, a loop must be initialised to read its
       contents. Add further code to read each line of text from the text file. The
       athlete code must be stored in the array named *arrAthletes*. The array
       should only contain the data of valid athlete codes. The invalid codes must
       be displayed as shown below.

       The total number of invalid and valid codes must be displayed after all the
       invalid codes.

       Example of output:
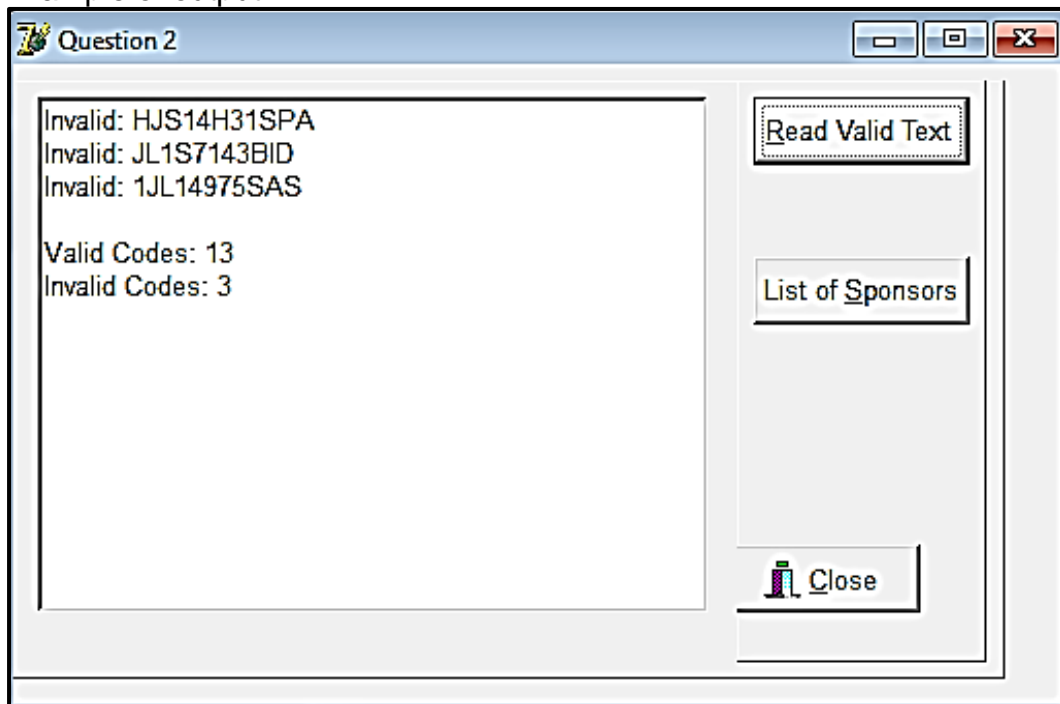
                                                                                  (19)

2.4   Write code for the **List of Sponsors** event handler. The last three characters of all the valid athlete codes indicate the sponsor of the athlete. For example, the code for athletes sponsored by FNB all end with the letters '*FNB*' and those sponsored by Spar with the letters '*SPA*'.

This procedure must display a list of all the sponsor codes as well as the total number of sponsors. (No duplicates)

Example of the output:
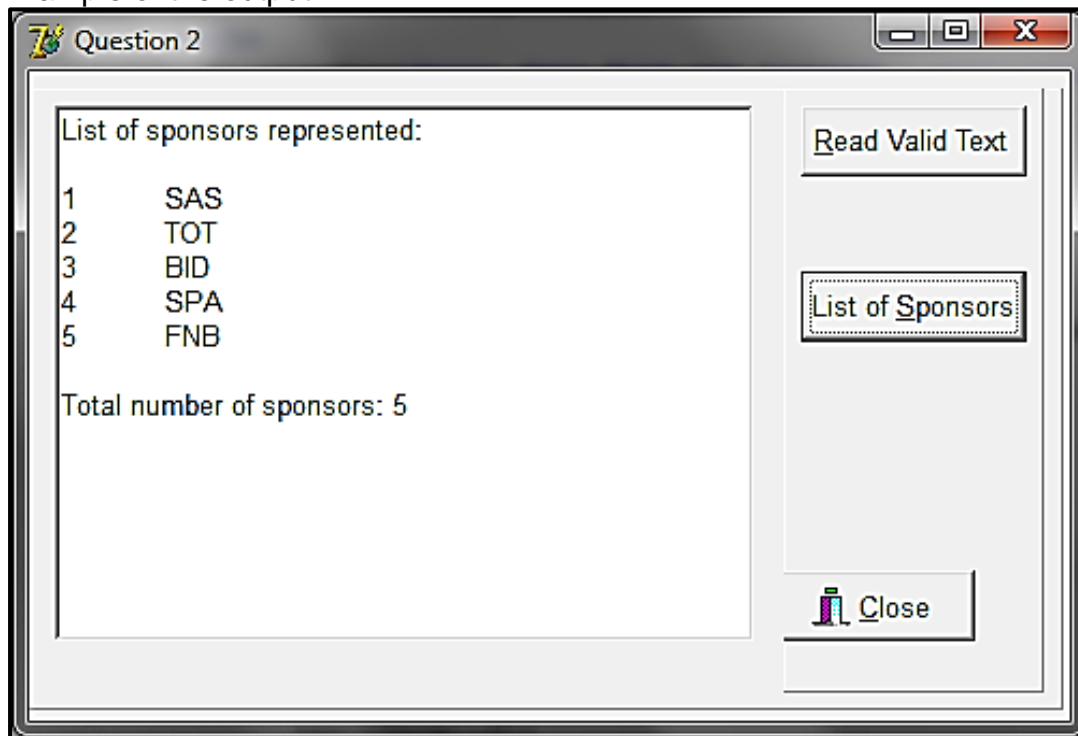
(11)

•	Enter your name and surname as a comment line in the first line of the file named Question2_u.pas
•	Save the program.
•	Make a printout of the code of the Question2_u.pas file.

[48]

QUESTION 3: DELPHI PROGRAMMING

Soon after running an event a unique code with the athlete's time in milliseconds, initials and gender is generated. An example of the code is *10102GH6*.

The first five digits represent the time in milliseconds that the athlete recorded. If this time is divided by 1000 then we get the time in seconds. In this case the athlete took 10.102 seconds.

GH are the initials of the athlete.

6 represents the gender of the athlete 0-4 female and 5-9 male.

Open Question3_p.dpr.

3.1   Write a method called **SortResults** which must sort the given array *arrResults* in ascending order.   (8)

3.2   Write code for the event handler of the **FindFastest** button. The output will be the code of the fastest athlete and the seconds the athlete took to complete the race.

       (Hint: Make use of the sorting procedure in QUESTION 3.1)

```
Code of fastest athlete is : 10152GH6
This athlete ran in 10 . 091 Seconds
```

      (5)

3.3   Write code for the event handler of the **GenderCount** button. This button must display the number of male and the number of female athletes. The gender is determined as follows: If the 8th digit of the code is between 5 and 9, the person is male, otherwise the person is female.

      Example of output:

```
Number of male athletes: 5
Number of female athletes: 7
```

      (9)

      Good programming techniques, indenting and making use of modular design in Question 2 and Question 3.   (3)

- Enter your name and surname as a comment line in the first line of the file named Question3_u.pas
- Save the program.
- Make a printout of the codes of the Question3_u.pas file.

[25]

QUESTION 4: DELPHI PROGRAMMING AND DATABASE

The database, *medals.mdb*, in the *Question4* folder supplied to you, contains results of some of the track events recorded at the interhouse athletics competition.

Field	Type	Description	Example
AthleteID	Autonumber	AutoID	
Name	Text	The name of the athlete	Thato
Surname	Text	The surname of the athlete	Mhlauli
Event	Text	The event participated in	100M
Position	Number	The position attained by the athlete	2
AgeGroup	Text	The age group of the athlete	U/13
Gender	Text	The gender of the athlete	F
House	Text	The house which the athlete belongs to	Rhino
Earning	Number	The amount of money earned by the athlete	20

The following is an example of the data contained in the table:

| athleteID	athName	athSurname	Event	Position	AgeGroup	Gender	House	Earning	Click to Add
1	Thina	Manu	100M	2	U/14	F	House1	R25.00	
2	Agie	Ruta	100M	1	U/14	F	House2	R50.00	
3	Reeva	Thambo	100M	3	U/14	F	House2	R15.00	
4	Agie	Ruta	200M	1	U/14	F	House2	R50.00	
5	Ronney	Ghuupt	400M	2	U/16	M	House2	R25.00	
6	Raaj	Naa	200M	3	Open	M	House3	R15.00	
7	Tyla	Mafu	400M	2	U/16	F	House1	R25.00	
8	Mavis	Gore	200M	2	U/16	F	House2	R25.00	
9	Diaane	Maluti	800M	1	U/16	F	House1	R50.00	
10	Thato	Terence	400M	3	U/14	F	House1	R15.00	
11	Jack	Shong	200M	2	Open	M	House3	R25.00	
12	Tyrone	Zungu	400M	1	U/14	M	House1	R50.00	
13	George	Reede	800M	3	U/16	M	House2	R15.00	

The program should be able to connect to the database named medals.mdb. If you find that the connectivity is not in place, use the following steps to establish connection with the database:
- Click on the TADOTable component named adoCon.
- Click on the Ellipse button (three dots) to the right of the 'ConnectionString' property in the Object Inspector.
- Click on the Build button which takes you to the Data Link Properties dialogue box.
- Select Microsoft Jet 4.0 OLE DB Provider and click on Next.
- The first option on the Connection tabsheet allows you to browse and find the medals.mdb file.
- Remove the user name Admin.
- Click on the Test Connection button. Click OK on each one of the open dialogue windows.

NOTE: If you cannot establish connectivity with the database at all when you execute the program you must still do and submit the programming code for marking.

4.1   **Option A**: Create a filter to display the data of the learners and events that are in House1. Indicate in a ShowMessage how many entries are recorded.
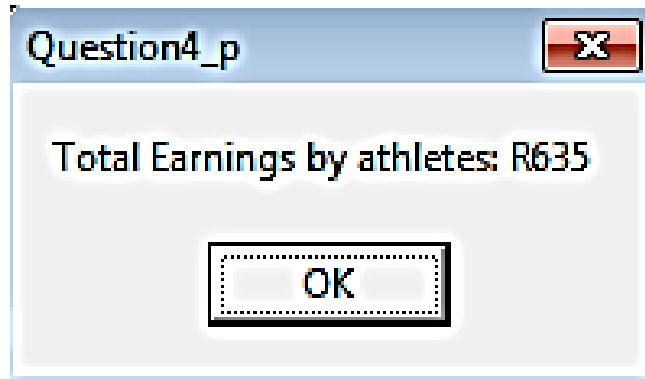
      Example of output:
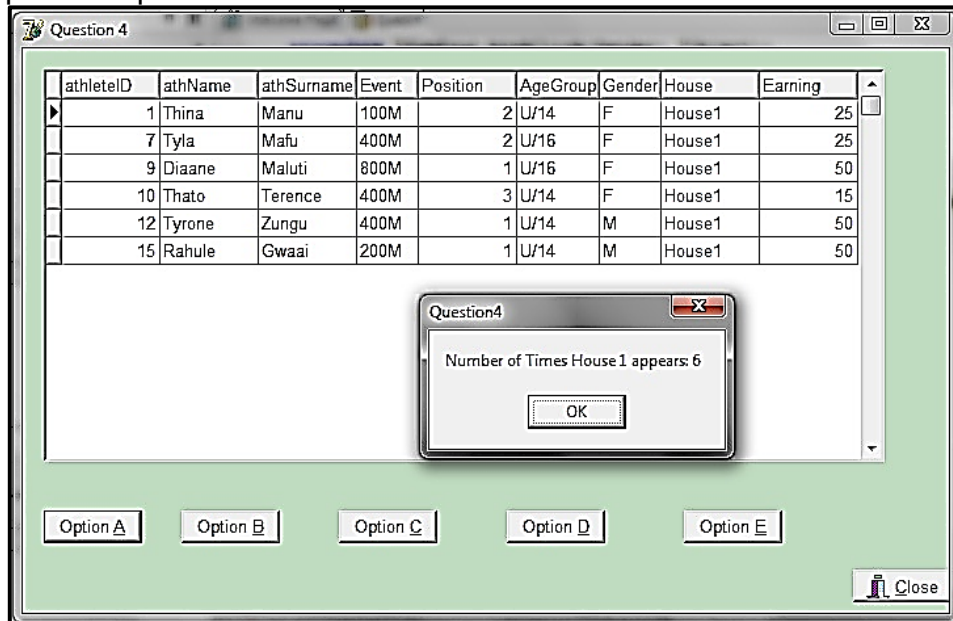
    (4)

4.2   **Option B**: The sponsors pay each athlete a certain amount each time he/she is amongst the top 3. The money differs with each event and player. Calculate the earnings by all the athletes.

      Example of output:

    (6)

4.3 **Option C**: Write code to display the data sorted in the following order: The House field sorted in descending order and the event field in ascending order.

Example of output:

(3)

4.4 **Option D:** Write code to determine if a learner with a certain surname is in the database of winners. Make use of an inputbox to enter the surname. Display the athlete's name and surname as shown below. If the athlete is not found in the database, display a suitable message.
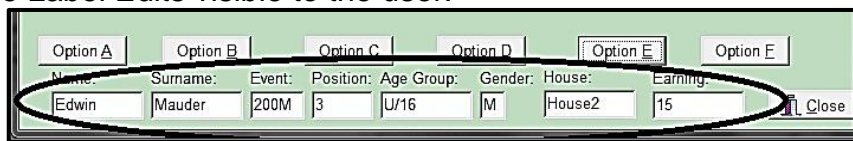
Example of output:

(7)

4.5 Adding a new athlete:

4.5.1 **Option E:** Write code in the Button E event handler, which will make all the Label Edits visible to the user.

(3)

4.5.2 **Option F:** Use the given data for the record and write code to insert the data into the table. (7)

- Enter your name and surname as a comment line in the first line of the file named Question4_u.pas.
- Save the program
- Make a printout of the code of the Question4_u.

---

[30]

TOTAL: 150