Province of the
EASTERN CAPE
EDUCATION

NATIONAL SENIOR CERTIFICATE

GRADE 12

SEPTEMBER 2012

INFORMATION TECHNOLOGY P1

MARKS: **120**

TIME **3 hours**

INFDM1

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 stiffy disk OR 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 which network folder it is. If the files are issued to you on a CD, you need to copy them onto your hard disk.

   QUESTION 1
   Question1_u.pas
   Question1_p.dpr
   Question1_u.dfm
   School.mdb
   LearnersTb.txt
   EducatorsTb.txt

   QUESTION 2
   Question2_u.pas
   Question2_p.dpr
   Question2_u.dfm
   Learner.txt

   QUESTION 3
   Question3_u.pas
   Question3_p.dpr
   Question3_u.dfm

   If a disk or CD containing the files was issued to you, write your surname on the label.

3. Save your work at regular intervals as a precaution against power failures. When saving please replace the 'X' with your surname.

4. Read ALL the questions carefully. Do only what is required by the question.

5. During the examination you may use the manuals originally supplied with the hardware and software. You may also use the HELP functions of the software. You may NOT refer to any other resource material.

6. At the end of this examination session you will be required to hand in the stiffy or CD given to you by the invigilator with your work saved on it, or you must make sure that all your work has been saved on the network as explained to you by the invigilator/educator. Ensure that all files can be read.

7. You also have to hand in printouts of the programming code for all the questions you have done.

8. All printing of programming questions will take place within an hour of the completion of the examination.

SCENARIO

Schools need to determine staffing for the following year and it is all dependent on the subject choices of each grade. With the Grade 9s choosing their new subjects, they need to be advised on the different subjects as well as requirements of certain subjects. For example, to do Physical Sciences and Information Technology, Mathematics is advisable. Another concern is to ensure that learners that cannot cope with the demands of Mathematics opt for Mathematical Literacy.

QUESTION 1: DATABASE AND DELPHI

The database, **School.mdb**, which contains data related to this topic, has been supplied to you in a folder named **Question 1**.

Two text files have been supplied as well. If you cannot use the database provided, use the text files named **EducatorsTb.txt** and **LearnersTb.txt** to create your own database named **School** containing two tables named **LearnersTb** and **EducatorsTb**. Change the data types of the fields of the tables to the specifications given below.

The **EducatorsTb** table stores data on the educators. The fields in this table are defined as follows:

EducatorsTb

Field Name	Data Type
Title	Text
Initials	Text
Surname	Text
TeacherCode	Text
Subjects	Text

The following table is an example of the data contained in the table named **EducatorsTb** in the database named **School.mdb**.

EducatorsTb

Title	Initials	Surname	TeacherCode	Subjects
Ms	A	Aerhardt	AA	ENG
Mr	A	Bosch	AB	AFR
Mrs	A	May	AM	CAT
Mr	A	Nancarrow	AN	ART
Ms	A	Simms	AS	ENG
Mr	B	Longhorn	BL	ENG
Mr	C	Byron	CB	LSC
Mrs	C	Van der Merw	CV	BEC
Ms	D	Cooper	DC	ENG
Mr	D	Kelly	DK	LSC
Dr	D	Milne	DM	TOUR
Mrs	E	Avner	EA	CAT
Mrs	E	Texeria	ET	CONS
Mr	G	Barendse	GB	AFR
Ms	G	Broekman	GM	MLIT
Mr	G	Nel	GN	AFR
Mrs	H	De Lange	HD	AFR

The **LearnersTb** table stores data on the learners in Grade 9. The fields in this table are defined as follows:

LearnersTb

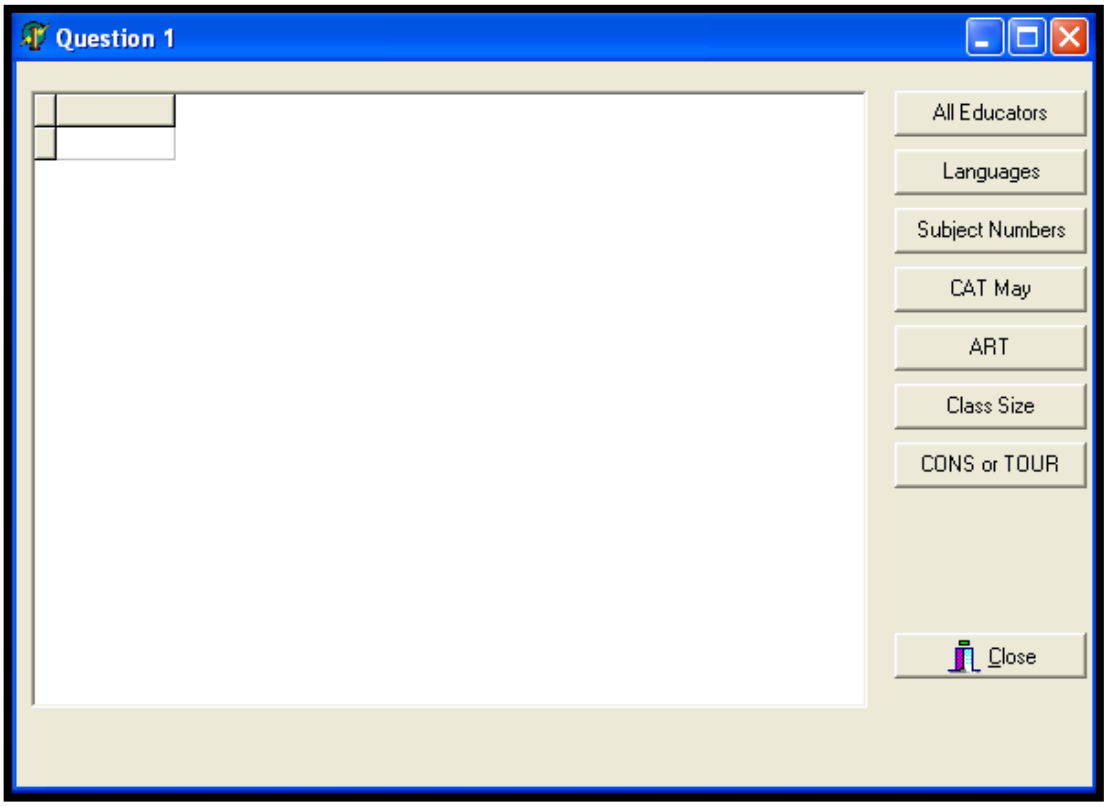
Field Name	Data Type	Description
SURNAME	Text	Surname of Learner
NAME	Text	Name of Learner
Class	Text	Class of Learner
S1	Text	Subject Combination 1: ENG
S2	Text	Subject Combination 2: AFR, XHO
S3	Text	Subject Combination 3: MAT, MLIT
S4	Text	Subject Combination 4: LO
S5	Text	Subject Combination 5: ART, LSC, IT, ECO
S6	Text	Subject Combination 6: MUS, PSC, CAT, BEC
S7	Text	Subject Combination 7: CONS, ACC, GEO, TOUR
TEACHERS	Text	TEACHER CODE OF ALL TEACHERS TEACHING LEARNER

The following table is an example of the data contained in the table named **LearnersTb** in the database named **School.mdb.**

LearnersTb

SURNAME	NAME	Clas	S1	S2	S3	S4	S5	S6	S7	TEACHERS
Jadezweni	Lee	A	ENG	AFR	MAT	LO	ECO	BEC	CONS	AA;SH;JS;ET;NJ;CV;LB
Nojoko	Jacqueline	A	ENG	XHO	MLIT	LO	ECO	PSY	TOUR	AA;MS;JS;DM;RR;NJ;GM
Adam	Philani	A	ENG	XHO	MAT	LO	IT	PSY	GEO	AA;MS;JS;MB;RR;JC;LB
Heystek	Jean-Marcelle	A	ENG	AFR	MAT	LO	ECO	PSY	ACC	AA;SH;JS;RR;NJ;LS;LB
Barnard	Anton	A	ENG	AFR	MAT	LO	LSC	MUS	TOUR	AA;SH;JS;DM;PN;CB;LB
Van Der Watt	Ross	A	ENG	AFR	MLIT	LO	LSC	CAT	GEO	AA;SH;JS;AM;CB;GM;JC
Witbooi	Robyn	A	ENG	AFR	MAT	LO	ECO	CAT	CONS	AA;SH;JS;EA;ET;NJ;LB
De Vos	Carmen	A	ENG	AFR	MAT	LO	LSC	BEC	GEO	AA;SH;JS;CV;CB;KW;LB
Nyamakazi	Jade	A	ENG	AFR	MLIT	LO	ECO	PSY	ACC	AA;SH;JS;RR;NJ;GM;LS
Swanepoel	Kholiwe	A	ENG	AFR	MLIT	LO	ECO	BEC	GEO	AA;SH;JS;NJ;CV;GM;KW
Lombard	Cershuan	A	ENG	AFR	MAT	LO	LSC	CAT	GEO	AA;SH;JS;EA;CB;KW;LB
Jeram	Nelisa	A	ENG	XHO	MAT	LO	LSC	BEC	GEO	AA;MS;JS;CV;CB;JC;LB
Gilson	Jonathan	A	ENG	AFR	MLIT	LO	ECO	BEC	ACC	AA;SH;JS;NJ;CV;GM;LS
Kivedo	Anne	A	ENG	AFR	MAT	LO	ECO	PSY	ACC	AA;SH;JS;RR;NJ;LS;LB
Pretorius	Sheldon	A	ENG	AFR	MAT	LO	LSC	PSY	GEO	AA;SH;JS;RR;CB;JC;LB
Roux	Bjorn	A	ENG	AFR	MLIT	LO	ECO	CAT	TOUR	AA;SH;JS;AM;DM;NJ;GM
Dealtry	Lindsey	A	ENG	AFR	MLIT	LO	ART	BEC	CONS	AA;SH;JS;AN;ET;CV;GM
Chivers	Stuart	A	ENG	AFR	MLIT	LO	LSC	PSY	GEO	AA;SH;JS;RR;CB;GM;KW
Twissell	Shakunthala	A	ENG	AFR	MAT	LO	LSC	BEC	ACC	AA;SH;JS;CV;CB;LS;LB
Pieterse	Carmen	A	ENG	AFR	MAT	LO	ART	PSY	ACC	AA;SH;JS;AN;RR;LS;LB
Oosthuizen	Ncumisa	A	ENG	AFR	MAT	LO	LSC	BEC	ACC	AA;SH;JS;CV;CB;LS;LB
Fumba	Pieter	A	ENG	AFR	MLIT	LO	LSC	PSY	GEO	AA;SH;JS;RR;CB;GM;JC

You have also been supplied with an incomplete Delphi program with a unit named **Question1_U** and a project named **Question1_P** in the folder named **Question 1**. Open the incomplete program.



- The program should be able to connect to the database named **School.mdb**. When you do QUESTION 1.1 and you find that the connectivity is not in place, use the following steps to establish connection with the database:
- Click on the ADOQuery component named **qrySchool**.
- 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 tab sheet allows you to browse and find the **School.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.

Marks will be awarded for the programming code that contains the SQL statements in the unit named Question1_U as well as code that makes use of an inputbox as required by the question.

1.1 Complete the code in the **All Educators** button by formulating an SQL statement to display all the fields in the EducatorsTb table in alphabetical order according to the surnames.

Example of output of the first few records:

Title	Initials	Surname	TeacherCode	Subjects
Ms	A	Aerhardt	AA	ENG
Mr	I	Allen	IA	MLIT
Ms	N	Arendse	NA	AFR
Mrs	E	Avner	EA	CAT
Mrs	S	Baker	SB	ENG
Mrs	M	Barclay	MB	IT
Mr	G	Barendse	GB	AFR
Mrs	L	Bielby	LB	MAT
Mr	A	Bosch	AB	AFR

(3)

1.2 Complete the code in the **Languages** button by formulating an SQL statement to display the Title, Surname and TeacherCode fields in the EducatorsTb table of all the teachers that teach a language (ENG/AFR/XHO).

Example of output:

Title	Surname	TeacherCode
Mr	Barendse	GB
Ms	Cooper	DC
Mr	Longhorn	BL
Mr	Sapuka	MS
Ms	Simms	AS
Ms	Aerhardt	AA
Ms	Shamley	SH
Mr	Bosch	AB
Mr	Papasavas	KP
Mrs	Purdon	TP
Mr	Nel	GN
Mrs	Baker	SB
Mrs	De Lange	HD
Ms	Arendse	NA

(5)

1.3 Complete the code in the **Subject Numbers** button by formulating an SQL statement that will determine how many Grade 9s have chosen to take a particular subject as entered via an inputbox. Display with a suitable heading.

Example of output if **IT** (which is in subject combination 5 – S5) was entered:

Potential Subject Numbers
23

(5)

1.4 Complete the code in the **CAT May** button by formulating an SQL statement that will display the name and surname of all the learners who will be in Mrs May's CAT class the following year. Mrs May's code is AM.

Example of output of the first few records:

Name	Surname
Natalie	Wessels
Jessica	Tuck
Nomawethu	Sindall
Candice	Nortje
Bjorn	Roux
Michelle	Stevens
Ross	Van Der Watt

(3)

1.5 The Art teacher has resigned and the school has decided not to offer Art to the future Grade 10s as there is not enough interest, but to let them rather change to Life Sciences. Complete the code in the **Art** button by formulating an SQL statement to update the change from Art to Life Science (LSC) in the LearnersTb. Display all the fields in the LearnersTb table once the change has been made. (5)

1.6 The Grade Head wants to ensure that all the classes are roughly the same size. Complete the code in the **Class Size** button by formulating an SQL statement that will display the class as well as the total number of learners in each class.

Example of output:

class	Class Sizes
A	30
B	30
C	33
D	31
E	29
F	29
G	29

(4)

1.7 Complete the code in the **CONS or TOUR** button by formulating an SQL statement to display all the fields of the learners who want to do Consumer Studies (CONS) or Tourism (TOUR), which are both in Subject Combination 7 (S7), and are in the 'G' class.

Example of the output of the first few lines:

SURNAME	NAME	Class	S1	S2	S3	S4	S5	S6	S7	T
Oelofse	Sian	G	ENG	AFR	MLIT	LO	LSC	CAT	TOUR	A
Smith	Thembinkosi	G	ENG	AFR	MAT	LO	IT	PSY	TOUR	A
Marsal	Ayabulela	G	ENG	XHO	MAT	LO	LSC	PSY	TOUR	A
Luscombe	Evan	G	ENG	AFR	MLIT	LO	LSC	PSY	CONS	A
Mtshepu	Mziyanda	G	ENG	XHO	MLIT	LO	ECO	PSY	TOUR	A
Bredenkamp	Lonzo	G	ENG	AFR	MLIT	LO	LSC	BEC	TOUR	A
Cadger	Bongani	G	ENG	AFR	MAT	LO	ECO	BEC	CONS	A
Duncan	Timothy	G	ENG	AFR	MLIT	LO	ECO	CAT	TOUR	A
Plant	Lauren	G	ENG	AFR	MLIT	LO	LSC	BEC	CONS	A
Fekisi	Kirsten	G	ENG	AFR	MLIT	LO	LSC	BEC	TOUR	A
Sikhakhane	Rehum	G	ENG	AFR	MLIT	LO	LSC	BEC	CONS	A

(5)

- Enter your name and surname as a comment line in the first line of the file named **Question1_UX.pas** containing the SQL statements.
- Save the unit **Question1_UX** and the project **Question1_PX** (File|Save All).
- Make a printout of the code of the **Question1_UX.pas** file. **[30]**

QUESTION 2: DELPHI PROGRAMMING

This question is intended to test object-oriented programming skills. You are required to produce a solution that includes all classes specified in the instructions. No marks will be allocated to any alternative solution such as one program not creating an object.

You have been supplied with a text file named **Learner.txt**, which contains information regarding Grade 9 learners.

The contents of the text file is shown below:

```
Karl Swart,65,78
Phillip Stroebel,54,65
Betsy Wepener,34,52
Angelique Welch,75,82
Robyn Witbooi,43,64
Patsy Carrington,80,64
Mathew Payi,67,63
Natalie Wessels,32,52
Patrick Mfuleni,71,65
Michelle de Klerk,82,76
```

The name and surname is given first, then the English mark followed by the Mathematics mark.

2.1 Create an object class (another unit) named **Subjects_UX** and save this unit as **Subjects_UX** in your **Question2_X** folder (X should be replaced by your surname). Write the following code as part of this class:

2.1.1 Define a class named **TLearner**. This class must contain the following private fields:
- fname
- feng
- fmaths
- faverage

Ensure that you choose appropriate data types for these fields. (4)

2.1.2 Write a parameterised **constructor** method which accepts the name, English and Maths marks. All the fields must be initialised in the constructor. (5)

2.1.3 Write a method named **CalcAverage** which calculates the average of the Eng. and Maths marks. (3)

2.1.4 Write a Boolean method named **Qualify** which will determine the subjects which each learner may take in Grade 10 depending on their Grade 9 marks. In order to qualify their average for English and Maths (faverage) must be 60 or more.
The method must return true if the learner qualifies to do IT, Physical Sciences or Mathematics and false if they do not qualify. (4)

2.1.5 Write a method named **tostring** that builds and returns a string with information about each learner:

fname<tab>feng<tab>fmaths<tab>faverage (3)

2.1.6 Write an appropriately named 'get' method (accessor method) to return the name of the learner. (2)

2.1.7 Write an appropriately named 'get' method (accessor method) to return the average of the learner. (2)

2.2 Write code to do the following in the **Question2_UX** unit in the given program.

2.2.1 Create an array named **arrLearners** that holds objects of **TLearner**. Write code in the OnActivate Event Handler of the form to read information from the text file **Learner.txt** according to the following steps:

a) Test if the file exists. Display a suitable message if the file does not exist and terminate the program.
b) Use a loop to ...
- Read a line of text from the text file
- Separate the text into the name, English mark and Math mark
- Use this information to create a new TLearner object and place the object in the array named arrLearners.

C) Use a counter variable to keep track of how many items there are in the array. (13)

2.2.2 **Menu Option: AVERAGE**
Display the name and average of the Mathematics and English marks for each learner.

Name	Average
Karl Swart	72
Phillip Stroebel	60
Betsy Wepener	43
Angelique Welch	79
Robyn Witbooi	54
Patsy Carrington	72
Mathew Payi	65
Natalie Wessels	42
Patrick Mfuleni	68
Michelle de Klerk	79

(4)

2.2.3 **Menu Option: IT, PHYSICAL SCIENCES, MATHEMATICS**
Call the tostring method to display the information of all the learners who qualify to do either IT, Maths or Physical Sciences.

Name	English	Maths	Average
Karl Swart	65	78	71.5
Angelique Welch	75	82	78.5
Patsy Carrington	80	64	72.0
Mathew Payi	67	63	65.0
Patrick Mfuleni	71	65	68.0
Michelle de Klerk	82	76	79.0

(4)

2.2.4 **Menu Option: SEARCH FOR A LEARNER**
Allow the user to enter the name of a learner. A suitable message must be displayed if the name was found or not. (11)

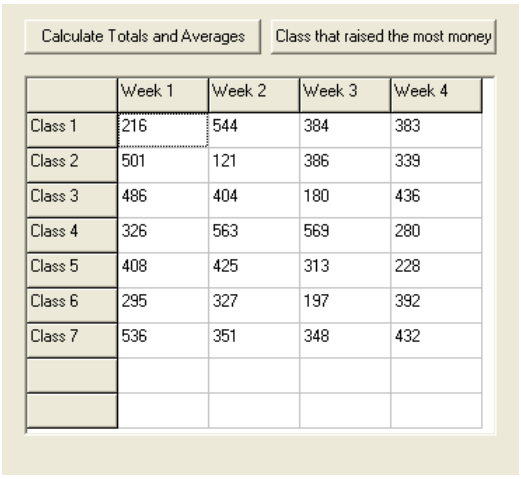
- Enter your name and surname as a comment line in the first line of the files named **Question2_UX.pas** and **Subjects_UX.pas**.
- Save the units **Question2_UX, Subjects_UX** and the project **Question2_PX** (File|Save All).
- Make a printout of the code of the **Question2_UX.pas** as well as the **Subjects_UX.pas** file. **[55]**

QUESTION 3: DELPHI PROGRAMMING

The Grade 10s have decided that they will be collecting donations for a charity organisation over a period of 4 weeks. The grade head has asked the Grade 12 IT class to write a program to manage the fundraising.

3.1 Write code for the **FormCreate** EventHandler that will place row headings (Class 1 – 7) and column headings (Week 1 – 4) in the StringGrid component. This procedure must also generate random numbers between 100 and 600 (inclusive) which will represent the amount of money collected each week by the respective classes.

Example of output:



	Week 1	Week 2	Week 3	Week 4
Class 1	216	544	384	383
Class 2	501	121	386	339
Class 3	486	404	180	436
Class 4	326	563	569	280
Class 5	408	425	313	228
Class 6	295	327	197	392
Class 7	536	351	348	432

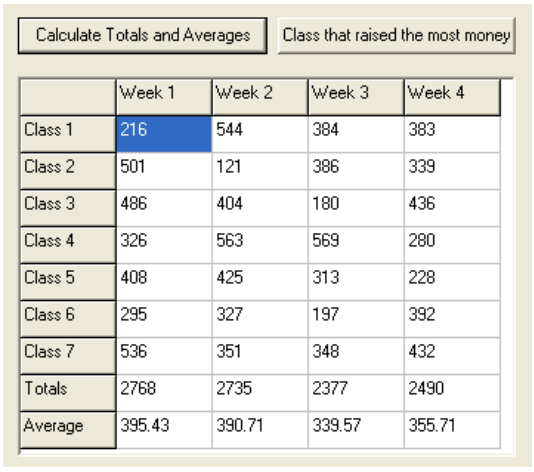
(8)

3.2 Write a procedure named **CalculateTotals** that will calculate the total amount of money collected by all the classes for each week. Display a heading as well as the totals for each week. (7)

3.3 Write a procedure named **CalculateAvgWeek** that will calculate the average amount of money collected by all the classes for each week. Display a heading as well as the averages for each week. (9)

3.4 Complete the code for the **Calculate Totals and Averages** EventHandler which will call the respective procedures.

Example of output:

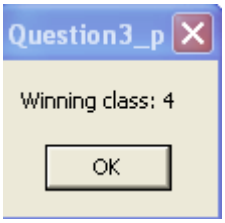


	Week 1	Week 2	Week 3	Week 4
Class 1	216	544	384	383
Class 2	501	121	386	339
Class 3	486	404	180	436
Class 4	326	563	569	280
Class 5	408	425	313	228
Class 6	295	327	197	392
Class 7	536	351	348	432
Totals	2768	2735	2377	2490
Average	395.43	390.71	339.57	355.71

(2)

3.5 Complete code for the **Class that raised the most money** EventHandler which will determine which class raised the most money. Use a ShowMessage component to display the winning class.

Example of output:



(9)

- Enter your name and surname as a comment line in the first line of the file named **Question3_UX.pas.**
- Save the unit **Question3_UX** and the project **Question3_PX** (File|Save All).
- Make a printout of the code of the **Question3_UX.pas** file. **[35]**

TOTAL: 120