



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**NATIONAL
SENIOR CERTIFICATE**

GRADE 12

INFORMATION TECHNOLOGY P1

NOVEMBER 2012

MARKS: 120

TIME: 3 hours

This question paper consists of 16 pages and 5 annexures.

INSTRUCTIONS AND INFORMATION

1. 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.
2. With regard to the formulation of the questions in terms of programming, no distinction has been made between the two programming languages in this question paper. Where required, specific instructions have been provided for Delphi and Java candidates respectively.
3. You require the files listed below in order to answer the questions. They are EITHER on a stiffer disk OR CD issued to you OR the invigilator/teacher will tell you where to find them on the hard drive of the workstation you are using OR in a network folder.

Question1_Delphi:

Question1P.dpr
Question1P.res
Question1U.dfm
Question1U.pas
tblCarnivores.txt
tblVetVisits.txt
ZooDB.mdb

Question1_Java:

tblCarnivores.txt
tblVetVisits.txt
TestQuestion1.java
Zoo.java
ZooDB.mdb

Question2_Delphi:

DataQ2.txt
Question2P.dpr
Question2P.res
Question2U.dfm
Question2U.pas
uQuest2.pas

Question2_Java:

DataQ2.txt
Quest2.java
TestQuestion2.java

Question3_Delphi:

DataQ3_Delphi.txt

Question3_Java:

DataQ3_Java.txt

If you received the files above on a disk (CD or stiffer), write your examination number on the label.

4. Type in your examination number as a comment in the first line of each program file that contains your programming code.
5. Your program should always be coded to answer the question in the way it has been formulated. You are not allowed to only copy the given output supplied in the question paper.
6. Read ALL the questions carefully. Do not do more than the questions require.

7. To help you to understand each question better, you have to read the entire question before answering any subquestions.
8. Save your work at regular intervals as a precaution against power failures.
9. There might be a technical interruption that prevents you from writing the examination, such as a power failure. When you resume writing the examination, you will be given the time remaining when the interruption began, and an additional 10 minutes.
10. 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. Candidates using Java may use the Java API files. You may NOT use any other resource material.
11. At the end of this examination session, you must hand in the disk or CD with all your work saved on it OR you must make sure that all your work has been saved on the hard drive/network as explained to you by the invigilator/teacher. Ensure that all files can be read.
12. Make printouts of the programming codes of all the programming questions you have done.
13. All printing of programming questions that you have done will take place within an hour of the completion of this examination.

SCENARIO

The Africa Zoo houses a large variety of animals. The animals are kept in different enclosures within specific areas of the zoo. The zoo also has an aquarium and a reptile park.

The zoo requires new software for the administration of the various activities at the zoo.

You are required to complete the following THREE questions using the programming language you have studied.

QUESTION 1: PROGRAMMING AND DATABASE

A Microsoft Office Access database named **ZooDB.mdb**, two text files (**tblCarnivores.txt** and **tblVetVisits.txt**) and an incomplete program are provided in the folder named **Question1_XXXX** where XXXX refers to the programming language you have studied.

The design of the tables in the **ZooDB** database and sample data for each table can be found in **ANNEXURE A**.

Do the following:

- Make a backup copy of the **ZooDB** database **BEFORE** you start answering QUESTION 1. You will need a copy of the original database to be able to test your program thoroughly.
- Rename the given folder for QUESTION 1 by replacing the name of the programming language you have studied with your examination number.
- Open the given incomplete program for QUESTION 1.
- Add your examination number as a comment in the first line of the program file.
- Compile and execute the program. The interface displays eight menu options as indicated in the section labelled **QUESTION 1** in **ANNEXURE B (Delphi)/ANNEXURE C (Java)**.

NOTE:

- An error message will be displayed if any of the options A–G are selected, due to the incomplete SQL statements.
- If you experience any problems using the database or connecting to the database, refer to **ANNEXURE D (Delphi)/ANNEXURE E (Java)** for troubleshooting hints.
- If you still experience database problems, you must nevertheless do the SQL code and submit it for marking. **Marks will only be awarded for the programming code that contains the SQL statements.**

- Complete the code for each menu option by formulating an appropriate SQL statement to display the results for the respective query as described in QUESTIONS 1.1 to 1.7 below.

NOTE: The code to some input statements and the code to execute the SQL statements and display the results of the queries have already been written as part of the given code.

1.1 Menu Option A

Display all the details of the animals stored in the **tblCarnivores** table, sorted firstly by the **FamilyName** field and secondly by the **ScientificName** field in **alphabetical order**.

Example of the output of the first five records:

EnclosureNo	FamilyName	ScientificName	GeneralName	NumAdults	NumYoung	EnclosureSize	Endangered
ZE7	Canidae	Canis adustus	Side-striped jackal	6	3	46	LE
ZE8	Canidae	Canis mesomelas	Black-backed jackal	3	3	52	LE
ZF1	Canidae	Lycaon pictus	African wild dog	2	1	60	EN
ZE9	Canidae	Otocyon megalotis	Bat-eared fox	2	3	52	LE
ZE6	Canidae	Vulpes chama	Cape fox	7	4	46	LE

:

(3)

1.2 Menu Option B

The user enters the family name of an animal via the keyboard. Display the scientific name, general name, enclosure number and enclosure size for animals belonging to the family name entered by the user, and housed in the ZE area of the zoo.

The **EnclosureNo** field indicates the specific enclosure within an area where the animals are housed, for example in enclosure number ZE5. (ZE represents the area of the zoo and 5 the specific enclosure in that area).

Example of the output of the animals in the *Canidae* family in the ZE area:

ScientificName	GeneralName	EnclosureNo	EnclosureSize
Vulpes chama	Cape fox	ZE6	46
Canis adustus	Side-striped jackal	ZE7	46
Canis mesomelas	Black-backed jackal	ZE8	52
Otocyon megalotis	Bat-eared fox	ZE9	52

(5)

1.3 **Menu Option C**

Display the categories of endangered species and the total number of animal species in each category housed at the zoo. Use a calculated field with the heading **CountAnimals** for the calculation.

Example of the output:

Endangered	CountAnimals
EN	1
LE	37
NE	1
VU	3

(4)

1.4 **Menu Option D**

Display the enclosure number and a calculated field showing the space available for each animal of the different mongoose species indicated in the **GeneralName** field.

Calculate the space available per animal by creating a formula which divides the enclosure size by the total number of animals housed in the enclosure. The calculated values should be displayed with a maximum of two decimal digits. Display the calculated field with the heading **SpacePerAnimal**.

HINT: Use the **NumAdults**, **NumYoung** and **EnclosureSize** fields as part of the formula.

Example of the output of the first five records:

EnclosureNo	SpacePerAnimal
ZD1	6.8
ZD2	5.5
ZD3	3.09
ZD4	10.5
ZD5	3.25

:

NOTE: The format of the **SpacePerAnimal** field may differ from the given example.

(6)

1.5 **Menu Option E**

Increase the number of young animals in the ZF1 enclosure by 3.

If the updating of the record was done successfully, an output message, stating that the record was successfully processed, will be displayed.

(4)

1.6 Menu Option F

The user is requested to enter the day of the month when the veterinarian (vet) visited the animals, for example 23.

Display the enclosure number, the general name, the date of the visit, the IDs of the particular animals visited and the reason why each of the animals was visited on the specific day of the month entered by the user.

HINT: Use the **DAY()** SQL function in the SQL statement.

Example of the output of the first five records for 23 September 2012:

EnclosureNo	GeneralName	VisitDate	Animal_ID	ReasonForVisit
ZF1	African wild dog	2012-09-23	ZF1_5	Assisted with birth
ZD3	Cape grey mongoose	2012-09-23	ZD3_3	Skin problem
ZD8	Selous' mongoose	2012-09-23	ZD8_1	Routine check-up
ZE6	Cape fox	2012-09-23	ZE6_5	Routine check-up
ZE4	Brown hyena	2012-09-23	ZE4_5	Routine check-up

:

NOTE: The format of the dates in the **VisitDate** field may differ from the given example.

(7)

1.7 Menu Option G

On 25 September 2012 the veterinarian examined the animal with the ID ZD5_3 in enclosure ZD5 for an ear infection. He indicated that a follow-up visit would be required. Add this data as a new record into the **tblVetVisits** table.

If the record was added to the table successfully, an output message stating that the record was processed successfully will be displayed.

HINT: Use option F to verify the adding of the record to the database. Use 25 September 2012 as input data.

(6)

- Enter your examination number as a comment in the first line of the file containing the SQL statements.
- Save your program.
- A printout of the code will be required.

[35]

QUESTION 2: OBJECT-ORIENTED PROGRAMMING

Animals are transferred to the zoo from time to time. The administrator of the zoo requires software to assist in placing these animals in suitable available enclosures.

The animals at the zoo are classified into three categories according to their size:

- L – Large animal
- M – Medium-sized animal
- S – Small animal

Different animal species are housed in different sized enclosures.

The files required for this question can be found in the folder named **Question2_XXXX** where XXXX refers to the programming language you have studied. You have been provided with a text file named **DataQ2.txt** and an incomplete program that consists of:

- A class unit (Delphi)/object class (Java) which describes the attributes of an enclosure and contains some methods
- A main form unit (Delphi)/test class (Java)

The text file contains the data of an unknown number of enclosures in the zoo.

The details of each enclosure appear on one line with the data items separated by semicolons (;) and hash (#) characters in the following format:

```
<Type of animal>;<Number of animals currently in the enclosure>#<Size of the enclosure in square metres>;<Category of animals based on size>#
```

Example of the data for the first five enclosures in the text file **DataQ2.txt**:

```
Cheetah;3#80.2;L#  
Ratel;7#50;S#  
xxx;0#20;X#  
Serval;5#80.75;M#  
xxx;0#36;X#
```

NOTE: XXX denotes an empty enclosure and therefore the category of the animals is unknown and is denoted by an X.

Do the following:

- Rename the given folder for QUESTION 2 by replacing the name of the programming language you have studied with your examination number.
- **Delphi programmers:**
 - Open the given incomplete program file **Question2P.dpr**.
 - Add your examination number as a comment in the first line of both the class unit (**uQuest2**) and the main form unit (**Question2U**).

- **Java programmers:**
 - Open the given incomplete object class **Quest2** and the test class **TestQuestion2**.
 - Add your examination number as a comment in the first line of both the object class (**Quest2**) and the test class (**TestQuestion2**).
- Compile and execute the program. The interface displays three menu options as indicated in the section labelled **QUESTION 2** in **ANNEXURE B (Delphi)/ANNEXURE C (Java)**.

2.1 **Do the following to complete the code in the class unit (Delphi)/object class (Java):**

The given **uQuest2** unit (Delphi)/**Quest2** class (Java) contains the declaration of four attributes for an enclosure object and the set (mutator) and get (accessor) methods for these attributes.

Write code for additional methods as described below.

2.1.1 Write code for a **constructor** method using parameter values to initialise the following attributes:

- Type of animal (fAType/type)
- Total number of this animal type in the enclosure (fNumber/number)
- Size of the enclosure in square metres (fSize/size)
- The category of the animals in the enclosure according to their size, that is L, M or S. (fCat/cat)

(4)

2.1.2 Write code for a method named **isSuitable** to determine whether an enclosure is suitable to house a specific group of animals. The method receives as parameters the number of animals the group consists of and their category based on their size. It returns a Boolean value.

The following applies:

- Animals can only be housed in an empty enclosure. An empty enclosure is indicated with "XXX" as animal type.
- The enclosure must be large enough to house the group of animals. The criteria for the size of the enclosures are (on the next page):

- A large animal (L) needs a minimum space of 18 square metres.
- A medium-sized animal (M) needs a minimum space of 12 square metres and a maximum space of less than 18 square metres.
- A small animal (S) needs a minimum space of 7 square metres and a maximum space of less than 12 square metres.

(7)

2.1.3 Write code for a **toString** method that will construct and return a string which includes labels and information about the object in the following format:

```
<Animal type>...<Category of the animals in the enclosure>
Enclosure size: <Enclosure size>
Number of animals: <Number of animals in the enclosure>
<Blank line>
```

Example of the output of the first two objects when their strings are returned by the **toString** method and displayed:

```
Cheetah...L
Enclosure size: 80.2
Number of animals: 3

Ratel...S
Enclosure size: 50.0
Number of animals: 7
```

(4)

2.2 Do the following to complete the code in the main form unit (Delphi)/test class (Java):

2.2.1 Declare an array capable of storing 30 enclosure objects and a counter variable to keep track of the number of objects in the array. (2)

2.2.2 Write code to test whether the text file exists.

If the file exists, write code to read lines of text from the text file. For each line of text, extract the data, create an enclosure object and assign the object to the array.

If the text file does not exist, display a suitable message and terminate the program.

NOTE: The objects have to be assigned to the array before the menu options are displayed. (15)

2.2.3 Complete each menu option as follows:

Menu Option A

Write code to display a numbered list of all the information for all the enclosures using the **toString** method.

Example of the output of some of the enclosures:

```
List of all the enclosures
=====
Enclosure number: 1
Cheetah...L
Enclosure size: 80.2
Number of animals: 3

Enclosure number: 2
Ratel...S
Enclosure size: 50.0
Number of animals: 7

Enclosure number: 3
XXX...X
Enclosure size: 20.0
Number of animals: 0

:
```

(4)

Menu Option B

A number of animals need to be transferred to the zoo and a suitable empty enclosure needs to be identified to house them.

Write code to allow the user to enter the following:

- The type of animal, for example Tiger
- The number of this type of animal
- The category of the animals based on size (for example L, M or S)

Use a conditional loop and the **isSuitable** method to search for a suitable empty enclosure in the array.

If a suitable empty enclosure is found, the attributes of the empty enclosure object in the array must be updated using the relevant set (mutator) methods. A message must be displayed indicating the enclosure number.

HINT: Use menu option A to display all enclosures to see whether the empty enclosure referred to in the message has been updated with the relevant information.

If a suitable empty enclosure is not found, an appropriate message must be displayed.

Test your program with the following test data:

Information for test data set 1:

Animal type: Tiger
Number of animals: 2
Size of the animals: L

Example of the output:

```
These animals were placed in enclosure number 5.  
  
List of all the enclosures  
=====  
:  
  
Enclosure number: 5  
Tiger...L  
Enclosure size: 36.0  
Number of animals: 2  
  
:
```

Information for test data set 2:

Animal type: Meerkat
Number of animals: 12
Size of the animals: S

Example of the output:

```
No suitable enclosure was found.
```

(11)

- Make sure that your examination number is entered as a comment in the first line of the class unit (Delphi)/object class (Java) as well as the main form unit (Delphi)/test class (Java).
- Save all the files.
- A printout of the code will be required.
- Print both the class unit (Delphi)/object class (Java) and the main form unit (Delphi)/test class (Java).

[47]

QUESTION 3: PROBLEM–SOLVING PROGRAMMING

The aquarium and the reptile park at the zoo decided to run a competition to inform visitors about their activities. Four displays labelled A, B, C and D have been constructed by the zookeepers in both the aquarium and the reptile park. Visitors take part in the competition by completing an entry form on which they indicate which display they regard as the best.

The information supplied by each participant on his/her entry form is captured as a string using the following format:

<Section><Display><Gender><Ticket number>

- **Section** indicates the section housing their favourite display. The section can be either the letter **A** (Aquarium) or the letter **R** (Reptile park).
- **Display** refers to the specific display in the indicated section which they selected as their favourite display. Each display in each section is indicated by the letters A, B, C or D.
- **Gender** refers to the participants' gender. The capital letter **M** or **F** indicates the entry form was filled out by an adult male (M) or adult female (F). The small letter **m** or **f** indicates the entry form was filled out by a boy (m) or a girl (f).
- **Ticket number** refers to the number printed on the entry form.

Examples of the strings that were captured:

- **RCf15**: A girl holding ticket number 15 selected display C of the reptile park (R) section as the best display.
- **ADM33**: An adult male holding ticket number 33 selected display D of the aquarium (A) section as the best display.

The folder **Question3_XXXX** contains a text file **DataQ3_XXXX.txt** where XXXX refers to the programming language you have studied.

The given text file contains the following:

- Code for the declaration of an array named **arrTic**. The array contains 32 strings representing the data captured from the completed entry forms.
- Java code to display a menu with three options.

Content of the **arrTic** array:

RCm158, ADM33, RCf250, RAf5, BRM32, ADm236, RCm23, RDM54, RCf17, RAM12, ADm9, RCF43, RDm140, RDm23, ACF113, ABf30, RDm22, ARf38, RCF8, RAf53, RCf12, ABF156, ADM31, ADM47, RAF48, ABF246, ABf59, RRM321, ABm36, RCF31, RAM445, ACn26

Do the following:

- Rename the given folder for QUESTION 3 by replacing the name of the programming language you have studied with your examination number.
- Create a new program/project/application.
- Add your examination number as a comment in the first line of the program file(s) you have created that will contain your code.

Java programmers: If your solution consists of more than one class file, make sure to add your examination number as a comment in all the class files.

- Save the program file(s) by using the **question number** as part of the filename in the renamed folder for QUESTION 3.
- Develop an interface as follows:

- **Delphi programmers:**

Develop an interface to display a menu as indicated in the section labelled **QUESTION 3** in **ANNEXURE B**.

- **Java programmers:**

Copy the code from the **DataQ3_Java** text file for displaying a menu as indicated in the section labelled **QUESTION 3** in **ANNEXURE C**.

- Use the following code to complete the "Quit"-menu option:
 - **Delphi programmers:**
 - `Application.Terminate;`
 - **Java programmers:**
 - `System.exit(0);`
- Copy the text for the declaration of the array which is supplied in the text file to your program file.
- Complete the code for each menu option as follows:

NOTE: Test your program by running the menu options in sequence, that is option A, then option B and finally option C.

3.1 Menu Option A

The data in the given array **arrTic** needs to be validated.

An entry is valid if the following conditions are met for the **first three characters**:

- The first character indicating the section of choice can only be the letter **A** or the letter **R**.
- The second character indicating the display of choice can only be the letters **A, B, C** or **D**.
- The third character indicating the gender can only be the letters **M, F, m** or **f**.

Write code to identify invalid entries in the **arrTic** array. For each invalid entry:

- Display the invalid entry
- Replace the invalid entry in the **arrTic** array with the letter **'Z'**.

Example of the output:

```
Invalid entries:  
BRM32  
ARf38  
RRM321  
ACn26
```

(8)

3.2 Menu Option B

The program has to randomly select a number. The number represents the position of the winning ticket from the array.

Display the randomly generated number and the code of the winning ticket with suitable labels.

If an invalid entry is selected the program has to display the word 'Invalid' and continue selecting tickets randomly until a valid entry is selected.

NOTE:

- You may assume that menu option A has been selected and executed already.
- Due to the random function, the number generated by your program may be different from those shown in the examples below.

Example 1:

```
The position of the winning ticket in the array: 24  
The winning ticket: ABM47
```

Example 2:

If a few invalid entries were randomly selected before a valid winning number was selected, the output would be as follows:

```
Invalid
Invalid
The position of the winning ticket in the array: 6
The winning ticket: ADm236
```

(11)

3.3 Menu Option C

Management wants to award medals to the three displays indicated as the best by the participants. They need a report showing the three medal-winning displays.

Points are awarded to the displays as follows:

- A display indicated as the best by an adult is awarded 12 points.
- A display indicated as the best by a child is awarded 5 points.

The medals are awarded as follows:

- The display with the highest total points is awarded the gold medal.
- The display with the second highest total points is awarded the silver medal.
- The display with the third highest total points is awarded the bronze medal.

Only data from valid ticket entries must be used to compile the report.

NOTE: An invalid entry is indicated by the single character 'Z' in the array after option A has been executed.

Example of the output:

```
Medal winning displays:
Medal      Display      Points
Gold       RC           61
Silver     AD           46
Bronze     AB           39
```

(19)

- Make sure your examination number is entered as a comment in the first line of any program files containing your code.
- Save all the files.
- A printout for the code will be required.

[38]

TOTAL: 120

ANNEXURE A: DATABASE STRUCTURE AND SAMPLE DATA

This annexure shows the data structure and sample data for the tables used in the **ZooDB** database in **QUESTION 1**.

tblCarnivores: This table contains data on all the carnivores housed at the zoo.

Table structure:

Field Name	Type	Size	Description
EnclosureNo	Text	4	A unique code assigned to each enclosure
FamilyName	Text	15	The scientific classification of the animal species housed in the enclosure
ScientificName	Text	30	The scientific name for the animals housed in the enclosure
GeneralName	Text	30	The general name for the animals housed in the enclosure
NumAdults	Number	Integer	The number of full-grown animals housed in the enclosure
NumYoung	Number	Integer	The number of young animals (not fully grown) housed in the enclosure
EnclosureSize	Number	Integer	The size of the enclosure in square metres
Endangered	Text	2	The endangered species category used. The categories are: LE – Least endangered means there are currently no identifiable risks to the species. VU – Vulnerable means the species is facing a high risk of extinction in the wild. NE – Not endangered means it does not meet any of the criteria that would categorise it as risking extinction but it is likely to do so in future. EN – Endangered means the species is facing an extremely high risk of extinction in the wild.

Sample data:

EnclosureNo	FamilyName	ScientificName	GeneralName	NumAdults	NumYoung	EnclosureSize	Endangered
ZA1	Felidae	Acinonyx jubatus	Cheetah	2	1	50	VU
ZA2	Felidae	Caracal caracal	Caracal	2	2	36	LE
ZA5	Felidae	Felis nigripes	Black-footed cat	2	0	36	VU
ZA9	Felidae	Felis silvestris	Wildcat	3	3	36	LE
ZB2	Felidae	Leptailurus serval	Serval	3	5	80	LE
ZB5	Felidae	Panthera pardus	Leopard	3	3	700	LE
ZB6	Felidae	Panthera leo	Lion	4	2	800	VU
ZB9	Viverridae	Civettictis civetta	African civet	3	0	50	LE
ZC2	Viverridae	Genetta tigrina	Cape genet	8	1	50	LE

tblVetVisits: This table contains data on the veterinarian's visits to some of the animals during one week of the month of September.

Table structure:

Field Name	Type	Size	Description
VisitID	Autonumber	Long integer	Number to uniquely identify every visit by the veterinarian
VisitDate	Date/Time	Short date	Date on which the veterinarian visited enclosure (yyyy/mm/dd)
EnclosureNo	Text	4	Code for enclosure that was visited
ReasonForVisit	Text	30	Reason why the veterinarian visited the animal in the enclosure
FollowUp	Yes/No		Was a follow-up visit scheduled? (Yes/No)
Animal_ID	Text	10	Code to uniquely identify the animal in the enclosure that was visited

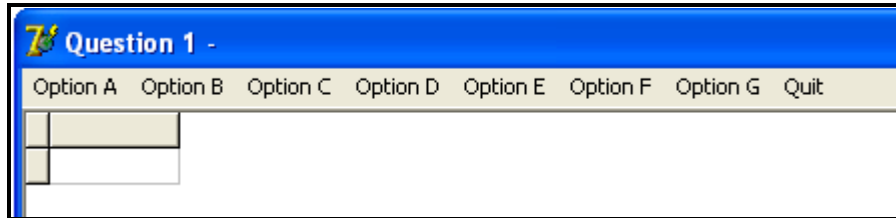
Sample data:

VisitID	VisitDate	EnclosureNo	ReasonForVisit	FollowUp	Animal_ID
1	2012/09/20	ZA1	Skin problem	<input checked="" type="checkbox"/>	ZA1_9
2	2012/09/20	ZC2	Routine check-up	<input type="checkbox"/>	ZC2_3
3	2012/09/20	ZC3	Injured eye	<input checked="" type="checkbox"/>	ZC3_8
4	2012/09/20	ZD4	Routine check-up	<input type="checkbox"/>	ZD4_2
5	2012/09/20	ZA5	Routine check-up	<input type="checkbox"/>	ZA5_2
6	2012/09/20	ZB6	Routine check-up	<input type="checkbox"/>	ZB6_6
7	2012/09/20	ZA9	Ear infection	<input checked="" type="checkbox"/>	ZA9_2

ANNEXURE B: DELPHI – GUI INTERFACES PER QUESTION

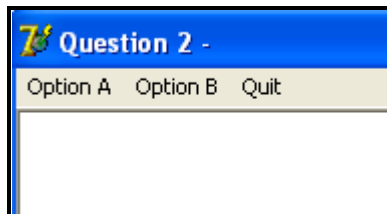
QUESTION 1

When you execute the program, the interface below will be displayed.



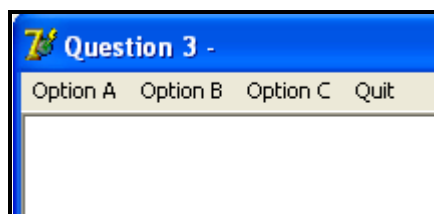
QUESTION 2

When you execute the program, the interface below will be displayed.



QUESTION 3

You are required to create the following interface as part of the solution for QUESTION 3. When you execute the program, the interface below must be displayed.



NOTE: Use the **MainMenu** component to create the menu.

ANNEXURE C: JAVA – GUI INTERFACES PER QUESTION**QUESTION 1**

When you execute the program, the interface below will be displayed.

```
MENU

Option A
Option B
Option C
Option D
Option E
Option F
Option G

Q - QUIT

Your choice?
```

QUESTION 2

When you execute the program, the interface below will be displayed.

```
MENU

Option A
Option B

Q - QUIT

Your choice?
```

QUESTION 3

Copy and use the code from the **DataQ3_Java** text file to create the following interface as part of the solution for QUESTION 3. When you execute the program, the interface below must be displayed.

```
MENU

Option A
Option B
Option C

Q - QUIT

Your choice?
```

ANNEXURE D: DELPHI – TROUBLESHOOTING DATABASE PROBLEMS

D.1 If you cannot use the given database:

- Create your own database with the name **ZooDB** that includes a table named **tblCarnivores** and another table named **tblVetVisits** in the same folder as your program for QUESTION 1.
- Import the two text files (**tblCarnivores.txt** and **tblVetVisits.txt**) to use as data for the different tables.
- The first line in the text files contains the field names to be used.

D.2 If your program cannot establish connectivity with the database:

- Make sure that the database file **ZooDB** is in the same folder as your program for QUESTION 1. If this is not the case, copy the database file into the same folder as your program.

D.3 If your program establishes a connectivity but no data is displayed:

- Click on the ADOQuery component named **qryRec**.
- Click on the Ellipsis 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.
- Click on the Provider tab to open the Provider tab sheet and select Microsoft Jet 4.0 OLE DB Provider. Click on the Next button.
- The Connection tab sheet will be displayed. The first option on the Connection tab sheet provides an Ellipsis button (three dots) that allows you to browse and look for the **ZooDB** file. You will find this file in the folder for QUESTION 1. Once you have found it, select the **ZooDB** file and then click on the Open button.
- Remove the user name Admin.
- Click on the Test Connection button.
- Click OK on each of the open dialogue windows.

ANNEXURE E: JAVA – TROUBLESHOOTING DATABASE PROBLEMS

E.1 If you cannot use the given database:

- Create your own database with the name **ZooDB** that includes a table named **tblCarnivores** and another table named **tblVetVisits** in the same folder as your program for QUESTION 1.
- Import the two text files (**tblCarnivores.txt** and **tblVetVisits.txt**) to use as data for the different tables.
- The first line in the text files contains the field names to be used.

E.2 If your program cannot establish connectivity with the database:

- Make sure that the database file **ZooDB** is in the same folder as your program for QUESTION 1. If this is not the case, copy the database file into the same folder as your program.

E.3 If you cannot establish connectivity with the given database with the given program files, use the following source code to ensure database connectivity:

```
try
{
    Class.forName ("sun.jdbc.odbc.JdbcOdbcDriver");
    String filename = "ZooDB.mdb";
    String database = "jdbc:odbc:Driver={Microsoft Access Driver (*.mdb)};DBQ=";
        database += filename.trim () + ";DriverID=22;READONLY=true}";
    Connection conn = DriverManager.getConnection (database, "", "");
}
catch (Exception e)
{
    System.out.println ("Unable to connect to the database");
}
```



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**NATIONAL
SENIOR CERTIFICATE**

GRADE 12

INFORMATION TECHNOLOGY P1

NOVEMBER 2012

MEMORANDUM

MARKS: 120

The memorandum consists of 32 pages.

GENERAL INFORMATION

- These marking guidelines are to be used as the basis for the marking session. They were prepared for use by markers, all of whom are required to attend a rigorous standardisation meeting to ensure that the guidelines are consistently interpreted and applied in the marking of candidates' scripts.
- It is acknowledged that there may be different views about some matters of emphasis or detail in the guidelines, and different interpretations of the application thereof.
- Note that learners who provide an alternate correct solution to that given in the marking guidelines will be given full credit for the relevant question.
- **ANNEXURES A, B and C** (pages 3–10) include the marking grid for each question for using either one of the two programming languages.
- **ANNEXURES D, E and F** (pages 11–19) contain the solutions for Delphi for QUESTIONS 1 to 3 in programming code.
- **ANNEXURES G, H, I and J** (pages 16–28) contain the solutions for Java for QUESTIONS 1 to 3 in programming code.
- Copies of ANNEXURES A, B and C (pages 3–6) should be made for each learner and completed during the marking session.

ANNEXURE A:**QUESTION 1: MARKING GRID – PROGRAMMING AND DATABASE****GENERAL NOTES:**

- Only penalise for the incorrect use of quotes once. Repeated incorrect use of quotes in follow up questions doesn't get penalised.
- The use of = for strings, the use of LIKE may be used as alternative

CENTRE NUMBER:		EXAMINATION NUMBER:	
QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
1.1	Query: Correct fields (or *) ✓; correct table ✓; correct ORDER BY both fields ✓	3	
	SQL: SELECT * FROM tblCarnivores ORDER BY FamilyName, ScientificName		
1.2	Query: Correct fields & table ✓; Correct WHERE clause displaying the correct family using input variable ✓ AND ✓ EnclosureNo starting with ZE ✓ using LIKE ✓	5	
	SQL(D): SELECT ScientificName, GeneralName, EnclosureNo, EnclosureSize FROM tblCarnivores WHERE EnclosureNo LIKE "ZE%" AND FamilyName = " + sX + "'		
	Alternative: ...FamilyName LIKE "%' + sX + '%" In Delphi accept Parameters wit SQL.		
1.3	Query: Correct field & table ✓; COUNT(*) ✓ AS CountAnimals ✓; GROUP BY Endangered ✓	4	
	SQL: SELECT Endangered, Count(*) AS CountAnimals FROM tblCarnivores GROUP BY Endangered		
	Alternative: Count(Endangered) Don't penalise for using Distinct		

<p>1.4</p>	<p>Query: Correct field & table ✓; <i>SpacePerAnimal</i> ✓ correctly calculated with brackets ✓; ROUND or FORMAT to 1 or 2 dec ✓; correct WHERE clause testing <i>GeneralName</i> for <i>mongoose</i> ✓ with LIKE ✓</p> <hr/> <p>SQL(D): SELECT EnclosureNo, Format(EnclosureSize / (NumAdults+NumYoung), '#.0#') AS SpacePerAnimal FROM tblCarnivores WHERE GeneralName LIKE "%mongoose"</p> <p>Alternative: Format(EnclosureSize/(NumAdults+NumYoung), '#.00') Format(EnclosureSize/(NumAdults+NumYoung), '0.00') Format(EnclosureSize/(NumAdults+NumYoung), '.00') Round(EnclosureSize/(NumAdults+NumYoung), 2)</p> <p>Also accept the use of ScientificName="Herpestidae"</p> <hr/> <p>SQL(J): SELECT EnclosureNo, Format(EnclosureSize / (NumAdults+NumYoung), '#.0#') AS SpacePerAnimal FROM tblCarnivores WHERE GeneralName LIKE '%mongoose'</p> <p>Alternative: Format(EnclosureSize/(NumAdults+NumYoung), '#.00') Format(EnclosureSize/(NumAdults+NumYoung), '0.00') Format(EnclosureSize/(NumAdults+NumYoung), '.00') Round(EnclosureSize/(NumAdults+NumYoung), 2)</p> <p>Also accept the use of ScientificName='Herpestidae'</p>	<p>6</p>	
<p>1.5</p>	<p>Query: UPDATE correct table ✓; SET the correct field ✓ with a formula increasing the value with 3 ✓; WHERE correct EnclosureNo ✓</p> <p>NOTE: the use of the same numerical field on both sides of the = sign for the formula.</p> <hr/> <p>SQL(D): UPDATE tblCarnivores SET NumYoung = NumYoung + 3 WHERE EnclosureNo = "ZF1"</p> <hr/> <p>SQL(J): UPDATE tblCarnivores SET NumYoung = NumYoung + 3 WHERE EnclosureNo = 'ZF1'</p>	<p>4</p>	

QUESTION 1: MARKING GRID – PROGRAMMING AND DATABASE (continued)

<p>1.6</p>	<p>Query: SELECT correct fields✓; FROM both tables✓; WHERE clause linking both tables on EnclosureNo ✓(left side =) ✓(right side =); using DAY✓ function on visitDate✓; with variable ✓</p> <p>SQL(D): SELECT tblVetVisits.EnclosureNo, GeneralName, VisitDate, ReasonForVisit, Animal_ID FROM tblCarnivores, tblVetVisits WHERE Day(VisitDate)='+sX+' AND tblCarnivores.EnclosureNo = tblVetVisits.EnclosureNo</p> <p>Alternative: Use aliases for tables names: SELECT C.EnclosureNo, GeneralName, VisitDate, ReasonForVisit, Animal_ID FROM tblCarnivores C, tblVetVisits V WHERE Day(VisitDate)='+sX+' AND C.EnclosureNo = V.EnclosureNo</p> <p>Alternative: Using JOIN notation: SELECT tblCarnivores.EnclosureNo, tblCarnivores.GeneralName, tblVetVisits.VisitDate, tblVetVisits.ReasonForVisit, Animal_ID FROM tblCarnivores INNER JOIN tblVetVisits ON tblCarnivores.EnclosureNo = tblVetVisits.EnclosureN WHERE Day(visitDate)='+sX</p> <p>NOTE: INNER JOIN may be replaced by LEFT or RIGHT JOIN</p>		
	<p>SQL(J): SELECT tblVetVisits.EnclosureNo, GeneralName, VisitDate, ReasonForVisit, Animal_ID FROM tblCarnivores, tblVetVisits WHERE Day(VisitDate)="+sX+" AND tblCarnivores.EnclosureNo = tblVetVisits.EnclosureNo</p> <p>Alternative: Use aliases for tables names: SELECT C.EnclosureNo, GeneralName, VisitDate, ReasonForVisit, Animal_ID FROM tblCarnivores C, tblVetVisits V WHERE Day(VisitDate)="+sX+" AND C.EnclosureNo = V.EnclosureNo</p> <p>Alternative: Use aliases for tables names AND CORRECT DATATYPE SELECT C.EnclosureNo, GeneralName, VisitDate, ReasonForVisit, Animal_ID FROM tblCarnivores C, tblVetVisits V WHERE Day(VisitDate)=""+sX+" AND C.EnclosureNo = V.EnclosureNo</p> <p>Alternative: Using JOIN notation: SELECT tblCarnivores.EnclosureNo, tblCarnivores.GeneralName, tblVetVisits.VisitDate, tblVetVisits.ReasonForVisit, Animal_ID FROM tblCarnivores INNER JOIN tblVetVisits ON tblCarnivores.EnclosureNo = tblVetVisits.EnclosureN WHERE Day([visitDate])="+sX</p> <p>NOTE: INNER JOIN may be replaced by LEFT or RIGHT JOIN</p>	<p>7</p>	

QUESTION 1: MARKING GRID – PROGRAMMING AND DATABASE (continued)

1.7	Query: INSERT INTO correct table✓; list 5 fields (not [VisitID] autonumber field)✓; Values in correct order as listed in fields✓; date value using #2012/10/25#✓; all text fields values✓; boolean field value✓ NOTE: If no fields listed but six values listed (1 mark ½)	6	
	SQL(D): INSERT INTO tblVetVisits (VisitDate, EnclosureNo, ReasonForVisit, FollowUp, Animal_ID) VALUES (#2012/10/25#, "ZD5", "Ear infection", True, "ZD5_3") Accept: yes/on/1 instead of true The use of " " for the date in the correct format (short date)		
	SQL(J): INSERT INTO tblVetVisits (VisitDate, EnclosureNo, ReasonForVisit, FollowUp, Animal_ID) VALUES (#2012/10/25#, 'ZD5', 'Ear infection', true, 'ZD5_3') Accept: yes/on/1 instead of true The use of ' ' for the date in the correct format (short date)		
	TOTAL:		
		35	

ANNEXURE B:**QUESTION 2: MARKING GRID – OBJECT-ORIENTED PROGRAMMING****GENERAL NOTES:**

- If the learner changed any given data type (e.g. character to string) penalise with ONE mark.
- Syntax error (e.g. ;) penalise only ONCE.
- In Java the use of single = in stead of == penalise only ONCE.

CENTRE NUMBER:		EXAMINATION NUMBER:	
QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
2.1.1	<p>Parameterised constructor FOUR correct parameters✓ with correct corresponding data type✓ Assign/set four parameters ✓✓</p> <p>NOTE: if all four assignment statements in incorrect order (max 1 mark)</p>	4	
2.1.2	<p>isSuitable METHOD: Parameters: number of animals and size category✓ Return a boolean ✓ Test if empty enclosure ✓; Calculate size✓ Test if suitable for animal ✓ (large/medium/small) Use nested IF/case/switch method with intervals using 7, 12 and 18 in logical format✓ (If using separate IF's a return variable must be used) Assign a return value/answer✓</p>	7	
2.1.3	<p>toString METHOD: Display type and category✓; Display two labels ✓; Display two numerical values: real/double enclosure size and number of animals✓; Display over three lines (Delphi: ...#13;...#10) (Java: ...\\n...\\n...\\n) ✓</p> <p>NOTE: Are allowed to call methods</p> <p>NOTE – Penalise with ONE mark if:</p> <ul style="list-style-type: none"> • No value is returned • Any incorrect variables used 	4	
2.2.1	<p>Declares object array – size 30 or more ✓; Counter for array✓</p>	2	

<p>2.2.2</p>	<p>INITIALISATION OF ARRAY: Test if file exist ✓ may use a IF...then...else/try... catch(FileNotFoundException e) (all statements must be included in the correct place)</p> <p><i>File doesn't exists:</i> Appropriate message ✓; Terminate program/event/exit ✓</p> <p><i>File does exists:</i> {DELPHI: AssignFile, Reset JAVA: Create object to read from file} ✓ ✓; Initialise array counter ✓ (also accept at declaration); Loop: while not eof/hasNext() ✓; Read line from text file ✓; Extract the four field values/variable values correctly (1 mark per field/variable using e.g. copy/pos/split/indexOf) ✓ ✓ ✓ ✓; Increment array counter ✓; Instantiate object using parameterized constructor ✓ ✓ (one mark left side = and one mark right side. Check correct parameter order from constructor)</p>	<p>15</p>	
<p>2.2.3</p>	<p>MENU OPTION A: Heading ✓; Loop through array ✓; Display enclosure number list ✓ Using toString method display enclosure information ✓</p> <p>MENU OPTION B: Input type, number, category of animal ✓; Initialize variables (counter) ✓; Conditional loop ✓ (inside array range AND flag) ✓ IF statement calling isSuitable method ✓ with correct order and matching parameters ✓ If found: Set 3 attributes at correct counter position in array using set methods ✓; Change flag ✓;</p> <p>Increment counter outside the if (inside the while) ✓</p> <p><i>Outside loop:</i> Display message with enclosure number ✓ (also accept if inside the IF...Then, inside the While loop) Display message if suitable enclosure was not found ✓</p> <p>NOTE: The use of a flag/break/exit in the correct place is acceptable. Incorrect placement of messages (maximum 1 mark)</p>	<p>4</p> <p>11</p>	
<p>TOTAL:</p>		<p>47</p>	

ANNEXURE C:**QUESTION 3: MARKING GRID – PROBLEM-SOLVING PROGRAMMING****GENERAL NOTES:**

- If the learner changed any given data type (e.g. character to string) penalise with ONE mark.
- Syntax error (e.g. ;) penalise only ONCE.
- In Java the use of single = in stead of == penalise only ONCE.
- In Java accept the use of the Scanner class instead of BufferedReader

CENTRE NUMBER:**EXAMINATION NUMBER:**

QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
3.1	<p>MENU OPTION A: Loop through array ✓; Validate first character: A or R ✓ Validate second character: A, B, C or D ✓ Validate third character: M, F, m or f ✓;</p> <p>Correct use of logical operators (AND/OR) in validating ✓✓ (or using nested if...then...else or separate IF statements or replace the invalid entries with Z and exit/break or using flag/variable)</p> <p>Display invalid entries ✓; Replace invalid entries with Z in array at correct index✓</p>	8	
3.2	<p>MENU OPTION B: Generate random number✓; Number in range of number of array elements (1-32 in Delphi or 0-31 in Java) ✓; Loop✓; Correct use of loop conditions✓; Check if valid ticket ✓; <i>IF valid:</i> stop the loop✓; <i>IF not valid:</i> generate another random value✓; Display message for invalid entry✓</p> <p><i>Outside loop:</i> Display message✓ and winning number✓; (also accept if inside the IF...Then, inside the While loop) Display winning ticket string from array✓</p>	11	

3.3	<p>MENU OPTION C: Declare data structure(s)✓ for category elements✓ of any data types (e.g. arrays for storing displays (AA, AB ...) and points awarded to each display; not 32 elements) <i>Calculating the points:</i> Nested loops: loop through points array (8 elements)✓ and loop through arrTic✓ (32 elements) in any order;</p> <p>IF valid ticket✓; check first two characters✓ of ticket vs category (AA, BB...)✓; check for adult/child✓; Increment points correctly for correct category (AA, BB...) ✓</p> <p><i>Sort both arrays:</i> Nested Loops✓ using correct indices for data structure declared✓; IF with correct condition ✓; Swop array values ✓; simultaneously swop the second value ✓ correct data type for temporary variable ✓</p> <p><i>Display results:</i> Heading(s)✓; Display output in columns/tabs✓; Display medals label (gold/silver/bronze)✓, correct category, and corresponding points value for that category✓</p>	19	
TOTAL:		38	

SUMMARY OF LEARNER’S MARKS:

	QUESTION 1	QUESTION 2	QUESTION 3	GRAND TOTAL
MAX. MARKS	35	47	38	120
LEARNER'S MARKS				

ANNEXURE D: SOLUTION FOR QUESTION 1: DELPHI

```

unit Question1_UMEMO;
//Solution for Question 1
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, DB, ADODB, Grids, DBGrids, ExtCtrls, Buttons, Menus;
type
  TfrmRec = class(TForm)
    qryRec: TADOQuery;
    dsrQry: TDataSource;
    grdRec: TDBGrid;
    mnuMain: TMainMenu;
    mnuOptionA: TMenuItem;
    mnuOptionB: TMenuItem;
    mnuOptionC: TMenuItem;
    mnuOptionD: TMenuItem;
    mnuOptionE: TMenuItem;
    mnuOptionF: TMenuItem;
    mnuOptionG: TMenuItem;
    mnuQuit: TMenuItem;
    procedure mnuOptionAClick(Sender: TObject);
    procedure mnuOptionBClick(Sender: TObject);
    procedure mnuOptionCClick(Sender: TObject);
    procedure mnuOptionDClick(Sender: TObject);
    procedure mnuOptionEClick(Sender: TObject);
    procedure mnuOptionFClick(Sender: TObject);
    procedure mnuOptionGClick(Sender: TObject);
    procedure mnuQuitClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  frmRec: TfrmRec;
implementation
{$R *.dfm}
//=====
procedure TfrmRec.mnuOptionAClick(Sender: TObject);
begin
  qryRec.Close;
  qryRec.SQL.Text:='SELECT * FROM tblCarnivores ORDER BY FamilyName,
ScientificName';
  qryRec.Open;
end;
//=====
procedure TfrmRec.mnuOptionBClick(Sender: TObject);
var
  sX : String;
begin
  sX := INPUTBOX('Question 1', 'Enter the family name', 'Canidae');
  qryRec.Close;
  qryRec.SQL.Text:= 'SELECT ScientificName, GeneralName, EnclosureNo,
EnclosureSize '+
    'FROM tblCarnivores '+
    'WHERE (FamilyName LIKE "%'+sX+'%" AND (EnclosureNo LIKE "ZE%"))';
  qryRec.Open;
end;
//=====
procedure TfrmRec.mnuOptionCClick(Sender: TObject);
begin

```

```

    qryRec.Close;
    qryRec.SQL.Text := 'SELECT Endangered, Count(NumAdults+NumYoung) AS
CountAnimals '+
                        'FROM tblCarnivores ' +
                        'GROUP BY Endangered';

    qryRec.Open;
end;
//=====
procedure TfrmRec.mnuOptionDClick(Sender: TObject);
begin
    qryRec.Close;
    qryRec.SQL.Text := 'SELECT EnclosureNo, Format((EnclosureSize/
(NumAdults+NumYoung)), "#.0#") AS SpacePerAnimal '+
                        'FROM tblCarnivores '+
                        'WHERE GeneralName LIKE "%mongoose%" ';

    qryRec.Open;
end;
//=====
procedure TfrmRec.mnuOptionEClick(Sender: TObject);
begin
    qryRec.Close;
    qryRec.SQL.Text := 'UPDATE tblCarnivores ' +
                        'SET NumYoung = NumYoung + 3 ' +
                        'WHERE EnclosureNo="ZF1"';

    qryRec.ExecSQL;
    MessageDlg('Record Processed Successfully', mtInformation, [mbOk], 0);
end;
//=====
procedure TfrmRec.mnuOptionFClick(Sender: TObject);
var
    sX : String;
begin
    sX := INPUTBOX('Question 1', 'Enter the day of the month when the visit took
place e.g. 23', '23');
    qryRec.Close;
    qryRec.SQL.Text:='SELECT C.EnclosureNo, GeneralName, VisitDate,
ReasonForVisit, Animal_ID ' +
                    'FROM tblCarnivores C, tblVetVisits V ' +
                    'WHERE Day(VisitDate)='+sX+' AND C.EnclosureNo = V.EnclosureNo';
    qryRec.Open;
end;
//=====
procedure TfrmRec.mnuOptionGClick(Sender: TObject);
begin
    qryRec.Close;
    qryRec.SQL.Text := 'INSERT INTO tblVetVisits ' +
                        '(VisitDate, EnclosureNo, ReasonForVisit, FollowUp,
Animal_ID) '+
                        'VALUES (#2012/10/25#, "ZD5", "Ear infection", True,
"ZD5_3")';

    qryRec.ExecSQL;
    MessageDlg('Record Processed Successfully', mtInformation, [mbOk], 0);
end;
//=====
procedure TfrmRec.mnuQuitClick(Sender: TObject);
begin
    Application.Terminate;
end;

end.

```

ANNEXURE E: SOLUTION FOR QUESTION 2: DELPHI**QUEST2 CLASS UNIT**

```
unit uQuest2_Memo;
{*** Solution for class unit of question 2 ***}
interface

TYPE
  TQuest2 = class(TObject)
  private
    fAType      : String;
    fNumber     : Integer;
    fSize       : Real;
    fCat        : Char;
  public
    constructor create(sAType: String;iNum: integer;rSize: Real;cCat: Char);
    function toString:String;
    function isSuitable(cCat:char; iNumber:integer):Boolean;
    procedure setAType(sAType : String);
    procedure setNumber(iNumber : Integer);
    procedure setSize(rSize : Real);
    procedure setCat(cCat : Char);
    function getAType:String;
    function getNumber:integer;
    function getSize:real;
    function getCat:Char;
  end;

implementation

uses SysUtils;

{ TQuest2 }

constructor TQuest2.create(sAType: String;iNum: integer;rSize: Real;cCat:
Char);
begin
  fAType := sAType;
  fNumber := iNum;
  fSize := rSize;
  fCat := cCat;
end;

function TQuest2.isSuitable(cCat:char; iNumber:integer):Boolean;
var
  rSpace :real;
begin
  Result := false;
  if fAType = 'XXX' then
  begin
    rSpace := fSize / iNumber;
    case cCat of
      'L': Result := rSpace >= 18;
      'M': Result := (rSpace >= 12) and (rSpace < 18);
      'S' : Result := (rSpace >= 7) and (rSpace < 12);
    end;
  end;
end;

function TQuest2.toString:String;
begin
```

```
    Result := fAType + '...' + fCat + #13 + 'Enclosure size: ' +
      FloatToStrF(fSize, ffFixed, 8,1) + #13 + 'Number of animals: ' +
      IntToStr(fNumber) + #13 + #13;
end;

procedure TQuest2.setAType(sAType: String);
begin
    fAType := sAType;
end;

procedure TQuest2.setSize(rSize: Real);
begin
    fSize := rSize;
end;

procedure TQuest2.setCat(cCat: Char);
begin
    fCat := cCat;
end;

procedure TQuest2.setNumber(iNumber: Integer);
begin
    fNumber := iNumber;
end;

function TQuest2.getAType:String;
begin
    Result := fAType;
end;

function TQuest2.getNumber:integer;
begin
    Result := fNumber;
end;

function TQuest2.getSize:real;
begin
    Result := fSize;
end;

function TQuest2.getCat:Char;
begin
    Result := fCat;
end;

end.
```

MAIN FORM UNIT

```
unit Question2U_Memo;
{*** Solution for main unit of question 2 ***}

interface

uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, StdCtrls, ComCtrls, Menus,
    uQuest2_Memo;

type
    TfrmQ2 = class(TForm)
        mnuMain: TMainMenu;
        mnuOptionA: TMenuItem;
```

```

    mnuQuit: TMenuItem;
    redQ2: TRichEdit;
    mnuOptionB: TMenuItem;
    procedure mnuQuitClick(Sender: TObject);
    procedure mnuOptionbClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure mnuOptionAClick(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    frmQ2: TfrmQ2;

implementation

var
    EnclosuresArr :array[1..30] of TQuest2;
    iCount :integer;

{$R *.dfm}
{$R+}

procedure TfrmQ2.FormCreate(Sender: TObject);
var
    TFile : TextFile;
    iPos, iNumber : integer;
    rSize :real;
    cCat :Char;
    sLine, sAnimal :String;
begin
    if FileExists ('DataQ2.txt') <> true then
        begin
            ShowMessage('File does not exist');
            Exit;
        end;
    AssignFile(TFile, 'DataQ2.txt');
    Reset(TFile);

    iCount := 0;
    while NOT EOF(TFile) AND (iCount < 30) do
        begin
            inc(iCount);
            readln(TFile, sLine);
            iPos := pos(';', sLine);
            sAnimal := copy(sLine, 1, iPos -1);
            delete(sLine, 1, iPos);

            iPos := pos('#', sLine);
            iNumber := StrToInt(copy(sLine, 1, iPos -1));
            delete(sLine, 1, iPos);

            iPos := pos(';', sLine);
            rSize := StrToFloat(copy(sLine, 1, iPos -1));
            delete(sLine, 1, iPos);

            cCat := sLine[1];

            EnclosuresArr[iCount] := TQuest2.create(sAnimal, iNumber, rSize, cCat);
        end;
    closeFile(TFile);

```

```
end;

procedure TfrmQ2.mnuOptionAClick(Sender: TObject);
var
  K :integer;
begin
  redQ2.Lines.Add('List of all enclosures');
  redQ2.Lines.Add('=====');
  For K := 1 to iCount do
    begin
      redQ2.Lines.Add('Enclosure number: ' + IntToStr(K) + #13 +
EnclosuresArr[K].toString);
    end;
  end;

procedure TfrmQ2.mnuOptionBClick(Sender: TObject);
var
  K,iNum :integer;
  bFound :boolean;
  cCat   :char;
  sAType :String;
begin
  sAType := InputBox('Animal type', 'Enter the type of animal for example
Tiger','Tiger');
  iNum := StrToInt(InputBox('Number of animals', 'Enter the number of
animals','2'));
  cCat := InputBox('Category', 'Enter the category (L/M/S)', 'L')[1];
  bFound := false;
  K := 1;
  While (bFound <> true) and (K <= iCount) do
    begin
      if EnclosuresArr[K].isSuitable(cCat, iNum) then
        begin
          EnclosuresArr[K].setAType(sAType);
          EnclosuresArr[K].setCat(cCat);
          EnclosuresArr[K].setNumber(iNum);
          bFound := true;
        end
      else
        inc(K);
      end;
    redQ2.Lines.Clear;
    if NOT(bFound) then
      redQ2.Lines.Add('No suitable enclosure was found')
    else
      begin
        redQ2.Lines.Clear;
        redQ2.Lines.Add('These animals were placed in enclosure number ' +
IntToStr(K));
        redQ2.Lines.Add(' ');
        mnuOptionA.Click;
      end;
  end;

procedure TfrmQ2.mnuQuitClick(Sender: TObject);
begin
  Application.Terminate;
end;
end.
```

ANNEXURE F: SOLUTION FOR QUESTION 3: DELPHI

```

unit Question3U_MEMO;
//Solution for Question 3...
interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Buttons, StdCtrls, ComCtrls, Menus;

type
  TfrmQuestion3 = class(TForm)
    MainMenu: TMainMenu;
    mnuOptionA: TMenuItem;
    mnuOptionB: TMenuItem;
    mnuOptionC: TMenuItem;
    mnuQuit: TMenuItem;
    redQ3: TRichEdit;
    procedure mnuOptionAClick(Sender: TObject);
    procedure mnuQuitClick(Sender: TObject);
    procedure mnuOptionBClick(Sender: TObject);
    procedure mnuOptionCClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmQuestion3: TfrmQuestion3;
  arrTic : Array[1..32] of string =
    ('RCm158', 'ADM33', 'RCf250', 'RAf5', 'RRM32', 'Adm236', 'RCm23', 'RDM54',
     'RCf17', 'RAm12', 'Adm9', 'RCF43', 'RDm140', 'RDm23', 'ACF113', 'ABf30',
     'RDm22', 'ARf38', 'RCF8', 'RAf53', 'RCf12', 'ABF156', 'ADM31', 'ADM47',
     'RAf48', 'ABF246', 'ABf59', 'RRM321', 'ABm36', 'RCF31', 'RAm445', 'ACn26');

implementation

{$R *.dfm}
{$R+}

var
  //arrays used in solution for Option C
  arrDisplay : Array[1..8] of string =
    ('AA', 'AB', 'AC', 'AD', 'RA', 'RB', 'RC', 'RD');
  arrPoints : Array[1..8] of Integer = (0,0,0,0,0,0,0,0);
  arrMedal : Array[1..3] of string = ('Gold', 'Silver', 'Bronze');

procedure TfrmQuestion3.FormCreate(Sender: TObject);
begin
  Randomize;
end;

procedure TfrmQuestion3.mnuOptionAClick(Sender: TObject);
var
  A : Integer;
begin
  redQ3.Lines.Clear;
  redQ3.Lines.Add('Invalid entries:');
  For A := 1 to 32 do
    IF (arrTic[A][1] in ['A','R']) AND
       (arrTic[A][2] in ['A'..'D']) AND

```

```

    (upCase(arrTic[A][3]) in ['M','F'])
  then      //valid ticket
  else
    begin //invalid ticket
      redQ3.Lines.Add(arrTic[A]);
      arrTic[A] := 'Z';
    end;
end;

procedure TfrmQuestion3.mnuOptionBClick(Sender: TObject);
var
  iTicket :integer;
  bValid  :boolean;
begin
  bValid := false;
  iTicket := random(32) + 1;
  while (bValid = false) do
    begin
      if arrTic[iTicket] = 'Z' then
        begin
          bValid := false ;
          redQ3.Lines.Add('Invalid');
          iTicket := random(32) + 1;
        end
      else
        bValid := true;
      end;
    redQ3.Lines.Add('The position of the winning ticket in the array: ' +
      intToStr(iTicket));
    redQ3.Lines.Add('The winning ticket: ' + arrTic[iTicket] );
  end;

procedure TfrmQuestion3.mnuOptionCClick(Sender: TObject);
var
  A, D, iPoint, iTemp : Integer;
  sTemp                : string;
begin
  //For each display calculate the number of points:
  for D := 1 to 8 do
    arrPoints[D] := 0;

  for D := 1 to 8 do
    begin
      For A := 1 to 32 do
        begin
          IF pos(arrDisplay[D], arrTic[A]) = 1 //only valid tickets <> Z
          then
            begin
              case arrTic[A][3] of
                'm', 'f' : iPoint := 5;
                'M', 'F' : iPoint := 12;
              end;
              inc(arrPoints[D], iPoint);
            end;
          end; //for A
        end; //for D

  //Sort the arrays according to points >> Any sorting method
  For A := 1 to 8-1 do
    For D := a+1 to 8 do
      IF arrPoints[A] < arrPoints[D]
      then
        begin //swop elements of both arrays

```



```
sTemp           := arrDisplay[D];
arrDisplay[D]   := arrDisplay[A];
arrDisplay[A]   := sTemp;

iTemp           := arrPoints[D];
arrPoints[D]    := arrPoints[A];
arrPoints[A]    := iTemp;
end;

//Display results
redQ3.Lines.Clear;
redQ3.Paragraph.TabCount := 2;
redQ3.Paragraph.Tab[0]   := 80;
redQ3.Paragraph.Tab[1]   := 150;
redQ3.Lines.Add('Medal winning displays:');
redQ3.Lines.Add('Medal' + #9 + 'Display' + #9 + 'Points');
for A := 1 to 3 do
  redQ3.Lines.Add(arrMedal[A] + #9+ arrDisplay[A]+#9+IntToStr(arrPoints[A]));
end;

procedure TfrmQuestion3.mnuQuitClick(Sender: TObject);
begin
  Application.Terminate;
end;

end.
```

ANNEXURE G: SOLUTION FOR QUESTION 1: JAVA

```
//Solution for Question 1...
import java.io.*;
import java.sql.*;
import javax.swing.*;
import java.util.Scanner;

public class TestQuestion1_Memo
{
    public static void main (String[] args) throws SQLException,IOException
    {

        BufferedReader inKb = new BufferedReader (new InputStreamReader
(System.in));

        Zoo DB = new Zoo();
        System.out.println();

        char choice = ' ';
        do
        {
            System.out.println("\n\n      MENU");
            System.out.println();
            System.out.println("      Option A");
            System.out.println("      Option B");
            System.out.println("      Option C");
            System.out.println("      Option D");
            System.out.println("      Option E");
            System.out.println("      Option F");
            System.out.println("      Option G");
            System.out.println();
            System.out.println("      Q - QUIT");
            System.out.println(" ");
            System.out.print("      Your choice? ");
            choice = inKb.readLine().toUpperCase().charAt(0);
            System.out.println(" ");
            String sql = "";
            switch(choice)
            {
                case 'A':          // Question 1.1
                    {
                        sql = "SELECT * FROM tblCarnivores ORDER BY FamilyName,
ScientificName";
                        DB.query(sql);
                        break;
                    }
//=====
                case 'B':          // Question 1.2
                    {
                        System.out.println("Enter the family name, e.g. Canidae");
                        String sX = inKb.readLine();
                        sql = "SELECT ScientificName, GeneralName,
EnclosureNo,EnclosureSize FROM tblCarnivores WHERE EnclosureNo LIKE 'ZE%' AND
FamilyName = '" + sX + "'";

                        DB.query(sql);
                        break;
                    }
//=====
                case 'C':          // Question 1.3
                    {
```

```

        sql = "SELECT Endangered, Count(NumAdults+NumYoung) AS
CountAnimals FROM tblCarnivores GROUP BY Endangered";
        DB.query(sql);
        break;
    }
//=====
    case 'D':        // Question 1.4
    {
        sql = "SELECT EnclosureNo,
Format(EnclosureSize/(NumAdults+NumYoung), '#.0#') AS SpacePerAnimal FROM
tblCarnivores WHERE GeneralName LIKE '%mongoose%'";
        DB.query(sql);
        break;
    }
//=====
    case 'E':        // Question 1.5
    {
        sql = "UPDATE tblCarnivores SET NumYoung = NumYoung + 3
WHERE EnclosureNo = 'ZF1'";

        DB.query(sql);
        break;
    }
//=====
    case 'F':        // Question 1.6
    {
        System.out.println("Enter the day of the month when the
first visit took place (for example 23)");
        String sX = inKb.readLine();
        sql = "SELECT tblCarnivores.EnclosureNo, GeneralName,
VisitDate, ReasonForVisit, Animal_ID FROM tblCarnivores, tblVetVisits WHERE
tblCarnivores.EnclosureNo = tblVetVisits.EnclosureNo AND Day(VisitDate) = "+
sX;

        DB.query(sql);
        break;
    }
//=====
    case 'G':        // Question 1.7
    {
        sql = "INSERT INTO tblVetVisits
(VisitDate,EnclosureNo,ReasonForVisit,FollowUp,Animal_ID) VALUES (#2012/10/25#,
'ZD5', 'Ear infection', true,'ZD5_3')";

        DB.query(sql);
        break;
    }
}
}while (choice != 'Q');

DB.disconnect();
System.out.println("Done");
}
}

```

ANNEXURE H: SOLUTION FOR QUESTION 2: JAVA**QUEST2 CLASS UNIT**

```
//Solution for Question 2 class unit...
public class Quest2Memo
{
    private String type;
    private int number;
    private double size;
    private char cat;

    public Quest2Memo(String type, int number, double size, char cat)
    {
        this.type = type;
        this.number = number;
        this.size = size;
        this.cat = cat;
    }

    public boolean isSuitable(char cat, int number)
    {
        boolean suitable = false;
        double space = 0;
        if (type.equalsIgnoreCase("XXX"))
        {
            space = size / number;

            if (cat == 'L' && space >= 18)
                suitable = true;

            if (cat == 'M' && (space >= 12 && space < 18))
                suitable = true;

            if (cat == 'S' && (space >= 7 && space < 12))
                suitable = true;
        }
        return suitable;
    }

    public String toString()
    {
        return type + "... " + cat + "\nEnclosure size: " + size + "\nNumber
of animals: " + number + "\n\n";
    }
    public void setAType(String type) {
        this.type = type;
    }

    public void setNumber(int number) {
        this.number = number;
    }

    public void setSize(double size) {
        this.size = size;
    }

    public void setCat(char cat) {
        this.cat = cat;
    }

    public String getAType() {
        return type;
    }
}
```

```

    }

    public int getNumber() {
        return number;
    }

    public double getSize() {
        return size;
    }

    public char getCat() {
        return cat;
    }
}

```

TEST CLASS (DRIVER CLASS)

```

//Solution for Question 2 Test class...
import java.io.*;
import java.util.Scanner;

public class TestQuestion2_Memo {

    static Quest2Memo[] enclosures = new Quest2Memo[30];
    static int cnt;

    public static void main(String[] args) throws Exception
    {
        BufferedReader kb = new BufferedReader(new
InputStreamReader(System.in));
        readFile();

        char choice = ' ';
        do {
            System.out.println("    MENU");
            System.out.println();
            System.out.println(" Option A");
            System.out.println(" Option B");
            System.out.println();
            System.out.println(" Q - QUIT");
            System.out.println("\n Your choice? ");
            choice =kb.readLine().charAt(0);
            switch (choice) {
                case 'A':
                    display();
                    break;
                case 'B':
                    optionB();
                    break;
                case 'Q':
                    System.out.println("Quit");
            }
        } while (choice != 'Q');
    }

    public static void readFile() {
        try
        {
            cnt = 0;
            Scanner sc = new Scanner (new FileReader("DataQ2.txt"));
            while (sc.hasNext())
            {

```

```
String line = sc.nextLine();
int pos1 = line.indexOf(';',0);
String aType = line.substring(0,pos1);

int posHash = line.indexOf('#',pos1);
int numberAn = Integer.parseInt(line.substring(pos1 +
1,posHash));

int pos2 = line.indexOf(';',posHash);
double size = Double.parseDouble(line.substring(posHash +
1,pos2));

int posHash2 = line.indexOf('#',pos2);
char cat = line.charAt(posHash2-1);

enclosures[cnt] = new Quest2Memo(aType, numberAn, size, cat);
cnt++;
}
sc.close();
}
catch (FileNotFoundException e) {
    System.out.println("File does not exist");
    System.exit(0);
}
catch (Exception f) {
    System.out.println(f);
}
}

public static void display() {
    System.out.println("List of all enclosures");
    System.out.println("=====");

    for (int k = 0; k < cnt; k++) {
        System.out.println("Enclosure number: " + (k+1)+"\n" +
enclosures[k].toString());
    }
}

private static void optionB() throws Exception {
    BufferedReader kb = new BufferedReader(new
InputStreamReader(System.in));

    boolean found = false;
    int count = 0;

    System.out.println("Enter the type of animal (for example Tiger)");
    String animal = kb.readLine();
    System.out.println("Enter the number of animals, e.g. 2");
    int numA = Integer.parseInt(kb.readLine());
    System.out.println("Enter the category (L/M/S)");
    char cat = kb.readLine().charAt(0);

    while (found == false && count < cnt)
    {
        if (enclosures[count].isSuitable(cat, numA))
        {
            found = true;
            enclosures[count].setAType(animal);
            enclosures[count].setNumber(numA);
            enclosures[count].setCat(cat);
        }
        else

```

```
        count++;
    }
    if (found == false)
    {
        System.out.println("No suitable enclosure was found");
    }
    else
    {
        System.out.println("\n\nThese animals were placed in enclosure
number " + (count +1 ));
        System.out.println("\n");
        display();
    }
}
}
```

ANNEXURE I: SOLUTION FOR QUESTION 3 WITH OOP: JAVA

```
import java.io.IOException;

public class TestQuestion3_Memo
{
    public static void main(String[] args) throws IOException
    {
        Question3_Memo test = new Question3_Memo();
        test.displayMenu();
    }
}

// Object class describing a Ticket object
public class Ticket
{
    private String section;
    private String display;
    private String gender;
    private int number;

    public Ticket(String ticket)
    {
        section = ticket.substring(0,1);
        display = ticket.substring(1,2);
        gender = ticket.substring(2,3);
        number = Integer.parseInt(ticket.substring(3,4));
    }

    public boolean isValid()
    {
        boolean valid = true;

        if (("ABCD".indexOf(display.charAt(0)) < 0) || (!(section.equals("A")) &&
        (!(section.equals("R")))) || ("MmFf".indexOf(gender.charAt(0)) < 0 ))
            valid = false;
        return valid;
    }

    public int getPointvalue()
    {
        if (gender.equals("M") || gender.equals("F") )
            return 12;
        else
            return 5;
    }

    public String getSection()
    {
        return section;
    }

    public String getDisplay()
    {
        return display;
    }

    public String getGender()
    {
        return gender;
    }
}
```



```

public int getNumber()
{
    return number;
}
}
//=====

import java.io.IOException;
import java.io.BufferedReader;
import java.io.InputStreamReader;

public class Question3_Memo
{
    String[] arrTic = {"RCm158", "ADM33", "RCf250", "RAf5",
        "BRM32", "ADm236", "RCm23", "RDM54",
        "RCf17", "RAm12", "ADm9", "RCF43",
        "RDm140", "RDm23", "ACF113", "ABf30",
        "RDm22", "ARf38", "RCF8", "RAf53",
        "RCf12", "ABF156", "ADM31", "ADM47",
        "RAf48", "ABF246", "ABf59", "RRM321",
        "ABm36", "RCF31", "RAM445", "ACn26"}; // size 32

    String[] arrDisplay = {"AA", "AB", "AC", "AD", "RA", "RB", "RC", "RD"};
    int[] arrPoints = {0,0,0,0,0,0,0,0};
    String[] arrMedal = {"Gold", "Silver", "Bronze"};

    BufferedReader inKb;
//=====
// Option A
    public void validate()
    {
        System.out.println("Invalid entries");
        for (int count = 0; count < 32; count++) {

            Ticket ticket = new Ticket(arrTic[count]);
            if (ticket.isValid() == false) {

                System.out.println(arrTic[count]);
                arrTic[count] = "Z";
            }
        }
    }
//=====
//Option B
    public void getWinningNumber()
    {
        boolean valid = false;
        int win = (int)(Math.random() * 32);
        while (valid == false)
        {
            if(arrTic[win].equals("Z"))
            {
                win = (int)(Math.random() * 32);
                System.out.println("Invalid");
            }
            else
                valid = true;
        }
        System.out.println("The position of the winning ticket in the array: "
+ (win+1));
        System.out.println("The winning ticket: " + arrTic[win]);
    }
}

```

```
//=====
// Option C
// Identify Medal Winners
public void getMedalWinners() throws IOException
{
    //For each display add points:
    for (int d = 0; d < 8; d ++){
        for (int t = 0; t < 32; t++){
            if( !(arrTic[t].equalsIgnoreCase("Z"))){

                Ticket ticket = new Ticket(arrTic[t]);

                String displayChoice = ticket.getSection() +
                    ticket.getDisplay();

                if (displayChoice.equalsIgnoreCase(arrDisplay[d])) {
                    arrPoints[d]= arrPoints[d] + ticket.getPointvalue();
                } //if
            } // if not Z
        } // for ticket
    } // for d

    //Sort the two arrays
    for (int a = 0; a < 8 -1; a++) {
        for (int d = (a+1); d < 8; d++) {
            if ( arrPoints[a] < arrPoints[d]){

                String tempD = arrDisplay[a];
                arrDisplay[a]= arrDisplay[d];
                arrDisplay[d] = tempD;

                int tempP = arrPoints[a];
                arrPoints[a] = arrPoints[d];
                arrPoints[d] = tempP;
            } // if
        } // for d
    } // for a

    //display medals
    System.out.println("Medal winning displays:");
    System.out.printf("%s%20s%20s\n", "Medal", "Display", "Points");
    for (int a = 0; a < 3; a ++){
        System.out.printf("%-8s%12s%21d\n", arrMedal[a],
            arrDisplay[a],arrPoints[a]);
    }
} // getMedalWinners

public void displayMenu() throws IOException
{
    inKb = new BufferedReader (new InputStreamReader (System.in));
    System.out.println();

    char choice = ' ';
    do
    {
        System.out.println("\n\n          MENU");
        System.out.println();
        System.out.println("          Option A");
        System.out.println("          Option B");
        System.out.println("          Option C");
        System.out.println();
        System.out.println("          Q - QUIT");
        System.out.println(" ");
    }
}

```

```
System.out.print("    Your choice? ");
choice = inKb.readLine().toUpperCase().charAt(0);
System.out.println(" ");
String sql = "";
switch(choice)
{
    case 'A':
        validate();
        break;
    case 'B':
        getWinningNumber();
        break;
    case 'C':
        getMedalWinners();
        break;
    case 'Q':
        System.out.println("QUIT");
        break;
}
}while(choice != 'Q');
}
}
```

ANNEXURE J: SOLUTION FOR QUESTION 3 WITHOUT OOP: JAVA

```

import java.io.IOException;
import java.io.BufferedReader;
import java.io.InputStreamReader;

public class Question3_Memo
{
    String[] arrTic = {"RCm158", "ADm33", "RCf250", "RAF5",
        "RRM32", "ADm236", "RCm23", "RDM54",
        "RCf17", "RAm12", "ADm9", "RCF43",
        "RDm140", "RDm23", "ACF113", "ABf30",
        "RDm22", "ARf38", "RCF8", "RAf53",
        "RCf12", "ABF156", "ADM31", "ADM47",
        "RAf48", "ABF246", "ABf59", "RRM321",
        "ABm36", "RCF31", "RAm445", "ACn26"};

    //arrays used in solution for Option C
    String[] arrDisplay = {"AA", "AB", "AC", "AD", "RA", "RB", "RC", "RD"};
    int[] arrPoints = {0,0,0,0,0,0,0,0};
    String[] arrMedal = {"Gold", "Silver", "Bronze"};

    BufferedReader inKb;

    //=====
    // Option A
    public void validate()
    {
        System.out.println("Invalid entries");
        for (int c = 0; c < 32; c++)
        {
            char firstchar = arrTic[c].charAt(0);
            char secondchar = arrTic[c].charAt(1);
            // or String secondchar = arrTic[c].substring(1,2);
            char thirdchar = arrTic[c].charAt(2);

            if ("ABCD".indexOf(secondchar) < 0 || (firstchar != 'A' && firstchar
            != 'R' ) || ("MmFf".indexOf(thirdchar) < 0 ))
            {
                System.out.println(arrTic[c]);
                arrTic[c] = "Z";
            }
        }
    }
    //=====
    //Option B
    public void getWinningNumber()
    {
        boolean valid = false;
        int win = (int)(Math.random() * 32);
        while (valid == false)
        {
            if(arrTic[win].equals("Z"))
            {
                win = (int)(Math.random() * 32);
                System.out.println("Invalid");
            }
            else
                valid = true;
        }
        System.out.println("The position of the winning ticket in the array: "
        + (win+1));
        System.out.println("The winning ticket: " + arrTic[win]);
    }
}

```

}

```
//=====
// Option C
// Identify Medal Winners
public void getMedalWinners() throws IOException
{
    // Write code for Option C
    //For each display add points:
    for (int d = 0; d < 8; d++) {
        for (int t = 0; t < 32; t++) {
            if( !(arrTic[t].equalsIgnoreCase("Z"))){
                String displayChoice = arrTic[t].substring(0,2);
                if (displayChoice.equalsIgnoreCase(arrDisplay[d])) {
                    char gender = arrTic[t].charAt(2);
                    if (gender == 'f' || gender == 'm')
                        arrPoints[d]= arrPoints[d] + 5;
                    else
                        arrPoints[d] = arrPoints[d]+ 12;
                } //if
            } // if not Z
        } // for t
    } // for d

    //Sort the two arrays
    for (int a = 0; a < 8 -1; a++) {
        for (int d = (a+1); d < 8; d++) {
            if ( arrPoints[a] < arrPoints[d]){

                String tempD = arrDisplay[a];
                arrDisplay[a]= arrDisplay[d];
                arrDisplay[d] = tempD;

                int tempP = arrPoints[a];
                arrPoints[a] = arrPoints[d];
                arrPoints[d] = tempP;
            } // if
        } // for d
    } // for a
    //display medals
    System.out.println("Medal winning displays:");
    System.out.printf("%s%20s%20s\n", "Medal", "Display", "Points");
    for (int a = 0; a < 3; a++) {
        System.out.printf("%-8s%12s%21d\n", arrMedal[a], arrDisplay[a],
            arrPoints[a]);
    }
} // getMedalWinners

public Question3_Memo() throws IOException
{
    inKb = new BufferedReader (new InputStreamReader (System.in));

    System.out.println();

    char choice = ' ';
    do
    {
        System.out.println("\n\n          MENU");
        System.out.println();
        System.out.println("          Option A");
        System.out.println("          Option B");
        System.out.println("          Option C");
        System.out.println();
    }
}

```

```
System.out.println("    Q - QUIT");
System.out.println(" ");
System.out.print("    Your choice? ");
choice = inKb.readLine().toUpperCase().charAt(0);
System.out.println(" ");
String sql = "";
switch(choice)
{
    case 'A':
        validate();
        break;
    case 'B':
        getWinningNumber();
        break;
    case 'C':
        getMedalWinners();
        break;
    case 'Q':
        System.out.println("QUIT");
        break;
}
}while(choice != 'Q');
}

public static void main(String[] args) throws IOException
{
    new Question3_Memo();
}
}
```



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**NATIONAL
SENIOR CERTIFICATE**

GRADE 12

INFORMATION TECHNOLOGY P2

NOVEMBER 2012

MARKS: 180

TIME: 3 hours

This question paper consists of 21 pages.

INSTRUCTIONS AND INFORMATION

1. This question paper consists of FIVE sections which are subdivided as follows:

SECTION A:	Multiple-choice questions	(10)
SECTION B:	Hardware and software	(50)
SECTION C:	Applications and implications	(25)
SECTION D:	Programming and software development	(48)
SECTION E:	Integrated scenario	(47)

2. Answer ALL the questions.
3. Read ALL the questions carefully.
4. The mark allocation generally gives an indication of the number of facts/reasons required.
5. Number the answers correctly according to the numbering system used in this question paper.
6. Write neatly and legibly.

SECTION A: MULTIPLE-CHOICE QUESTIONS**QUESTION 1**

Various options are given as possible answers to the following questions. Choose the answer and write only the letter (A–D) next to the question number (1.1–1.10) in the ANSWER BOOK.

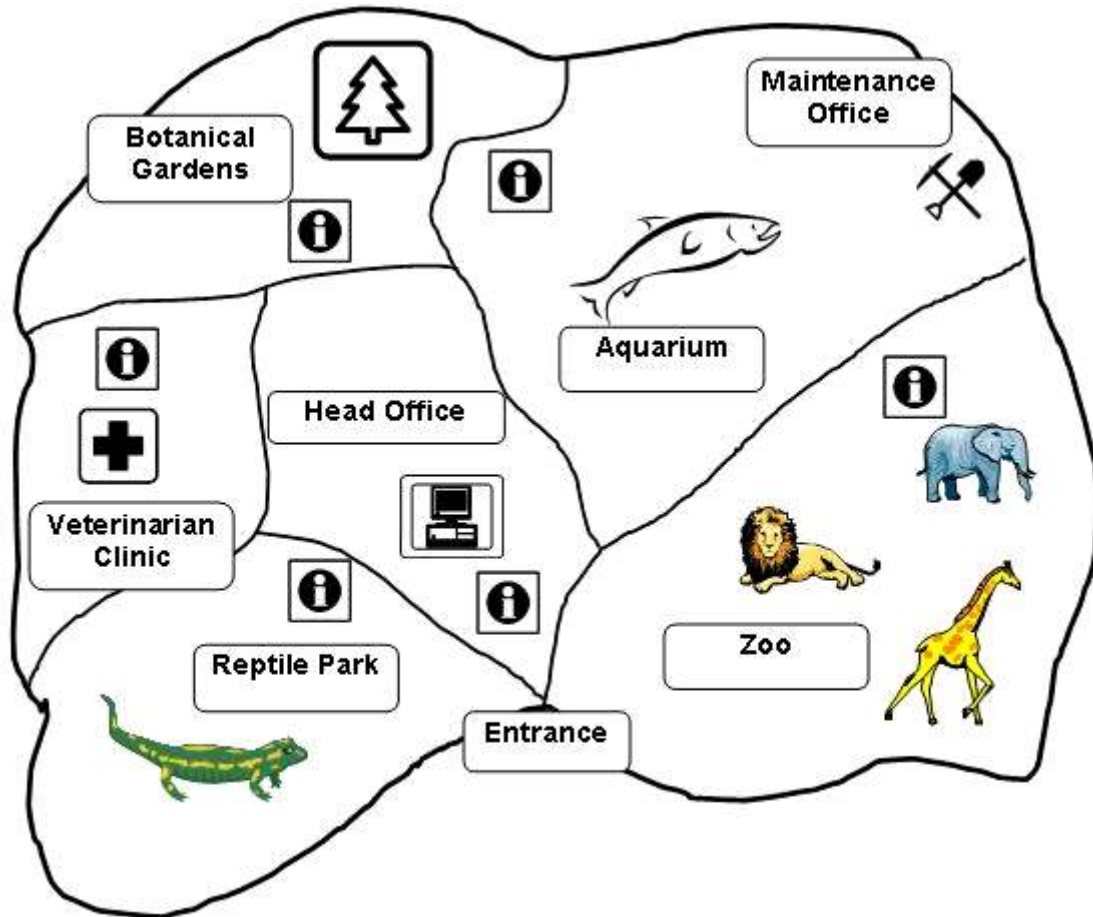
- 1.1 A software program that records what you type on the keyboard or the mouse clicks you make is called a ...
A worm.
B honeypot.
C keylogger.
D cookie. (1)
- 1.2 A ... is a shared library of instructions that can be used by different applications.
A GUI
B DLL
C RSS
D ADSL (1)
- 1.3 The process of data mining can be described as the ...
A deletion of large amounts of data from backup folders.
B analysis of large collections of data.
C formatting of old data files to take up less space.
D recovering of lost data on a disc. (1)
- 1.4 The switching technique where the best possible route is chosen to transfer data in a WAN is called ... switching.
A line
B packet
C circuit
D control (1)
- 1.5 ... is considered to be a low-level programming language.
A Java
B SQL
C An assembly language
D Delphi (1)
- 1.6 ... is non-volatile memory that can be used as secondary storage.
A SRAM
B DRAM
C Flash memory
D Cache memory (1)

- 1.7 Which operating system has been designed specifically to work on a cellphone?
- A Android
 - B Solaris
 - C Windows Vista
 - D Linux
- (1)
- 1.8 Software that translates each line of source code into machine code and executes it before moving to the next line of code is known as a/an...
- A assembler.
 - B translator.
 - C compiler.
 - D interpreter.
- (1)
- 1.9 The values below are saved into an array.
- | | | | | |
|---|---|---|---|---|
| a | e | i | o | u |
|---|---|---|---|---|
- The index value of the element 'a' in the array is 0. The index value of the element 'u' in the array will be ...
- A 0.
 - B 4.
 - C 5.
 - D 6.
- (1)
- 1.10 The ... is a special register in the ALU that temporarily keeps the result of any calculation.
- A memory-data register
 - B CMOS
 - C program counter
 - D accumulator
- (1)

TOTAL SECTION A: 10

SCENARIO

The Red Feather Nature Park consists of various smaller parks, including a reptile park, an aquarium, a zoo and a botanical garden as indicated in the diagram below.



Each smaller park has its own administration offices with computers that are linked to a server at the head office. The head office is situated near the main entrance to the Park. Most of the administrative work is done here.

In a new project to serve the community, learners from various schools volunteer to work at the Red Feather Nature Park during their holidays. Some of the learners will work with the animals, feeding them, cleaning the cages, et cetera. Some of the learners will receive a short training course as guides. Your expertise as an IT learner is needed in a group that is assigned to Mr Eagle, the Park's administration manager. Your group will help solve hardware-, software- and network-related problems.

Your group will also assist with new technology that will be implemented where cellphones can be used during guided tours. The official website of the Park (www.redpark.co.za) also needs some attention.

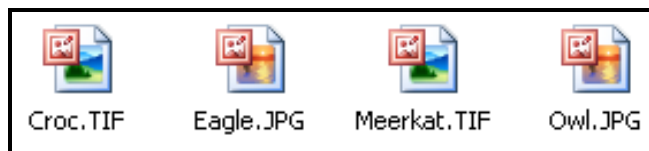
SECTION B: HARDWARE AND SOFTWARE**QUESTION 2**

- 2.1 One of the computers in the reptile park's offices needs to be upgraded. This computer among other things is used to upload photographs of animals at the Red Feather Nature Park to the website.
- 2.1.1 Currently a 17" CRT screen is in use with a refresh rate of 70 Hz.
- (a) Define the term *refresh rate*. (2)
 - (b) Give THREE reasons why it is recommended that the CRT screen should be replaced with an LCD screen. (3)
- 2.1.2 The hard disk contains mostly old photographs and is almost full.
- (a) All of the photographs, except those taken by the lead photographer at the Park, must be deleted. Fortunately metadata was set up.
 - (i) What is *metadata*? (2)
 - (ii) Give ONE example of metadata that is applicable to this situation. (1)
 - (b) The new trend in hard drive storage devices is to increase the amount of data that can be stored in the same amount of space. Name ONE way in which this can be achieved. (1)
- 2.1.3 Mr Eagle has been advised to replace the RAM of the computer in order to enhance its performance.
- (a) How will you ensure that the new RAM is compatible with the current motherboard? (1)
 - (b) Some of the learners in the group are of the opinion that, if there was enough virtual memory, there is no need to upgrade the RAM.
 - (i) Explain what *virtual memory* is. (2)
 - (ii) Explain why virtual memory will not be the solution to better performance. (1)
 - (c) The statement "*Upgrading RAM means more cache memory is available*" was made. Give THREE reasons why this statement is NOT true. (3)

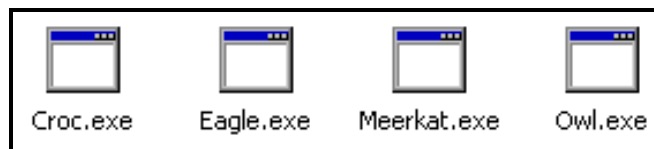
- 2.1.4 Most of the peripheral devices required will be connected to the computer using USB ports. Standard drivers will be used.
- Write down the full term for the acronym *USB*. (1)
 - Name ONE port other than the USB port that can be used to connect a digital video camera to a computer. (1)
 - Define a *driver* in the context of computing. (2)
 - What is meant by the term *standard driver*? (1)
 - Name ONE device that makes use of a standard driver. (1)

- 2.2 Mr Eagle is experiencing problems when trying to open photographs saved on the hard drive. He noticed that the icons of these files have been replaced. This suggests that the hard disk might have been infected by a virus.

Previous icons:



Current icons:



- 2.2.1 What is a *computer virus*? (2)
- 2.2.2 State TWO ways of preventing a virus from being launched on a computer system. (2)
- 2.2.3 The virus could not be removed by any of the latest antivirus programs. Therefore the hard drive needs to be formatted.

Write down the TWO missing words to correctly complete the following statement. Write only the answer next to the question number (2.2.3(a–b)).

During the process of physical formatting (a) ... and (b) ... are created on the disk. (2)

- 2.2.4 Currently all the photographs, letters and other documents, such as financial reports and order forms, are saved in the root folder on the hard drive.
- Draw a simple diagram of a folder structure to illustrate an improved structure for organising the files on a hard drive. Make provision for at least TWO levels of subfolders. (2)
- 2.3 The Red Feather Nature Park has created new feeding stations in the lion cage. Staff members need to monitor whether the lions are visiting both the old and the new feeding stations using RFID tags.
- 2.3.1 Write down the full term for the acronym *RFID*. (1)
- 2.3.2 Describe how RFID tags could be used to facilitate the monitoring of the lions. (2)
- 2.4 The server at the main office has to be replaced.
- Give TWO reasons why a server is better suited to ensure high performance rather than a regular PC. (2)
- 2.5 The chipset is an integral part of the design of a computer.
- 2.5.1 Briefly describe the primary function of the chipset. (1)
- 2.5.2 The following devices must be connected to the chipset. For each device indicate whether it will be connected to either the north bridge or south bridge.
- (a) USB (1)
- (b) PCI Express (1)
- (c) Network card (1)
- 2.6 The performance of the CPU should be optimised to ensure a high-performance computer system.
- 2.6.1 One of the techniques used to improve the performance of a CPU is pipelining. Define *pipelining*. (2)
- 2.6.2 Explain the role cache memory plays in the performance of a CPU. (2)
- 2.6.3 CPUs are often over-clocked to increase performance. Overheating is the major problem that arises because of this practice.
- (a) What is meant by *over-clocking a CPU*? (1)
- (b) Name TWO ways in which overheating can be prevented. (2)

2.7 Mr Eagle found the following information on GPS tours on the Internet.

A GPS tour (using the Global Positioning System) is an audio tour or a multimedia tour that provides pre-recorded spoken commentary, normally through a mobile device.

GPS tours can be created by using a combination of software and hardware and can be downloaded from the Internet for cellphones.

- 2.7.1 What type of communication connection is used in GPS technology? (1)
- 2.7.2 Give TWO practical examples of how GPS technology can be used at the park other than conducting GPS tours. (2)
- 2.7.3 Name a possible file format for an audio file that can be downloaded from the Internet onto any type of cellphone. (1)

TOTAL SECTION B: 50

SECTION C: APPLICATIONS AND IMPLICATIONS**QUESTION 3: e-COMMUNICATION**

Mr Eagle made the following statement: 'The Internet is an incredible communication tool, allowing new technology such as e-communication, e-banking and e-shopping.'

- 3.1 Many visitors requested that e-communication be in place as a standard facility at the park so that they can at least access their e-mails.
- 3.1.1 State TWO advantages of e-communication from the Park's point of view. (2)
- 3.1.2 State ONE disadvantage of having e-communication facilities at the Park from the visitors' point of view. (1)
- 3.2 The administration office requested that SSL should be implemented as an encryption protocol.
- 3.2.1 How will the Park's full website address (<http://www.redpark.co.za>) change once SSL is implemented? (1)
- 3.2.2 Briefly explain what *encryption* is. (2)
- 3.2.3 What is required to be able to decrypt an SSL-encrypted e-mail? (2)
- 3.2.4 Which ONE of the following activities will NOT need to make use of SSL encryption? Write only the answer (A–C) next to the question number (3.2.4).
- A The online information centre of the zoo responding to a visitor requesting the toll-free telephone number of the centre
- B A reply to the head of security indicating their new updated patrol routes
- C A person replying to a request for donations with personal details regarding his donation (1)
- 3.3 Give TWO practical and interesting examples of how podcasting can be used at the park. (2)
- 3.4 A new screensaver was downloaded from the Internet. Mr Eagle is concerned about the possible threat of a trojan.
- Explain what a *trojan* is. (2)

[13]

QUESTION 4: SOCIAL AND ETHICAL ISSUES

The Red Feather Nature Park has a code of ethics which enforces ethical behaviour within the organisation.

- 4.1 In terms of IT systems at the Park, describe TWO ways in which the electricity consumption could be reduced. (2)
- 4.2 The Park has a policy of replacing computers every five years.
- 4.2.1 State TWO responsible ways how the Park can dispose of their old computers. (2)
- 4.2.2 Do you think replacing computers every five years is a good policy? Give a reason for your answer. (2)
- 4.3 The administrator is concerned about sensitive financial data that has been saved on the hard disks of the computers they want to replace.
- State TWO measures that can be put in place to ensure that this data does not fall into the wrong hands. (2)
- 4.4 The human resources department is considering requesting access to the Facebook profiles of new employees in order to find out more about them.
- Do you think this is ethical? Justify your answer. (2)
- 4.5 The management at the Red Feather Nature Park is aware of the possibility of computer crime taking place in an e-communication environment.
- Explain the term *computer crime*. (2)

[12]**TOTAL SECTION C: 25**

SECTION D: PROGRAMMING AND SOFTWARE DEVELOPMENT**QUESTION 5: ALGORITHMS AND PLANNING**

The development of the custom software required by the Park has been outsourced. As a member of the IT committee you are requested to assist the outsourced company with the planning and design of the required software.

- 5.1 A member of the panel has compiled a database called **PlantsDB** containing data of plants the Park needs to buy and nurseries supplying these plants. Various types of plants are required, such as trees, shrubs and creepers, et cetera. One of the tables in the database is the **tblNursery** table with the following layout:

tblNursery			
Key	Field Name	Data Type	Description
🔑	CatalogueNumber	Text	Unique number
	BotanicalName	Text	Botanical name of the plant
	GeneralName	Text	Name of the plant
	PoisonousToAnimals	Number	Is the plant poisonous to animals?
	FertiliserType	Text	Type of fertiliser suitable for the plant
	FertiliserCost	Text	Cost of the fertiliser
	Nursery	Text	Name of the nursery supplying the plant
	NurseryPhoneNumber	Number	Contact number of the nursery supplying the plant
	NurseryContactPerson	Text	Name of the contact person at the nursery supplying the plant

- 5.1.1 Not much effort has been put into selecting suitable data types for the fields in the table. Suggest a *more suitable data type* for each of the following fields:
- (a) **FertiliserCost** (1)
- (b) **NurseryPhoneNumber** (1)
- 5.1.2 The preferred data type for the field **PoisonousToAnimals** is Boolean. Explain why the data type Number can be used as a suitable alternative. (1)
- 5.1.3 Only the letters K, M or U are allowed to be entered in the **FertiliserType** field. The entered data into this field needs to be validated.
- (a) Write down the condition that must be used to apply validation to this field. (3)
- (b) Give a suitable error message that should be displayed when invalid data is entered. (1)

5.2 Another table **tblTreeOrders** is used to store data on the orders placed for trees. The table has not been normalised.

tblTreeOrders			
Key	Field Name	Data Type	Description
	OrderNo	Text	Number of the order
🔑	OrderDate	Date/Time	Date the order was placed
	Tree1	Text	Name of the tree
	Description1	Text	Description of the tree
	Quantity1	Number	Quantity of the trees that were ordered
	CostPrice1	Currency	Cost price of the tree
	Tree2	Text	Name of the tree
	Description2	Text	Description of the tree
	Quantity2	Number	Quantity of the trees that were ordered
	CostPrice2	Currency	Cost price of the tree
	TotalAmount	Currency	Total monetary value of this order

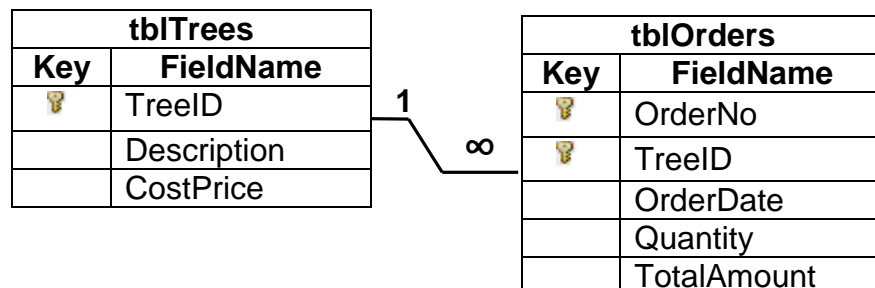
5.2.1 *Insert anomalies* is one of the problems that can occur when the table has not been normalised.

The following orders are placed separately but on the same day. In each case, give a reason why the insert anomaly will occur when the data has to be captured in the **tblTreeOrders** table.

(a) In the morning the manager orders three different kinds of trees using order number AB230. (2)

(b) In the afternoon another two trees need to be ordered using order number AB245. (2)

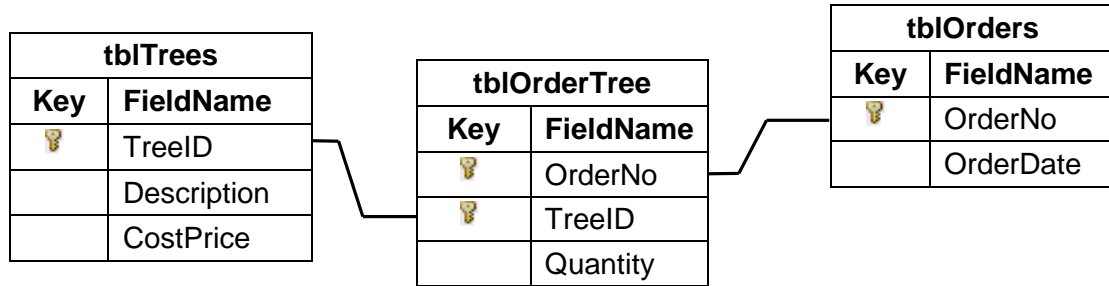
5.2.2 The following is a possible solution for normalising the **tblTreeOrders** table into 1NF. The **tblOrders** table makes use of a combined primary key to refer to records in the table.



(a) What is a *combined primary key*? (2)

- (b) Give TWO reasons why the use of a combined primary key is required in this table. (2)
- (c) One of the aims of second normal form is to prevent partial dependencies. Define the term *partial dependency*. (1)

5.2.3 The following is an attempt to normalise the given **tblTreeOrders** table into 2NF.



- (a) Identify the type of relationship between the **tblOrders** table and the **tblOrderTree** table in the diagram. (1)
- (b) Give ONE example of a field that can act as the foreign key in the **tblOrderTree** table. (1)
- (c) The **TotalAmount** field does not appear in any of the 2NF tables. Will the user still be able to determine the total price of an order? Give a brief explanation to substantiate your answer. (2)

5.3 The cleaning staff gets paid per week at a rate of R100,00 per enclosure they clean. The software developer has compiled the following algorithm to calculate and display the payment of each cleaner. A program has been developed based on this algorithm.

```

Line
1   Name ← input the name of the cleaner
2   Pay ← 0
3   Loop for each of the seven days
4     Total ← 0
5     Number ← Input the number of enclosures cleaned
6     Total ← Total + Number
7     Pay ← Number * 100
8   End loop
9   Display the name, number of enclosures cleaned and
   the payment the cleaner receives
    
```

5.3.1 Define an *algorithm*. (2)

- 5.3.2 In line 5 the user is required to enter a number which is saved into the integer variable **Number**.
- (a) The user enters the number -3 (negative three) when prompted to enter the number of enclosures cleaned.
- (i) What will the impact of this input be on the execution of the program based on this algorithm? (1)
- (ii) Give a brief explanation to substantiate your answer to QUESTION 5.3.2(a) (i). (1)
- (b) The user types in the phrase 'FIVE ENCLOSURES' when prompted to enter the number of enclosures cleaned. What type of error (Syntax/Runtime) will occur? (1)
- 5.3.3 When the program is executed and the user enters a valid number when prompted to enter the number of enclosures cleaned, the payments are calculated and displayed incorrectly. This is the result of TWO logical errors in the given algorithm.
- (a) Explain, in general, what a *logical error* is. (2)
- (b) Explain how to correct EACH of the TWO logical errors in the given algorithm. In your answer make use of the line numbers to refer to the statements in the algorithm. (2 x 2) (4)
- 5.4 A text file called **Clinic.txt** is used to keep track of the animals admitted to the clinic and their caretakers. Below is an example of data in the text file. The number of lines the file contains is unknown.

```
NAME,TYPE,DATE_ADMITTED,CARETAKER
Rose,Impala,2012/04/23,John
Shambo,Lioness,2012/05/01,Koos
Joe,Kudu,2012/05/06,John
Nugget,Impala,2012/05/15,John
Shambo,Lioness,2012/05/27,Thabo
Peach,Impala,2012/06/20,Mary
Suzi,Kudu,2012/06/22,John
```

The following incomplete algorithm was developed as part of the administration system:

Line	
1	Read the first line of text from the file.
2	Initialise a WHILE-loop to read data from the text file.
3	Read one line of text from the text file.
4	Extract the name of the animal, animal type, date of admission and name of the caretaker from the line read from the file.
5	IF ...
6	Display the name of the animal.
7	End loop

5.4.1 Give ONE reason why a WHILE loop is a more suitable control structure than a FOR loop in the given algorithm. (1)

5.4.2 Line 5 of the given algorithm may be completed in a number of different ways.

Analyse the aims and IF statements below and write down ONLY the correct missing logical operator (NOT/AND/OR) to complete each of the IF statements in order to accomplish the set aim in each case.

(a) Aim: List the dates when Shambo, the lioness, was admitted to the clinic.

IF (animal name is Shambo) ... (animal type is a lioness) (1)

(b) Aim: List the dates when all the impala and kudu were admitted to the clinic.

IF (animal type is impala) ... (animal type is kudu) (1)

5.4.3 Write down the name(s) of the animal(s) that will be displayed when line 5 of the given algorithm is replaced with the following IF statement:

IF NOT (admitted during the month of May) OR (animal type is impala) AND (John is the caretaker) ... (3)

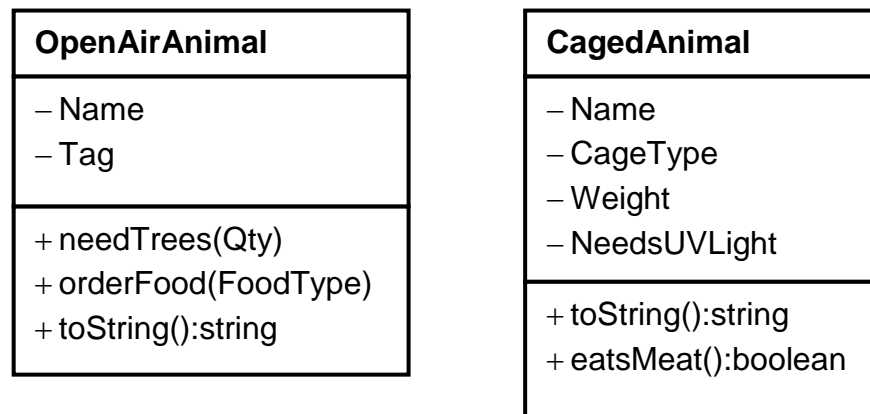
5.5 The software developer who created the administration system for the veterinary clinic made use of object-orientated programming (OOP) principles.

The following class diagram (on the next page) is to be used to store information on a sick animal. Study the class diagram and answer the questions that follow.

ANIMAL	
- ID - Name - Diagnosis - Medication - Hospitalised	Private attributes
+ Constructor() + Constructor(ID) + getAnimalID():string + getAnimalName():string + setDiagnose(Diagnosis) + setMedicationUsed(Medicine) + toString():string	Public methods

- 5.5.1 Two constructor methods are listed in the class diagram above.
- What is the purpose of a constructor in object-orientated programming? (1)
 - How will the program know which constructor to run because both constructors have the same name? (1)
 - What is the term used for having more than one method called by the same name? (1)
- 5.5.2 The software developer needs a method that returns whether the animal is hospitalised or not.
- Classify this method as either a set (mutator) or a get (accessor) method. (1)
 - Write a statement to be added to the given class diagram that will refer to this method. Use a suitable name for the method in your answer. (2)
- 5.5.3 Two new classes to describe specific types of animals housed at the clinic were created, such as the following:
- CagedAnimal** class: animals requiring a closed cage, for example birds and reptiles
 - OpenAirAnimal** class: animals that can be housed in an enclosure in the open air, for example lions and impala

Study the two class diagrams below and answer the questions that follow.



The following algorithm was designed as part of the clinic's administration system:

Line	
1	Define object <u>OpenAir</u> as OpenAirAnimal class type
2	Define object <u>Caged</u> as CagedAnimal class type
3	IF (OpenAirAnimal.Name) = (CagedAnimal.Name)
4	Display message: 'Animals have the same name'
5	IF OpenAir.NeedTrees(toString)
6	Display message: 'Animal needs trees in cage'

- (a) What is the purpose of the toString() method? (1)
- (b) Explain why the IF statement in line 3 is an invalid statement. (2)
- (c) Explain why the IF statement in line 5 is an invalid statement. (2)

TOTAL SECTION D: 48

SECTION E: INTEGRATED SCENARIO**QUESTION 6**

The Red Feather Nature Park makes use of ICTs for marketing purposes.

- 6.1 The Park recently launched its website and still has a lot to learn regarding the design and maintenance of websites.
- 6.1.1 Why is it important for the Park to have a website? (2)
- 6.1.2 Describe THREE requirements of a well-designed website. (3)
- 6.1.3 The administration manager has heard that hackers could gain access and interfere with the Park's website.
- (a) Give ONE possible indication/reason that may make one conclude that a website has been hacked. (1)
- (b) State TWO ways of preventing the hacking of a website. (2)
- (c) Give TWO reasons why hackers usually try to gain access to websites. (2)
- 6.1.4 A website is *search-engine friendly* when it is displayed high up in search results for relevant keywords.
- (a) Why is it important for the Park to have a search-engine-friendly website? (1)
- (b) Google makes use of spider programs in this regard. What is a *spider program*? (2)
- 6.1.5 Navigation is an important part of a website's functionality.
- Name TWO aspects regarding a website's design that will promote good website navigation. (2)
- 6.1.6 The webmaster uses FTP to update the Park's website.
- (a) Define the concept *FTP*. (2)
- (b) Explain the difference between *HTTP* and *FTP*. (2)
- (c) State another way in which the webmaster could update the website. (1)

- 6.1.7 A server is used to host the park's website. The server makes use of RAID technology to protect the website's data.
- (a) RAID Level 0 will not protect the website's data. Explain why this is the case. (2)
 - (b) Motivate why RAID Level 1 will protect the website's data by briefly explaining how RAID Level 1 works. (3)
 - (c) Name TWO operating systems that could be used on the server. (2)
 - (d) Explain how web caching improves the speed of accessing a website. (2)
- 6.2 All the buildings in the Park are connected via an Ethernet network making use of the TCP/IP protocol.
- 6.2.1 Define the term *communication protocol*. (2)
- 6.2.2 Define the term *network topology*. (1)
- 6.2.3 Name the topology that is used in an Ethernet network. (1)
- 6.2.4 The Park is more than five kilometres wide.
- Give TWO reasons why UTP cables are NOT the preferred choice for cabling the network. (2)
- 6.2.5 The botanical garden and the maintenance offices are outer buildings which make use of a VPN to connect to the administration office.
- (a) Briefly explain what a *VPN* is. (2)
 - (b) State ONE disadvantage of using a VPN. (1)
- 6.3 The administration manager has been informed that RSS feeds are a good way to distribute the latest news about the park.
- 6.3.1 Many people prefer RSS feeds instead of visiting a website for news. Give TWO possible reasons why. (2)
- 6.3.2 Name TWO other ways in which newsworthy events at the Park could be distributed online. (2)

- 6.4 The Park sends out a monthly e-mail newsletter to all its subscribers.
- 6.4.1 Describe TWO ways in which the author of the e-mail could prevent the e-mail being flagged as spam by the recipients' e-mail software. (2)
- 6.4.2 An unsubscribe option is included at the bottom of every newsletter. Why is it important that this option is included? (1)
- 6.4.3 The e-mail newsletter has over 5 000 subscribers. A newspaper article has been scanned to be attached and sent out with the e-mail newsletter. The scanner gives the option to create the document in a PDF or TIFF file format.
- (a) Which file format (PDF or TIFF) would you recommend? (1)
- (b) Give ONE reason for your answer. (1)
- TOTAL SECTION E: 47**
GRAND TOTAL: 180



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

NATIONAL SENIOR CERTIFICATE

GRADE 12

INFORMATION TECHNOLOGY P2

NOVEMBER 2012

MEMORANDUM

MARKS: 180

This memorandum consists of 19 pages.

SECTION A: MULTIPLE-CHOICE QUESTIONS**QUESTION 1**

- | | | |
|------|-----|-----|
| 1.1 | C ✓ | (1) |
| 1.2 | B ✓ | (1) |
| 1.3 | B ✓ | (1) |
| 1.4 | B ✓ | (1) |
| 1.5 | C ✓ | (1) |
| 1.6 | C ✓ | (1) |
| 1.7 | A ✓ | (1) |
| 1.8 | D ✓ | (1) |
| 1.9 | B ✓ | (1) |
| 1.10 | D ✓ | (1) |

TOTAL SECTION A: 10

SECTION B: HARDWARE AND SOFTWARE**QUESTION 2**

- 2.1 2.1.1 (a) *Refresh rate* is the number of times/speed of how often✓ the screen must be refreshed or redrawn✓ per second

NOTE: Do not accept the word rate/rating (2)

- (b) *Any THREE valid reasons* ✓✓✓
- Uses less electricity/ LCD more environmentally friendly
 - Doesn't flicker/Less eye strain/Lower refresh rate
 - Uses less space on desktop
 - Lightweight screen
 - LCD has better screen resolution/better image quality
 - LCD is newer technology than CRT

NOTE:
Do not accept cheaper/faster (3)

- 2.1.2 (a) (i) Metadata is data✓ about data✓

Also accept:

Metadata is additional/detailed attributes/properties about data/file

NOTE: Do not accept any example (2)

- (ii) *Any ONE correct example of metadata* ✓

- Model of camera
- Author (photographers name)
- Title/key words/comments/subject/category
- Last saved by/Date last saved/Last printed
- Revision number/Application name/Company
- Date created

Accept any alternative correct example

NOTE: Do not accept basic file properties e.g. size, filename, etc. (1)

- (b) *Any ONE method* ✓

- Reducing the size of the read/write heads/density of tracks
- Change the process of storing data e.g. using vertical/perpendicular recording
- Using file compression

- 2.1.3 (a) *Any ONE fact* ✓
- It must be of the same size (number of pins)/Must fit into the slots on the motherboard
 - The speed of the RAM must be compatible with the motherboard/FSB/System bus
- NOTE:**
Also accept any mention of motherboard compatibility/checking/comparing the specifications of the RAM (1)
- (b) (i) *Any ONE of the following descriptions about virtual memory*
The operating system use secondary storage ✓ (hard drive) space as memory ✓ (temporary or simulated or additional RAM).
- OR**
- Space on the hard disk that can be used to act as memory. (2)
- (ii) *Any ONE fact why virtual memory isn't the solution* ✓
- Virtual memory is slow/using hard disk space as memory slows down the performance
 - Thrashing may occur
 - To be effective the need for a large amount of secondary storage (hard disk space) arises (1)
- (c) • RAM and cache are different ✓ types of memory (DRAM and SRAM).
• Cache memory is found in the CPU, RAM on the motherboard. ✓
• Upgrading RAM means replacing DIMMs while upgrading cache means replacing the CPU ✓ (3)
- 2.1.4 (a) Universal Serial Bus ✓ (1)
- (b) *Accept any ONE of the following:* ✓
- Firewire (Windows)
 - Thunderbolt (Mac)
 - HDMI (1)
- (c) *Any ONE of the following driver descriptions*
Program ✓ that tells an operating system how to communicate with a specific device. ✓
- OR**
- Software designed to control a piece of hardware. (2)

- (d) *Standard driver. Any ONE correct fact ✓*
- is already part of the operating system
 - no need to load extra drivers
- (1)

- (e) *Any ONE device that connects to USB port that can make use of standard/generic driver ✓*
- Examples of devices:
- Mouse.
 - Keyboard.
 - External storage device (e.g. hard drive/flashdisk).
 - Printer/scanner/camera
- (1)

2.2 2.2.1 A *computer virus* is a software program ✓ that is intended to damage ✓ files on a computer system without the user's knowledge or permission.

OR

Potentially damaging software that affects a computer negatively. (2)

- 2.2.2 *Any TWO of ✓✓*
- Always scan a memory stick (or any other device) for viruses before it is used.
 - Install a firewall/antivirus.
 - Do not boot your computer from a device such as a memory stick or external hard drive of unknown origin.
 - Have an up-to-date antivirus program/software installed on your computer/Update your antivirus program regularly
 - Disable the "autorun" function.
 - Opening any unknown/suspicious e-mail attachments
- Or any other correct answer.* (2)

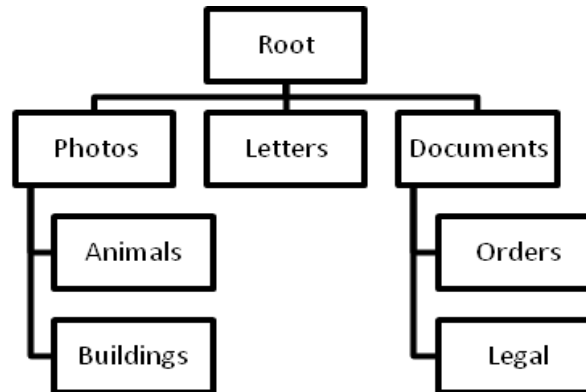
- 2.2.3 *Any TWO of the following in any order: ✓✓*
- tracks (2)
 - sectors
 - clusters

2.2.4 *Accept any correct labelled visual diagram of a file structure showing three levels of file management. ✓✓*

NOTE:

Any structure without labels – ONE mark

EXAMPLE:



(2)

2.3 2.3.1 *RFID: Radio Frequency Identification ✓* (1)

2.3.2 Each lion can have a tag (transponder) ✓ and a receiver can be on each of the feeding stations that need to be monitored ✓. (2)
Also accept any correct alternative explanation

2.4 *Accept any TWO ways a server is better suited to ensure higher performance than a PC ✓✓*

Examples:

- A higher performance CPU.
- More RAM.
- Higher performance hard drives/SCSI/higher RPM drives.
- Server software/more robust operating systems

NOTE:

A larger hard drive/more storage space not acceptable.

Giving an example is acceptable for one mark.

(2)

2.5 2.5.1 *Accept any ONE correct function of the chipset ✓*
 • Control data transfer/data flow between components on the motherboard.
 • Determine the maximum capability of the motherboard. (1)

2.5.2 (a) South Bridge ✓ (1)

(b) North Bridge ✓ (1)

(c) South Bridge ✓ (1)

2.6 2.6.1 *Pipelining is a method of processing where the processor is able to read new instructions ✓ from memory before the instruction that is busy being processed is completed. ✓* (2)

- 2.6.2 *Any TWO correct facts about cache ✓✓*
TWO MARKS FOR:
- Instructions/data probably needed next by the CPU are placed in cache memory which reduced time lost while waiting for instructions/data from RAM as a result it improves the performance of the CPU.
- ONE MARK EACH FOR:*
- Cache memory is situated inside the CPU or close by.
 - Cache faster type of memory – can feed the CPU with instructions at a faster rate.
 - Instructions/data probably needed next by the CPU are placed in cache memory. (2)
- 2.6.3 (a) *Over-clocking* is when the settings of the CPU are changed in order for it to run at higher rates than it was designed for. ✓
Accept any similar explanation
- NOTE:** *Also accept the following (due to translation error)*
Clock multiplication:
- is the increasing of the CPU's speed in relation to the motherboards speed
 - allows the running of the CPU at a faster rate than the motherboard (1)
- (b) *Any TWO correct facts about preventing overheating ✓✓*
- Large heat sink on CPU/North Bridge chipset.
 - Using thermal paste
 - Heat conductive plate over system bus.
 - Additional cooling fans in case.
 - Alternative cooling systems – any sensible example. (2)
- 2.7 2.7.1 Satellite/Microwaves✓ communication (1)
- 2.7.2 *Any TWO valid examples from scenario (zoo/reptile park/aquarium) ✓✓*
- Guiding someone to a specific animal enclosure.
 - Give you information about the layout of the park.
 - Together with audio feed it can give you information about animals at a specific location.
 - Using satellite technology (GPS), audio and/or multimedia content is triggered based on a user's location in the park. (2)
- 2.7.3 MP3/wav ✓
NOTE: Do not accept avi/MP4 or any other video format (1)

TOTAL SECTION B: 50

SECTION C: APPLICATIONS AND IMPLICATIONS**QUESTION 3: e-COMMUNICATION**

- 3.1 3.1.1 *Accept any TWO advantages from the parks perspective ✓✓*
- The park can use e-communication to promote their services and products.
 - The park's services get promoted world wide/potential visitors to the park may view the parks' services on the Internet before visiting the park.
 - Visitors may be informed of disruptions in services using an e-mail/sms
 - Cost effectiveness with explanation/example (2)
- 3.1.2 *Accept any ONE disadvantage from a visitors viewpoint ✓*
- People get distracted by phone calls/sms.
 - Do not get away from work related issues.
 - Chatting online can disrupt the outing.
 - Public places have insecure Internet connections (1)
- 3.2 3.2.1 The http changes to **https**✓://www.redpark.co.za (1)
- 3.2.2 *Encryption* is a method for changing data✓ into an unreadable format✓
- ONE MARK FOR :*
A correct example explaining encryption
- TWO MARKS FOR:*
Scrambling data (2)
- 3.2.3 The message must be decrypted using the private✓ key✓. (2)
- 3.2.4 A ✓ (1)
- 3.3 *Accept any TWO sensible practical examples of using podcasting. ✓✓*
- Daily commentary about the health of an endangered animal can be recorded.
 - Keep the visitors updated on activities taking place at the park.
 - Recordings of the actual birth process can be made and released via podcasting.
 - Press releases regarding the arrival of new animals/babies can be released via podcasting. (2)
- 3.4 Trojan is a program with malicious intent✓, presenting itself as some useful/harmless program✓. (2)

[13]

QUESTION 4: SOCIAL AND ETHICAL ISSUES

- 4.1 *Any TWO correct ways to reduce electricity consumption – have to be IT related ✓✓*
- Use "greener" hardware which uses less power.
 - Make use of laptops instead of desktop computers.
 - Turn off computers when not used.
 - Share resources such as printer, scanners and other peripherals.
 - Computer hardware may make use of sleep mode. (2)
- 4.2 4.2.1 *Accept any TWO valid responsible recommendations to dispose of old computers ✓✓*
- Request special e-waste centres to collect the old computers/Separate the metal and plastic parts and send to appropriate recycle centres.
 - Donate the computers to a school or charity.
 - Make old machines available to a second hand dealership. (2)
- 4.2.2 YES or NO
NOTE: Must be supported by a valid reason to get the marks
Accept any valid reason for replacing old computers ✓✓
If answered YES:
- This is a cost effective approach.
 - Computers should last five years but will then be old enough to warrant an upgrade.
 - New technology advances in computing make it ideal to upgrade every five years.
- If answered NO:**
- Too expensive
 - The period of 5 years may be viewed as too long and a shorter term of 3 years may be advocated. (2)
- 4.3 *Any TWO viable recommendations ✓✓*
- Software is available that will perform a low level format on the hard drives to ensure that data cannot be retrieved.
 - If the hard drive is no longer required it can also be destroyed.
 - Lock the old hard drives in the vault of the nature park for archiving purposes.
 - Make use of file encryption software to encrypt the data and files.
- NOTE:**
Do not accept format/quick format of hard drive.
Do not accept making backups (2)

4.4 YES or NO

NOTE: Must be supported by a valid reason to get the marks

Accept ONE correct explanation ✓✓

If answered YES:

- The company has the right to know what type of person they are employing.
- The employee may be a potential embarrassment to the company due to his statements on Facebook.
- By viewing the profile the company may prevent victimization of other employees.

If answered NO:

- It is intrusive on a person's private life.
- An employee has the right to confidentiality provided they do not do anything that jeopardizes the organization they work for.
- The constitution protects freedom of speech and association. (2)

4.5 Computer crimes are illegal activities✓ performed using a computer✓. (2)

NOTE:

Also accept an example.

Do not accept rephrasing of the question e.g. "crime using a computer"

[12]

TOTAL SECTION C: 25

SECTION D: PROGRAMMING AND SOFTWARE DEVELOPMENT**QUESTION 5: ALGORITHMS AND PLANNING**

- 5.1 5.1.1 (a) Currency ✓
Also accept: Number (1)
- (b) Text ✓
Also accept: Text with maximum 10 characters (1)
- 5.1.2 *Number is suitable* because you can use any combination of two numbers e.g. 0 and 1 or 0 and -1 to indicate true/yes or false/no ✓ (1)
- 5.1.3 (a) Accept any correct explanation of the condition for validation in pseudo or programming code. ✓✓✓
Examples:
• "K" OR "U" OR "M"
• IN("K","U","M") (3)
NOTE:
Accept double/single/no quotes
Do not accept rephrasing of the given question
- (b) *Accept any appropriate error message* ✓
Examples:
• Make only use of the letters K/M/U.
• Only the letters K, M or U are allowed (1)
- 5.2 5.2.1 (a) The table doesn't provide for more than two different types of trees✓✓ to be bought per order.
Accept any similar explanation (2)
- (b) With the field OrderDate as primary key, only one record with a certain date is allowed in the table. It means you are only allowed one order for a certain date ✓✓
Accept any similar explanation (2)
- 5.2.2 (a) *Combined primary key* consists of two or more fields✓ combined together as primary key✓, in order to uniquely identify each record
NOTE:
Do not accept two primary keys per table (2)

- (b) *Accept any TWO valid reasons ✓✓*
- To uniquely identify each record
 - So that one order may have more than two types of trees.
- **If TreeID is not part of primary key (KEY = OrderNo):**
 - You can only buy one tree per order.
 - A new order has to be placed every time you buy more than one type of tree
 - **If OrderNo is not part of primary key (KEY = TreeID):**
 - You can order a certain type of tree only once
 - If you want to buy the same type of tree you have to assign a new TreeID for that tree. (2)
- (c) *Partial dependency – each field in the table must be fully dependant on the whole single primary key/part of the combined primary key ✓ (1)*
- 5.2.3 (a) One-to-Many ✓
(*note the order of the 1 and many*)
Also accept:
- 1 to ∞ / 1 \rightarrow ∞
 - 1 to M / 1 \rightarrow M (1)
- (b) Any ONE of ✓
- OrderNo
 - TreeID (1)
- (c) YES ✓
Accept ONE correct explanation/reason ✓
- The field TotalAmount can be a calculated field
 - The total for the order can be calculated using a function/formula in a query.
 - The field can be calculated using an aggregate function such as SUM
- NOTE:**
- If only answered YES – no mark allocated (2)
- 5.3 5.3.1 Algorithm is a set of steps ✓ to solve a problem. ✓
Also accept any other valid/similar explanation (2)

- 5.3.2 (a) (i) **Impact:** ✓ (*accept ONE correct impact*)
- Nothing/The execution of the program won't be affected/The program will produce negative output.
 - No run-time error will occur.
 - The flow of the program will continue and the user will be asked to enter the next day's number of enclosures cleaned. (1)
- (ii) **Explanation:** ✓ (*accept a valid explanation for the impact mentioned*)
- No data validation is done on the input value and therefore the program will accept the value typed in.
 - The input data (-3) is an acceptable integer number but not valid/correct given the scenario. Being negative number it will be deducted from the total number of enclosures cleaned.
 - A logical error will occur with the displayed value being incorrect - the total number of enclosures cleaned will be less than the actual number – the cleaning staff will be paid less than required (1)
- (b) Runtime ✓ (1)
- 5.3.3 (a) Logical error is when the output received/displayed ✓ is incorrect or not as expected ✓ because of an error based on logic. (2)
Also accept any other valid/acceptable explanation
- (b) *Accept ONE correction for each logical error*
Correction for error Line 4: ✓
Move line 4 to any position above the starting point of the loop ✓.
- OR**
- Move line 4 between lines 1/2 or 2/3.
- Correction for error Line 7:** ✓
Change line 7 to: $\text{Pay} \leftarrow \text{Pay} + (\text{Number} * 100)$ ✓
- OR**
- Change line 7 to: $\text{Pay} \leftarrow \text{Total} * 100$
Move line 7 to outside the loop, between lines 8 and 9. (2 x 2) (4)
- 5.4 5.4.1 *Accept any ONE correct reason for using a WHILE in stead of FOR-loop* ✓
- The size of the text file is unknown
 - For-loops are used when the number of repetitions is known (1)
- 5.4.2 (a) AND ✓ (1)
- (b) OR ✓ (1)

5.4.3 *All names ✓✓✓*

Rose
Nugget
Peach
Suzi

NOTE: *Subtract one mark for each error. (maximum three errors)* (3)

- 5.5 5.5.1 (a) *Accept ONE correct purpose of the constructor method ✓*
- It creates an instance of/initialises the object
 - It reserves/creates memory space for the object in RAM (1)
- (b) The compiler will look at the arguments (parameters) and find the constructor with matching number/type of parameter(s). ✓ (1)
- (c) Overloading ✓ (1)
- 5.5.2 (a) Accessor ✓ (get) (1)
- (b) + `getHospitalisation()` ✓ : boolean ✓
Accept any alternative name for the method
Accept any appropriate return data type e.g. string/integer (2)
- 5.5.3 (a) *Accept ONE aim of toString() method ✓*
- Give a standard output/description of the attributes of an object
 - Allows the output of the values of the attributes of an object
 - To combine all attributes of the class in a single string (1)
- (b) *Accept ONE correct reason ✓✓*
- The use of the class name as a reference to an object in the program is not allowed
 - The use of private attributes is not possible in another unit/class
- NOTE:**
Do not accept any programming syntax (2)
- (c) *Accept ONE correct reason ✓✓*
- The toString method's return type is string; whereas the parameter's data type is integer
 - The method NeedTrees is a mutator method (setting a value) – cannot be part of another statement
 - The IF statement requires a logical expression to evaluate. (2)

TOTAL SECTION D: 48

SECTION E**QUESTION 6: INTEGRATED SCENARIO**

- 6.1 6.1.1 *Accept any TWO correct reasons for having a website ✓✓*
- Many people use the Internet to search for tourist destinations or nature parks specifically.
 - The website may be used in the marketing the park.
 - The website can be an education tool in the parks social responsibility to the community.
 - The website allows international contact.
 - Online bookings (2)
- 6.1.2 *Accept any THREE recommendations/aspects of a well designed website ✓✓✓*
- The website should have an attractive design with enough space between elements to make it easy to read.
 - There should be appropriate use of graphics and colours.
 - A standard template should be used across all the pages so that the user can easily understand the website.
 - Content should be well written and up to date.
 - The website should be easy to navigate.
 - User friendly (3)
- 6.1.3 (a) *Accept any ONE of the following: ✓*
- The website will not load.
 - A webpage has been defaced such as an image changed.
 - The site has been deleted or replaced by a different web page.
 - Antivirus software detects a virus on the website when it is browsed.
- Accept any correct alternative answer (1)*
- (b) *Accept any TWO ways of preventing hacking ✓✓*
- The server should have all software updates installed.
 - The website should be well programmed so that there is not a way that a hacker can gain entry through a particular webpage.
 - A firewall should be used to protect the website.
 - An administrator should monitor traffic to the website to catch any suspicious activity.
 - Do not use the default password for the underlying database. (2)

- (c) *Accept any TWO reasons why hackers need access* ✓✓
- For financial gain.
 - To prove that they can to gain status amongst their peers.
 - To point out security flaws in the website to the administrator.
 - A political issue that the hacker has with the organisation behind the website.
 - For entertainment
- Accept any correct alternative answer* (2)
- 6.1.4 (a) *Accept any ONE reason for having search engine friendly website* ✓
- So that it receives the most visits possible to the website.
 - People will most likely open the websites listed at the top of the result set.
 - The name of the park gets promoted and more people are aware of the park.
- Accept any correct alternative explanation/reason* (1)
- (b) *Accept any ONE explanation* ✓ (indexing) ✓ (database)
TWO MARKS FOR:
- Spiders are programs that index websites in a local database.
 - Spiders are programs that traverse all linked pages where possible on the internet and store the content in a database. When a user searches, the search engine searches its own database which is fast.
- ONE MARK FOR:
- A program that makes a lot of information accessible by linking it to the search engine/site.
 - Spiders are programs that index websites
- (2)
- 6.1.5 *Accept any TWO aspects about good navigation* ✓✓
- It should be clearly visible on every page.
 - It should perform in a consistent and uniform manner.
 - It should function on any browser.
 - It should show the layout of the website.
 - It should have a Home page link.
 - Displays different categories on different pages
- (2)
- 6.1.6 (a) *FTP: A standard* ✓ *that permits file uploading and downloading* ✓ *between different computers.*
- ONE MARK FOR:
File Transfer Protocol. (2)

- (b) **HTTP** is used to serve up web pages that can be interpreted by a browser and displayed in a graphical way✓

FTP can only transfer complete files✓

NOTE:

If FTP is explained in Question 6.1.6 (a) and only HTTP answered in (b) two marks. (2)

- (c) *Accept any ONE other way of updating website ✓*
- He could make use of a content management system
 - He could have a web-based file management system to update files
 - He could update files directly on the server (1)

- 6.1.7 (a) RAID 0 creates one logical drive and does not duplicate✓, mirror or backup data in any way✓.
Accept any correct alternative explanation. (2)

- (b) RAID 1 operates by duplicating/mirroring✓ the data✓ across two hard disks ✓ having high fault tolerance. (3)

- (c) *Any TWO server operating systems ✓✓*
- Ubuntu Server
 - Windows Server
 - FreeBSD
 - Novell
 - Unix
 - Linux (2)

- (d) Web caching saves files of recently accessed websites on the hard disk✓. When the visited site is revisited the web browser does not have to re-download the website from the Internet again.✓ (2)

- 6.2 6.2.1 *Communication protocol* is a set of rules/procedures✓ that is used to determine how data is sent between two devices✓ in a network. (2)

Also accept any TWO of the following facts about protocol:

The protocol determines:

- if compression of data is required
- the fault tolerance/control to determine if the data sent was received correctly
- the method that the sending device determines when transmission is terminated
- the method that the receiving device indicates that the complete message was received

- 6.2.2 *Network topology* is the possible layout or physical connection of the devices in a network. ✓ (1)
- 6.2.3 Star/Bus ✓ (1)
- 6.2.4 *Accept any TWO correct reasons for not using UTP* ✓✓ (2)
- UTP is highly susceptible to signal loss/attenuation/cable length is limited to ± 100m
 - Using UTP cable for 5 km would require signal boosters every 100m – extra cost implication
 - UTP is susceptible to EMI
 - UTP is susceptible to eavesdropping
- 6.2.5 (a) VPN is a private network ✓ which uses the Internet to connect computers as if they are connected via a LAN ✓. (2)
- Also accept any ONE of the following explanations:*
- This allows a client to join a private or secure network using a non-secure connection.
 - VPN uses encryption to protect access to the network or the data on the network.
- Also accept Virtual Private Network for one mark.
- (b) *Accept ONE disadvantage of using a VPN* ✓ (1)
- Can be slow.
 - Not available for all clients if internet connection not available/can be "off line" due to cable theft of Telkom lines.
 - If user forget password – cannot connect to VPN
 - Specialised security measures needed/security can be easily compromised
- 6.3 6.3.1 *Accept any TWO correct reasons for preferring RSS* ✓✓ (2)
- An RSS feed reader updates itself automatically/only supply the content that changed.
 - User receives instant updates.
 - Safer than browsing the website directly.
- 6.3.2 *Accept any TWO correct online alternatives* ✓✓ (2)
- Facebook
 - Twitter
 - An e-mail newsletter/ mailing lists.
 - Podcasting
 - Forums
 - Blog
 - UTube
- NOTE: Do not accept Wiki, IM, IRC, RSS Feed**

- 6.4 6.4.1 *Accept any TWO correct ways for e-mails not to be flagged as spam ✓✓*
- The e-mail should not contain too many hyperlinks
 - The e-mail should be sent from an appropriate address
 - There should not be full word capital letters in the subject or body
 - The e-mail may not contain suspect words/phrases/attachments
 - The e-mail needs an appropriate subject line
 - Too many recipients (2)
- 6.4.2 *Accept any ONE correct fact for having a unsubscribe option✓*
- A user should always have the option to unsubscribe
 - If no unsubscribe option, the email is classed as spam and can be reported. (1)
- 6.4.3 (a) PDF document ✓ (1)
- (b) *Accept any ONE correct reason ✓*
- PDF is a compressed universal format that can contain text and images.
 - PDF documents can be password protected to prevent editing of file
 - TIFF is uncompressed and is not as widely supported. (1)
- TOTAL SECTION E: 47**
GRAND TOTAL: 180