

**NATIONAL
SENIOR CERTIFICATE**

GRADE 12

SEPTEMBER 2015

INFORMATION TECHNOLOGY P1

MARKS: 150

TIME: 3 hours



This question paper consists of 12 pages.

INSTRUCTIONS AND INFORMATION

1. This is a three-hour examination. Because of the nature of this examination, it is important to note that you will not be permitted to leave the examination room before the end of the examination session.
2. You require the files listed below in order to answer the questions. They are either on a CD issued to you, or the invigilator/educator will tell you where to find them on the hard drive of the workstation you are using or in a network folder.

Question1	Question2	Question3
Question1_u.pas Question1_p.dpr Question1_u.dfm	Question2_u.pas Question2_p.dpr Question2_u.dfm clsTravel_u.pas Destinations.txt	Question3_u.dfm Question3_u.pas Question3_p.dpr

3. If a CD containing the above files was issued to you, write your name and surname on the label of the CD.
4. Save your work at regular intervals as a precaution against power failures.
5. Rename the *P1 Data folder* as your *Name and Surname*.
6. Type in your name and surname as a comment in the first line of each program.
7. Read ALL the questions carefully. Do only what is required.
8. At the end of this examination session you will be required to hand in the CD with all the files with the work you have done or you must make sure that ALL the files with your work have been saved on the network as explained to you by the invigilator/educator.
9. Ensure that ALL files can be read.
10. During the examination you may use the HELP functions of the software. You may NOT make use of any other resource material.

SCENARIO:

Organising a holiday can be very time consuming. Booking a hotel and tours are the tasks that are carried out by a travel agency for a customer's convenience. You have been asked to help out in completing various programs made for an agency.

QUESTION 1: DELPHI PROGRAMMING

You have been provided with the following incomplete Delphi program that will be used by a manager at a hotel.

Open **Question1_p.dpr** and complete the program:

1.1 Button [Question 1.1]

Enter the name and surname of the client in the text boxes provided. Select the room that will be booked.

Create and display a line of text as output that indicates the name and surname of the client, as well as the selected room.

Example of output:

(4)

1.2 Button [Question 1.2]

Enter the number of days that the client wants to stay in a single room.

- The first 5 days are charged at the normal rate of R250 per night.
- The additional days will be charged at 96% of the normal charge (4% discount).
- Display the amount to be paid formatted as currency.

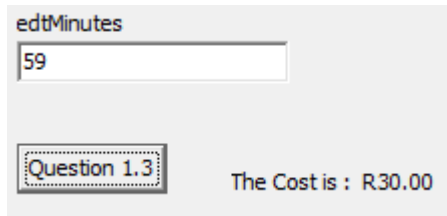
Example of output:

(8)

1.3 Button [Question 1.3]

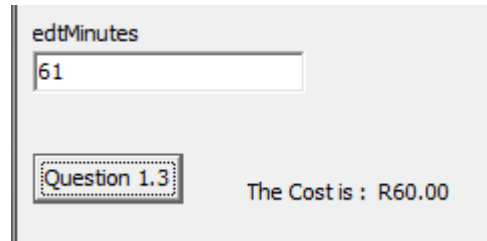
Enter the number of minutes that a client has used on the internet service at the hotel. Display the total cost to be paid (formatted as currency), charged at R30 per hour. For any part of an hour used, the client is charged a full hour's rate.

Example of output:



edtMinutes
59

Question 1.3 The Cost is : R30.00



edtMinutes
61

Question 1.3 The Cost is : R60.00

(6)

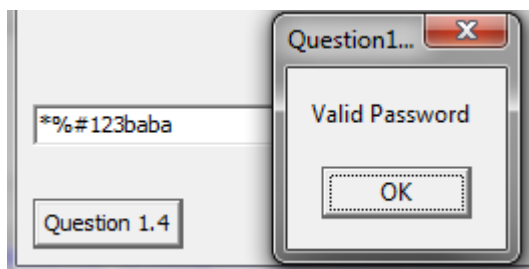
1.4 Button [Question 1.4]

A guest must provide a password when using the hotel's computer system. The password must have:

- 10 characters
- At least three characters from this list (*,?,#,%)
- At least three numbers

When the button is clicked the program must check the validity of the password entered.

Example of output:

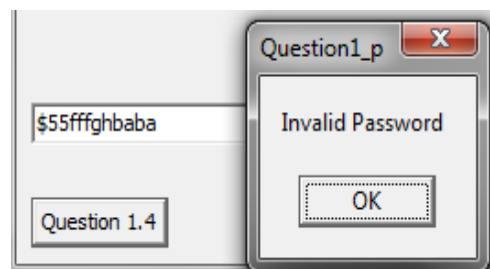


*%#123baba

Question 1.4

Valid Password

OK



\$55ffffhbabab

Question 1.4

Invalid Password

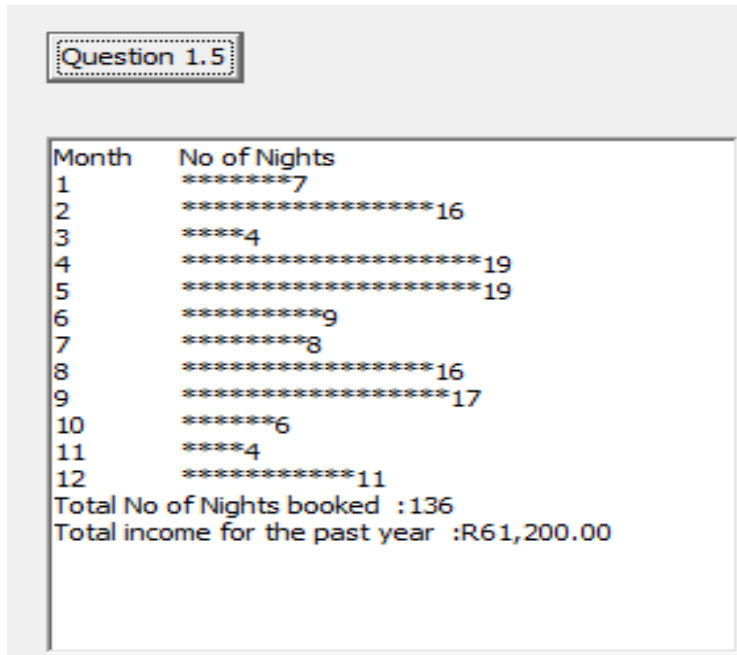
OK

(15)

1.5 Button [Question 1.5]

The number of nights that a double room has been booked is randomly generated for each month, for the past 12 months. The maximum number of nights that the room could be booked for, is 20 nights per month.

A graph is generated in the output area showing the randomly generated nights it has been booked. The total nights booked for the year as well as the total income generated by the double room is also displayed.



Use the following algorithm:

Clear the output area

Format the output area to display the information in columns

Write headings

Initialise all variables

Start outer loop bound by the number of months

Randomly generate the first number

Initialise the string variable

Start inner loop bound by the randomly generated variable

Generate stars and create a new string

Close inner Loop

Display month number, stars as well as how many stars

Calculate Total Nights booked for the year

Close outer loop

Display Total Nights

Display Total Cost if Double Room's rate is R450 per night

Take note that output will be different as random numbers are generated.

Example of output: (17)

- Enter your name and surname as a comment line in the first line of the file named Question1_u.pas.
- Save the program.
- Make a printout of the code of the Question1_u.pas file. [50]

QUESTION 2: OBJECT-ORIENTED PROGRAMMING

You are assisting a travel agency with their quotation system for reservations made at various destinations. They specialise in local tourist attractions.

2.1 Open the class unit, **clsTravel_u.pas**, and complete the code.

2.1.1 Declare the following private fields with appropriate data types. (3)

Name of Attribute	Description
fbooking	booking number of reservation
fdest	destination to be visited
fdate	date when the destination will be visited
fpeople	number of people
fdiscount	amount (in Rands) of discount they qualify for
ftotal	total amount (in Rands) for the booking

2.1.2 Write a **constructor** which will receive the booking number, destination, date, number of people and total amount as parameters. These parameters must be used to initialise the attributes of the class. All attributes must be initialised. (3)

2.1.3 Write a method, **CalcDiscount**, which will calculate the amount of discount a client qualifies for.

Discount is calculated as follows:

- If the reservation was made for any time in February, May, August or November, they will receive 5% discount.
 - If the number of people in the group is 15 or more then they will qualify for 2,5% discount, regardless of the time of the year (any month).
- (12)

2.1.4 Write a method, **CalcTotal**, which will update the total amount by subtracting the discount. (2)

2.1.5 Write a **toString** method to construct a string which contains the information in the format as shown below.

Booking Number: <booking number>
Destination: <destination>
Date: <date>
Number of People: <number of people>
Discount: <discount amount formatted to currency>
Total: <total amount formatted to currency>

(5)

2.1.6 Write a method which will return the destination. (2)

- 2.2 The text file, **Destinations.txt**, contains the data needed to compile a report on the bookings made by clients.

The layout of the text file is as follows:

```
<booking nr>;<destination>;<date of booking>;<number of people>;<total  
amount quoted>
```

Example of the first few lines of the text file:

```
AF256;Augrabies Falls;08/11/2015;6;600  
BRC54;Blyde River Canyon;13/12/2015;7;878.50  
KNP10;Kruger National Park;05/12/2015;14;15409.80  
HI839;Hluhluwe-Imfolozi Park;13/11/2015;3;2697  
D1028;Drakensberg;12/11/2015;6;600  
KT888;Kgalagadi Transfrontier Park;17/11/2015;3;4020
```

Complete the code in the main unit (**Question2_u.pas**).

- 2.2.1 The user will select a destination from the combobox.

Test whether the text file exists. If it does not exist, display a suitable message and exit.

Do the following if the text file exists:

- Make use of a conditional loop to extract the information from the text file.
- Use the data from the line of text to instantiate a new object.
- Calculate the amount of discount that the client qualifies for. Call the appropriate method.
- The new total, after discount has been deducted, must be calculated.
- Display all the booking information using the **tostring** method. If 'ALL' was selected in the combobox, then all the destinations must be displayed.

(26)

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

[53]

QUESTION 3: PROBLEM SOLVING

The agency needs an application for changing the flight date and generating a boarding pass for its clients.

NOTE: Read the following sections carefully before attempting to answer this question:

- GUI AND DATA SUPPLIED
- INSTRUCTIONS
- PROBLEM REQUIREMENTS
- MARK ALLOCATION

GUI AND DATA SUPPLIED

The GUI that is supplied contains all the components.

Data for clients who have booked for the flights has been provided in the 2-dimensional array declared as **ar2flights**. The array contains information for 26 booked clients.

An example of the first five records:

Name	Date	From	To
John	09/09/2015	JHB	CPT
Sarah	12/11/2015	ELS	JHB
Pete	10/01/2015	JHB	CPT
Lorna	19/11/2015	DBN	JHB
Maxwell	12/11/2015	ELS	DBN

Column 1 – Name of the client
 Column 2 – Date of the flight
 Column 3 – Departure City
 Column 4 – Destination City

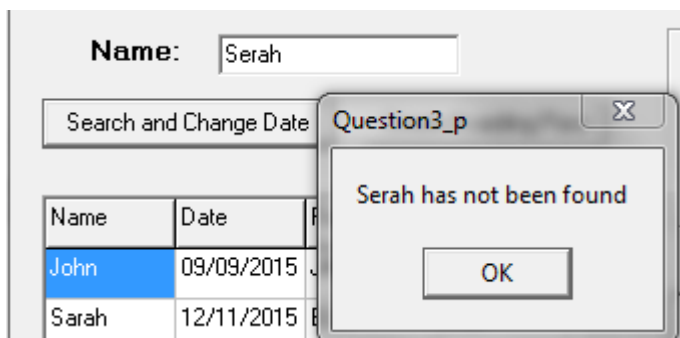
INSTRUCTIONS:

- Open the incomplete project file **Question3_p.dpr** in the **Question3** folder.
- Insert your name and surname as a comment in the first line of the unit file **Question3_u.pas**.
- Write code to answer the question according to the requirements stated in the section that follows.

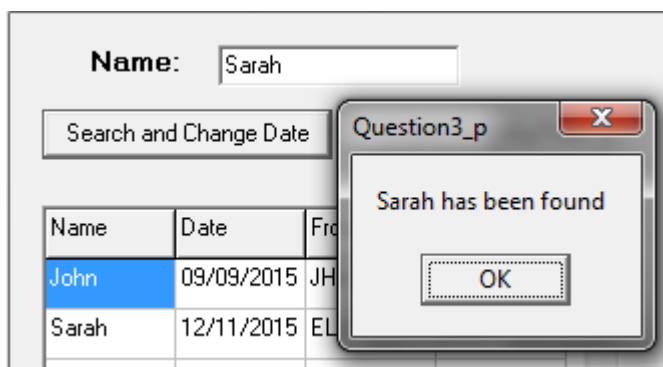
PROGRAM REQUIREMENTS:

- Display the information of the 2d array in the string grid as the program loads.
- Search the array to check whether the name that has been entered in the text box is in the array. If the name is in the 2d array then ask the user to enter the new departure date. Change the date in the array to the new date, display the modified array.
- Display suitable messages as shown in the examples.

Example of output if name is not found:



Example of output if name is found:



The screenshot shows a flight booking application interface. At the top, there is a 'Name:' label followed by a text input field containing 'Sarah'. Below this is a 'Search and Change Date' button. A table lists passengers: John (09/09/2015, JH), Sarah (12/11/2015, EL), and Pete (10/01/2015, JH). A modal dialog titled 'Enter Date' is open, showing a date input field with '15/11/2015' and 'OK' and 'Cancel' buttons. The background is slightly dimmed.

The screenshot shows the same flight booking application interface as above. The 'Enter Date' dialog is now replaced by a smaller modal dialog titled 'Question3_p' with the message 'Date successfully changed' and an 'OK' button. The table now shows Sarah's date as 15/11/2015.

On this particular plane, the layout looks as follows regarding Business Class (row 1–6) and Economy Class (row 7–30):

1	A	C	
2	A	C	
3	A	C	
4	A	C	
5	A	C	
6	A	C	
7	A	B	C
8	A	B	C
..	A	B	C
30	A	B	C

AISLE

D	F	
D	F	
D	F	
D	F	
D	F	
D	F	
D	F	
D	E	F
D	E	F
D	E	F
D	E	F

Randomly generate the seat number based on the type of ticket that was booked (Business Class or Economy Class).

Display the final details in the output area and save it to a file so that the client can print the information. Use the client's name as the file name.

Take note that output will be different as random numbers are generated.

Examples of output:

Sarah

Change Date Create Boarding Pass

Date	From	To
15/09/2015	JHB	CPT
15/11/2015	ELS	JHB
15/01/2015	JHB	CPT
15/11/2015	DBN	JHB
15/11/2015	ELS	DBN

Class

☒ Business Class

☐ Economy Class

Name : Sarah
Date : 15/11/2015
From : ELS
To : JHB
Seat : 1A
****File Sarah.txt successfully saved****

Sarah

Change Date Create Boarding Pass

Date	From	To
15/09/2015	JHB	CPT
15/11/2015	ELS	JHB
15/01/2015	JHB	CPT
15/11/2015	DBN	JHB
15/11/2015	ELS	DBN

Class

☐ Business Class

☒ Economy Class

Name : Sarah
Date : 15/11/2015
From : ELS
To : JHB
Seat : 27B
****File Sarah.txt successfully saved****

MARK ALLOCATION		Maximum marks
Application of good programming techniques (indentation, variable names) and modular design.		5
Displaying contents of the 2-dimensional array with headings		6
Searching the name, changing the date.		16
Generating the seat number.		14
Saving the file and displaying.		6

- Enter your name as a comment in the first line of the program file.
- Save the program.
- A printout of the code may be required.

[47]

TOTAL: 150