CONFIDENTIAL

MALAYSIAN INSTITUTE OF INFORMATION TECHNOLOGY

FINAL EXAMINATION

JANUARY 2016 SEMESTER

SUBJECT CODE	:	ISB42603
SUBJECT TITLE	:	ADVANCED PROGRAMMING
LEVEL	:	BACHELOR
TIME / DURATION	:	9.00 am – 11.30 am
		(2 ½ HOURS)
DATE	:	29 MAY 2016

INSTRUCTIONS TO CANDIDATES

1. Please read CAREFULLY the instructions given in the question paper.
2. This question paper has information printed on both sides of the paper.
3. This question paper consists of two sections; Section A and Section B.
4. Answer ALL questions in Section A. For Section B, answer TWO (2) questions only.
5. Please write your answers on the answer booklet provided.

THERE ARE 7 PAGES OF QUESTIONS, EXCLUDING THIS PAGE.

SECTION A (Total: 40 marks)

INSTRUCTION: Answer ALL questions in this section.
Please use the answer booklet provided.

Question 1

a. Write the code segment to display the message box when a certain button is clicked, as shown in Figure 1. The greeted name must be extracted from the text boxes *txtFirstName* and *txtLastName*.

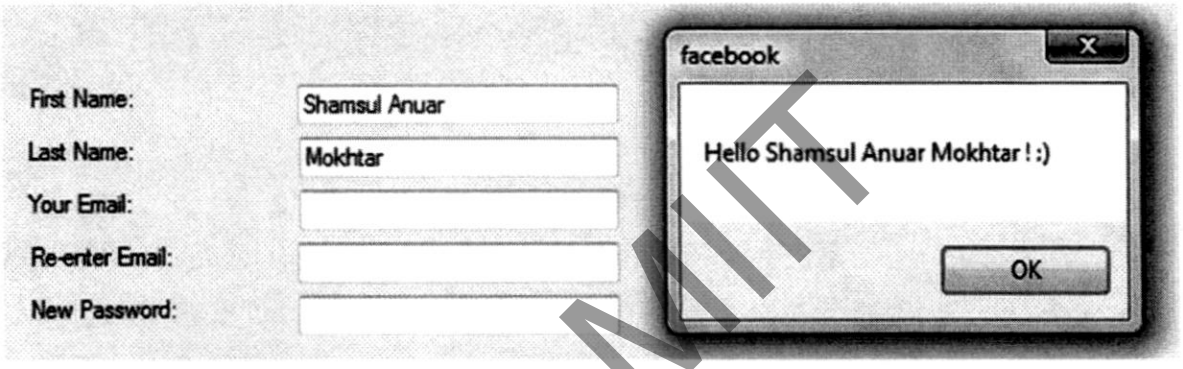


Figure 1

(5 marks)

b. Write the code segment to display the combo box (name: *cmbYear*) as shown in Figure 2. Use appropriate loop statement that displays the days from 2015 to 1957 in descending order.

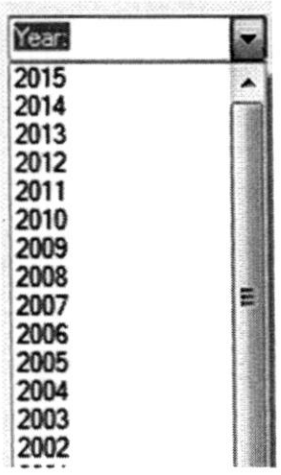


Figure 2

(5 marks)

Question 2

a. Explain concisely the concept of class in C#.

(2 marks)

b. Access modifiers define the accessibility level of any member of a class or assembly. Name any TWO (2) access modifiers in C# and describe the scope of accessibility for each.

(6 marks)

c. A piece of C# code is given as follows:

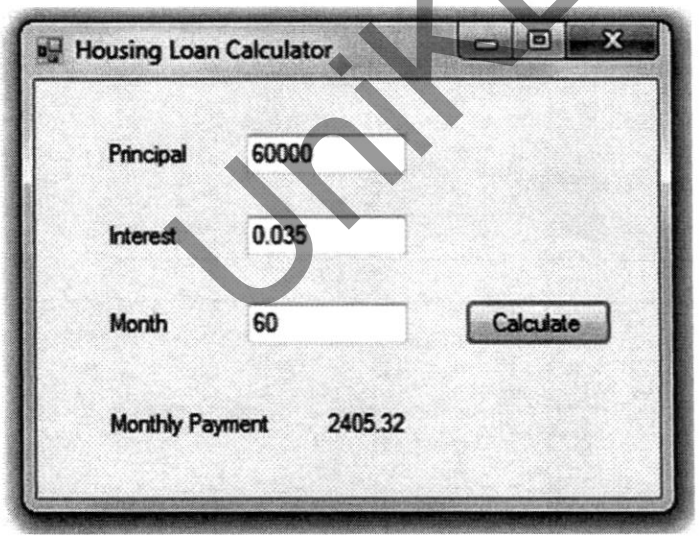
```
Countries malaysia = new Countries();
```

Explain what happens when the code is executed.

(2 marks)

Question 3

The figure below shows a program to calculate and display the monthly payment (M) for a housing loan for any given principal (P), monthly interest (I) and number of month (N). Using suitable named controls and variables, write the code segment for the button_click event to perform the task. The form and the formula for calculating the monthly payment is given in Figure 3. Display the result in two decimal places.



$$M = P \frac{[\,I(1+I)^N\,]}{[\,(1+I)^N - 1\,]}$$

M = Monthly payment

P = Principal
(i.e. loan amount, e.g. RM 60,000)

I = Monthly interest
(e.g. 3.5%, i.e. 0.035)

N = Number of months
(e.g. 5 years, i.e. 60 months)

Figure 3

(10 marks)

Question 4

Write a code segment to calculate the electricity bill. The electricity tariff is calculated based on three blocks of units used.

Block 1:	First 500 units	-	RM 0.28 for every unit
Block 2:	Subsequent 300 units	-	RM 0.35 for every unit
Block 3:	The subsequent after previous blocks	-	RM 0.42 for every unit

It is given that the variables unitTotal, unitBlock1, unitBlock2, unitBlock3 and amountTotal have been declared as type double. The code segment must calculate and assign the values to the unitBlock1, unitBlock2, unitBlock3 and amountTotal variables based on any given unitTotal. You are NOT REQUIRED to write the code to read/assign values from/to control items on the form.

(10 marks)

SECTION B (Total: 60 marks)

INSTRUCTION: Answer TWO questions only in this section.
Please use the answer booklet provided.

Question 5

The program as shown in Figure 4 allows the look up of the correct shipping charge based on a given shipping rate table given in Table 1.

Table 1

Weight/Category	A	B	C	D
1 lb	1.00	1.50	1.65	1.85
3 lb	1.58	2.00	2.40	3.05
5 lb	1.71	2.52	3.10	4.00
10 lb	2.04	3.12	4.00	5.01
> 10 lb	2.52	3.75	5.10	7.25

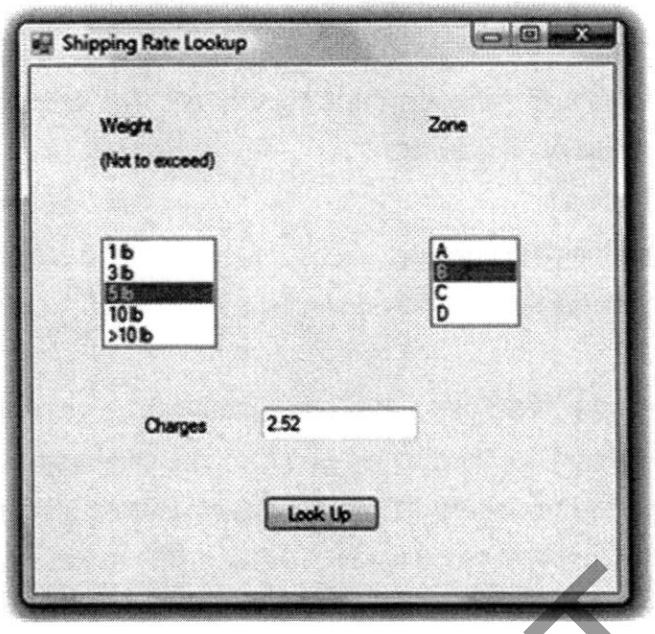


Figure 4

a. Write the code to declare and initialise 2-dimensional array to store rate table values.

(6 marks)

b. Write the code to populate the both listboxes with the necessary items. Use suitable names for the listbox controls.

(9 marks)

c. Write the code segment for the Look Up button click event. Display a suitable messagebox dialog if the user forgot to select any of the listboxes. Use suitable name controls on the form.

(15 marks)

Question 6

The program shown in Figure 5 accepts input in the form of coursework and final exam marks. The program uses a class that calculates the overall mark, grade and grade point. The overall mark is calculated based on coursework and final exam. The grade and grade point are determined based on the overall mark.

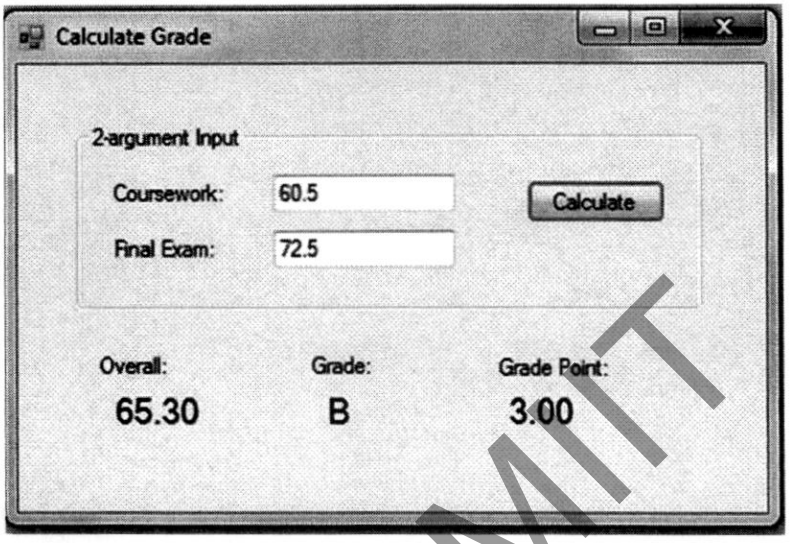


Figure 5

The following partial code for the *ClsGrade* class used in the program in Figure 5 is given in Figure 6.

```
class ClsGrade
{
    private double coursework, final, mark, gradepoint;
    private string grade;

   //Constructor
    public ClsGrade(double coursework, double final)
    {
        // Assume that the code to calculate
        // the overall mark, grade and grade point
        // has been written here ...
        // ..........
        // ..........
    }

  // Write the code to declare the properties of the class here
  ...
  // .........
  // .........
 }
```

Figure 6

a. Write the complete code to declare the properties of the *ClsGrade* class using the set and get accessors. It is given that the class has two input fields, namely *CourseworkMark* and *FinalMark*, and three output fields, namely *OverallMark*, *GradePoint* and *Grade*.

(15 marks)

b. Give the segment of the code for the method when the Calculate button shown in Figure 5 is clicked. The method must use the *ClsGrade* class based on the input from the text boxes. The partial code is given in Figure 7.

```
private ClsGrade grd;

private void btnCalculate_Click(object sender, EventArgs e)
{

}
```

Figure 7

(15 marks)

Question 7

Figure 8 shows a form to add student information to a SQL Server Express database named StudentDB.mdf. The related table is tblStudents consisting of varchar type columns Name, Email, StudentID, ICNumber, SexCode and StateCode.

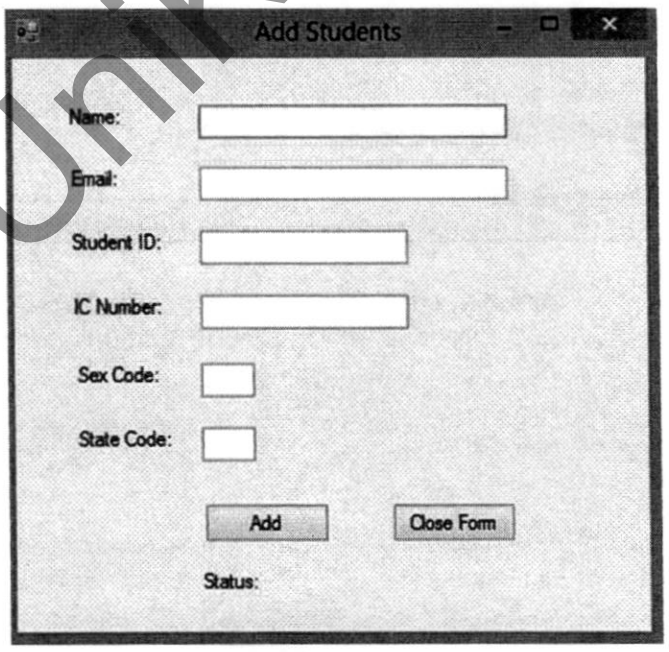


Figure 9

a. Write the code to create the SQL connection to the database.

(7 marks)

b. Using suitable names for the textboxes, write the code to create the SQL command to add input data from the textboxes into the database table.

(15 marks)

c. Incorporating the try-catch-finally exception handing statements, write the code to open connection, execute the SQL command and finally close the connection. Upon successful execution, display a message onto the *Status* label. If there is exception, an error message box will be shown.

(8 marks)

END OF QUESTION