**2018**

**MCA**

(December)

**Course No: 305**

**(GUI Programming using VB.NET)**

**(Practical)**

Full Marks: 75

Time: 3 Hours

*The figures in the margin indicate full marks for the questions*

Attempt any ***two*** questions from **UNIT I** and any ***one*** from **UNIT II**

**UNIT I**

1. Design a Windows Form Application with proper windows controls and add a class *SentenceTest* to do the following functions on an input sentence.
2. Count the occurrences of each word in a sentence.
3. Display the shortest and the longest words in a sentence. (20)
4. Using a Windows Form Application, create the following controls dynamically and display the prime and non-prime numbers from 1 to *n*.
5. Three labels
6. A textbox to take a range of a number *n* as an input from the user.
7. Two listboxes, one will be used to display prime numbers and the other for non-prime numbers.
8. A button. (20)
9. Design a MDI windows application with a proper MenuStrip options *File* and *Format* with a RichTextBox control. The File option contains sub options *Open*, *Save As* and *Exit*. When user clicks on Open, OpenFileDialog box should be displayed with filters for only (\*.txt) to open an existing Text file in the RichTextBox, Save As should display SaveAsDialog box to save the a file with a new file name and Exit to quit from the application. Format option should display a FontDialog box that will be used to format the text in a file. After formatting a text, the changes should be saved with a new file name. (20)

**UNIT II**

1. Create a database in Microsoft Access called Messages.accdb/Messages.mdb that has a *Message* table. The table consists of the following fields:

* *MessageId* (Number) is a unique id for every message
* *Sender\_Name* (Text) is the name of the sender
* *Receiver\_Name* (Text) is the name of the receiver
* *Msg* (Text) is a message in simple text form

Design an MDI Windows Application with proper user input validation, which will provide the following functions:

1. Add new information into the table where the MessageId is generated.
2. Delete existing information based on the selected MessageId.
3. The application should also display the information of the selected MessageId before deletion.
4. Update an existing record based on unique MessageId.
5. Display all the message information of a particular sender & receiver. (35)
6. Design an ASP.Net Web Service that has two web methods. The methods should return a boolean value and accept an integer number as argument. The first method should determine if the number is a Strong Number and the second method should determine if the number is a Vampire Number.

Strong number is a number whose sum of factorial of digits is equal to the given number. For example, 145 is a strong number since, 1! +4! +5! =1+24+120=145.

Vampire number is a composite natural number with even number of digits, which can be broken down into two numbers of equal lengths, and the multiplication of those two numbers will be equal to the original number. For example, 1260 (4 digit number) can be broken down into (12, 60), (21, 60), (61, 20) and so on, the order of the numbers is not decided, one random order is to be chosen. The pair which can give the original number is known as **fangs.** In this case, order (21, 60) are the fangs because their product will get **21 \* 60** = **1260** (the original number) hence the above number is vampire number. (35)

***Evaluation Criteria:***

*Input/Output Interface Design: 20%*

*Logic: 40%*

*Completion: 20%*

*Result: 20%*