97694

BCA 6th Semester
Examination – November , 2018

INTRODUCTION TO .NET

Paper : BCA-309

Time : Three Hours]   *[Maximum Marks : 80*

Before answering the questions, candidates should ensure that they have been supplied the correct and complete question paper. No complaint in this regard, will be entertained after examination.

Note : Attempt *five* questions in all by selecting *one* question from each unit. Question No. **1** is *compulsory*. All questions carry equal marks.

1. (a) What is Common Type System ?   2 × 8 = 16

   (b) What are sealed classes ?

   (c) What is function overriding ?

   (d) Why C# is more object-oriented ?

   (e) What is metadata in .NET ?

   (f) What are constructors in C# ?

   (g) What are delegates ?

   (h) What are interfaces in C# ?

UNIT – I

2. How .NET technology differs from COM ? What is .NET framework ? What is the relationship between .NET framework and COM/COM+/DCOM ? Illustrate the importance of its building blocks through a diagram.   16

3. Explain the following :

(a) Architecture of .NET platform   8

(b) Namespaces in .NET   8

UNIT – II

4. (a) What is C# ? What are its characteristics ? Explain.

8

(b) What are various data types supported by C# ? Illustrate.   8

5. Explain the following :
(a) Boxing and Unboxing   8
(b) Class Libraries in .NET   8

UNIT – III

6. What are different types of operators supported by C# ? Chart these operators, discuss their precedence and associativity through suitable examples.   16

7. Explain the following :

(a) Operator overloading   8

(b) for and for each loops   8

UNIT – IV

8. What is exception handling in C# ? What are the major tasks involved in handling exceptions ? What are the exceptions that occur commonly in C# programs ? Answer the following :   16

(i) Why is proper ordering of catch blocks necessary in C# ?

(ii) What happens when an exception is caused in an inner try block of a nested try block ?

(iii) How exception-handling mechanisms can be used for debugging a program ?

9. Explain the following :

(a) Automatic memory management   8

(b) Abstract class and methods   8