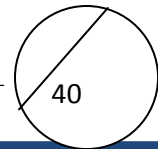


Iteration #: _____ Date: _____

Duration: 60 mins

Name: _____

EMP ID: _____



1. Which of the following statements are TRUE about the .NET CLR?

1. It provides a language-neutral development & execution environment.
2. It ensures that an application would not be able to access memory that it is not authorized to access.
3. It provides services to run "managed" applications.
4. The resources are garbage collected.
5. It provides services to run "unmanaged" applications.

- A. Only 1 and 2
- B. Only 1, 2 and 4
- C. 1, 2, 3, 4
- D. Only 4 and 5
- E. Only 3 and 4

2. Which of the following statements is correct about the C#.NET code snippet given below?

```
Namespace ConsoleApplication
{
    class Sample
    {
        public int func()
        {
            return 1;
        }
        public Single func()
        {
            return 2.4f ;
        }
    }
    class Program
    {
        static void Main(string[ ] args)
        {
            Sample s1 = new Sample();
            int i;
            i = s1.func();
            Single j;
            j = s1.func();
        }
    }
}
```

- A. func() is a valid overloaded function.
- B. Overloading works only in case of subroutines and not in case of functions.
- C. func() cannot be considered overloaded because: return value cannot be used to distinguish between two overloaded functions.

- D. The call to `i = s1.func()` will assign 1 to `i`.
 - E. The call `j = s1.func()` will assign 2.4 to `j`.
3. Which of the following statements is correct about constructors in C#.NET?
- A. A constructor cannot be declared as `private`.
 - B. A constructor cannot be overloaded.
 - C. A constructor can be a static constructor.
 - D. A constructor cannot access static data.
 - E. this reference is never passed to a constructor.
4. Which of the following statements is correct about the C#.NET code snippet given below?

```
class Sample
{
    private inti;
    public Single j;
    private void DisplayData()
    {
        Console.WriteLine(i + " " + j);
    }
    public void ShowData()
    {
        Console.WriteLine(i + " " + j);
    }
}
```

- A. x
 - B. `DisplayData()` cannot be declared as `private`.
 - C. `DisplayData()` cannot access `j`.
 - D. `ShowData()` cannot access to `i`.
 - E. There is no error in this class.
5. Which of the following statements is correct about the C#.NET code snippet given below?

```
inti;
int j = new int();
i = 10;
j = 20;
String str;
str = i.ToString();
str = j.ToString();
```

- A. This is a perfectly workable code snippet.
- B. Since `int` is a primitive, we cannot use `new` with it.
- C. Since an `int` is a primitive, we cannot call the method `ToString()` using it.
- D. `i` will get created on stack, whereas `j` will get created on heap.

E. Both i and j will get created on heap.

6. Which of the following statements are correct about the C#.NET code snippet given below?

```
namespace ConsoleApplication
{
    class Sample
    {
        int i, j;
        public void SetData(int ii, intjj)
        {
            this.i = ii;
            this.j = jj
        }
    }
    class MyProgram
    {
        static void Main(string[] args)
        {
            Sample s1 = new Sample();
            s1.SetData(10, 2);
            Sample s2 = new Sample();
            s2.SetData(5, 10);
        }
    }
}
```

- A. The code will not compile since we cannot explicitly use this.
- B. Using this in this program is necessary to properly set the values in the object.
- C. The call to SetData() is wrong since we have not explicitly passed the this reference to it.
- D. The definition of SetData() is wrong since we have not explicitly collected the this reference.
- E. Contents of this will be different during each call to SetData().

7. Which of the following is the correct way to find out the index of the second 's' in the string "She sells sea shells on the sea-shore"?

A.

```
String str = "She sells sea shells on the sea-shore";
int i;
i = str.SecondIndexOf("s");
```

B.

```
String str = "She sells sea shells on the sea-shore";
int i, j;
i = str.FirstIndexOf("s");
j = str.IndexOf("s", i + 1);
```

C.

```
String str = "She sells sea shells on the sea-shore";
int i, j;
i = str.IndexOf("s");
j = str.IndexOf("s", i + 1);
```

D.

```
String str = "She sells sea shells on the sea-shore";
```

```
inti, j;  
i = str.LastIndexOf("s");  
j = str.IndexOf("s", i - 1);
```

E.

```
String str = "She sells sea shells on the sea-shore";  
inti, j;  
i = str.IndexOf("S");  
j = str.IndexOf("s", i);
```

8. Which of the following statements is correct?
- A. Static methods can be a virtual method.
 - B. Abstract methods can be a virtual method.
 - C. It is necessary to override a virtual method.
 - D. When overriding a method, the names and type signatures of the override method must be the same as the virtual method that is being overridden.
 - E. We can override virtual as well as non-virtual methods.
9. Which of the following are necessary for Run-time Polymorphism?
- 1. The overridden base method must be virtual, abstract or override.
 - 2. Both the override method and the virtual method must have the same access level modifier.
 - 3. An override declaration can change the accessibility of the virtual method.
 - 4. An abstract inherited property cannot be overridden in a derived class.
 - 5. An abstract method is implicitly a virtual method.
- A. 1, 3
 - B. 1, 2, 5
 - C. 2, 3, 4
 - D. 4 only
10. Which of the following statements should be added to the subroutine fun() if the C#.NET code snippet given below is to output 9 13?

```
class BaseClass  
{  
    protected inti = 13;  
}  
class Derived: BaseClass  
{  
    inti = 9;  
    public void fun()  
    {  
        // [*** Add statement here ***]  
    }  
}
```

- A. Console.WriteLine(base.i + " " + i);
- B. Console.WriteLine(i + " " + base.i);

- C. Console.WriteLine(mybase.i + " " + i);
- D. Console.WriteLine(i + " " + mybase.i);
- E. Console.WriteLine(i + " " + this.i);

11. Which of the following statements is correct about the C#.NET code snippet given below?

```
int a = 10;
int b = 20;
int c = 30;
enum color: byte
{
    red = a,
    green = b,
    blue = c
}
```

- A. Variables cannot be assigned to enum elements.
- B. Variables can be assigned to any one of the enum elements.
- C. Variables can be assigned only to the first enum element.
- D. Values assigned to enum elements must always be successive values.
- E. Values assigned to enum elements must always begin with 0.

12. Which of the following CANNOT be used as an underlying datatype for an enum in C#.NET?

- A. byte
- B. short
- C. float
- D. int

13. Explain the difference between overriding and overloading. (3 marks).

14. Which of the following statements are correct about the C#.NET code snippet given below?

```
namespace ConsoleApplication
(
class Sample
{
    private enum color : int
    {
        red,
        green,
        blue
    }
    public void fun()
    {
        Console.WriteLine(color.red);
    }
}
class Program
{
    static void Main(string[ ] args)
    {
```

```
        // Use enum color here  
    }  
}  
}
```

-
1. To define a variable of type enum color in Main(), we should use the statement,color c; .
 2. enum color being private it cannot be used in Main().
 3. We must declare enum color as public to be able to use it outside the class Sample.
 4. To define a variable of type enum color in Main(), we should use the statement,Sample.color c; .
 5. We must declare private enum color outside the class to be able to use it inMain().

A. 1, 2, 3

B. 2, 3, 4

C. 3, 4

D. 4, 5

15. Which of the following statements are the correct way to call the method Issue() defined in the code snippet given below?

```
namespace College
{
    namespace Lib
    {
        class Book
        {
            public void Issue()
            {
                // Implementation code
            }
        }
        class Journal
        {
            public void Issue()
            {
                // Implementation code
            }
        }
    }
}
```

```
College.Lib.Book b = new College.Lib.Book();
b.Issue();
Book b = new Book();
b.Issue();
using College.Lib;
Book b = new Book();
b.Issue();
using College;
Lib.Book b = new Lib.Book();
b.Issue();
using College.Lib.Book;
Book b = new Book();
b.Issue();
```

- A. 1, 3
- B. 2, 4
- C. 3
- D. 4, 5

16. Which of the following statements is correct about namespaces in C#.NET?

- A.** Namespaces can be nested only up to level 5.
- B.** A namespace cannot be nested.
- C.** There is no limit on the number of levels while nesting namespaces.
- D.** If namespaces are nested, then it is necessary to use using statement while using the elements of the inner namespace.
- E.** Nesting of namespaces is permitted, provided all the inner namespaces are declared in the same file.

17. Which of the following statements is correct about the using statement used in C#.NET?

- A. using statement can be placed anywhere in the C#.NET source code file.
- B. It is permitted to define a member at namespace level as a using alias.
- C. A C#.NET source code file can contain any number of using statement.
- D. By using using statement it is possible to create an alias for the namespace but not for the namespace element.
- E. By using using statement it is possible to create an alias for the namespace element but not for the namespace.

18. Which of the following CANNOT belong to a C#.NET Namespace?

- A. class
- B. struct
- C. enum
- D. Data
- E. interface

19. Which of the following is the correct way to define a variable of the type structEmp declared below?

```
structEmp
{
    private String name;
    private int age;
    private Single sal;
}
```

- 1. Emp e(); e = new Emp();
- 2. Emp e = new Emp;
- 3. Emp e; e = new Emp;
- 4. Emp e = new Emp();
- 5. Emp e;

- A. 1, 3
- B. 2, 5
- C. 4, 5
- D. 1, 2, 4

20. Which of the following statements are correct about the structure declaration given below?

```
struct Book
{
    private String name;
    protected int totalpages;
    public Single price;
    public void Showdata()
    {
        Console.WriteLine(name + " " + totalpages + " " + price);
    }
    Book()
    {
        name = " ";
        totalpages = 0;
        price = 0.0f;
    }
}
Book b = new Book();
```


- 4. ArrayList
- 5. HashTable

- A.** 1 and 2 only
- B.** 3 and 4 only
- C.** 5 only
- D.** 1, 2 and 5 only
- E.** All of the above

25. For the code snippet shown below, which of the following statements are valid?

```
public class Generic<T>
{
    public T Field;
    public void TestSub()
    {
        T i = Field + 1;
    }
}
class MyProgram
{
    static void Main(string[] args)
    {
        Generic<int> gen = new Generic<int>();
        gen.TestSub();
    }
}
```

- A.** Addition will produce result 1.
- B.** Result of addition is system-dependent.
- C.** Program will generate run-time exception.
- D.** Compiler will report an error: Operator '+' is not defined for types T and int.
- E.** None of the above.

26. For the code snippet shown below, which of the following statements are valid?

```
public class Test
{
    public void TestSub<M> (M arg)
    {
        Console.WriteLine(arg);
    }
}
class MyProgram
{
    static void Main(string[] args)
    {
        Test test = new Test ();
        test.TestSub("Test ");
        test.TestSub(4.2f);
    }
}
```

}

- A. Program will compile and on execution will print: Test 4.2
- B. A non generic class Hello cannot have generic subroutine.
- C. Compiler will generate an error.
- D. Program will generate a run-time exception.
- E. None of the above.

27. For the code snippet given below, which of the following statements are valid?

```
public class MyContainer<T> where T: IComparable
{
    // Insert code here
}
```

- 1. Class MyContainer requires that it's type argument must implement IComparable interface.
- 2. Type argument of class MyContainer must be IComparable.
- 3. Compiler will report an error for this block of code.
- 4. This requirement on type argument is called as constraint.

- A. 1 and 2 Only
- B. 1, 2 and 3 Only
- C. 1 and 4 Only
- D. All of the above
- E. None of the above

28. Which of the following statements is valid about advantages of generics?

- A. Generics shift the burden of type safety to the programmer rather than compiler.
- B. Generics require use of explicit type casting.
- C. Generics provide type safety without the overhead of multiple implementations.
- D. Generics eliminate the possibility of run-time errors.
- E. None of the above.