Question Results

***Score 1.00 of 1***

1. Which utility improves the performance of managed applications through use of native images?

|  |  |
| --- | --- |
|  | gacutil |
| Correct | ngen |
| **Feedback:**  Used to compile managed assemblies (Dlls and Exes) into native code. |
|  | sn |
|  | ildasm |

***Score 1.00 of 1***

2. Which of the following is not a challenges for COM?  
a. DLL Hell  
b. Reference counting  
c. Message pumping  
d. Language independence

|  |  |
| --- | --- |
|  | DLL Hell |
|  | Reference counting |
|  | Message pumping |
| Correct | Langauge independence |
| **Feedback:**  COM is a binary interface standard that provides a language neutral way of implementing software components and also enables interoperation |

***Score 1.00 of 1***

3. What is true about Managed Code(MC)?

|  |  |
| --- | --- |
|  | Managed code(MC) is compiled by the JIT(Just In Time) compilers. |
|  | Managed code(MC) is where resources are Garbage Collected(GC) |
|  | Managed code(MC) runs on top of Windows OS. |
| Correct | Managed code(MC) is written to target the services of the Common Language Runtime (CLR). |
| **Feedback:**  Any .NET based compiler generates an MSIL. Microsoft Intermediator Language targets a .NET runtime. |

***Score 1.00 of 1***

4. Parallel computing is a new feature of .NET framework 4.0

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  Parallel computing is a new programming model for writing multithreaded and asynchronous code that greatly simplifies the work of application and library developers and this was introduced in .NET framework 4.0 |
|  | No |

***Score 1.00 of 1***

5. Rapid application development is one of the benefits of using Visual Basic as a programming langauges

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  Visual basic has a graphical user interface that enables fast paced development |
|  | No |

***Score 1.00 of 1***

6. Visual Basic is used by developers to build:

|  |  |
| --- | --- |
|  | GUI applications |
|  | Non-GUI applications |
| Correct | Both of the above |
| **Feedback:**  Visual basic enabled rapid application development because it is a GUI based programming language |
|  | None of the above |

***Score 1.00 of 1***

7. Choose the correct statement for Component Object Model (COM)

|  |  |
| --- | --- |
|  | COM is a framework to address the issue of backward compatibility |
|  | COM is a programming language that enables language independence |
| Correct | COM is a binary standard that provides a language neutral way of implementing software components |
| **Feedback:**  COM is a binary interface standard that address the issue of interoperability amonst Microsoft programming languages |
|  | None of the above |

***Score 1.00 of 1***

8. Thread management is not one of the core services provided by Common Language Runtime

|  |  |
| --- | --- |
|  | Yes |
| Correct | No |
| **Feedback:**  Thread management is one of the core services provided by CLR. The other core services include code compilation, memory allocation, garbage collection etc., |

***Score 1.00 of 1***

9. The service of CLR that enables all .NET languages to be converted to MSIL

|  |  |
| --- | --- |
|  | Common Type System |
|  | Intermediate Code |
|  | Component Object Model |
| Correct | Common Language Specification |
| **Feedback:**  Common Language Specification ensures that all .NET programming languages are converted to MSIL and makes them language neutral. |

***Score 1.00 of 1***

10. CLR converts intermediate language to

|  |  |
| --- | --- |
|  | MSIL |
| Correct | machine code |
| **Feedback:**  CLR converts all MSIL to machine code so that the operating system understand the code for execution |
|  | VB code |
|  | decimal code |

***Score 1.00 of 1***

11. The service of CLR that defines how types are declared, used and managed in the CLR is

|  |  |
| --- | --- |
| Correct | Common Type System |
| **Feedback:**  Common Type System defines how types are used by CLR and provides runtime support for cross-language interoperability |
|  | Common Language Specification |
|  | Code compilation |
|  | All of the above |

***Score 1.00 of 1***

12. Choose the correct statement with respect to Garbage Collector

|  |  |
| --- | --- |
| Correct | Garbage collector is the memory manager in CLR. |
| **Feedback:**  Garbage Collector is the automatic memory manager of CLR and manages the memory on the managed heap and helps reclaim unused memory |
|  | Garbage collector is the code compiler in CLR |
|  | Garbage collector converts the code to native code |
|  | None of the above |

***Score 1.00 of 1***

13. .NET tools enable developers in which of the functions  
a. Create  
b. Debug  
c. Test  
d. Deploy  
e. Manage applications

|  |  |
| --- | --- |
|  | a, b, c only |
|  | a, b, d only |
|  | b, d, e only |
| Correct | a, b, c, d, e |
| **Feedback:**  .NET tools are utility programs that enable developers to create, debug, test, deploy, and manage applications that target the .NET framework easily. |

***Score 1.00 of 1***

14. Choose the correct use of the tool llasm.exe

|  |  |
| --- | --- |
|  | To view and manipulate the contents of the global assembly cache |
|  | Improves the performance of managed applications through the use of native images |
| Correct | To generate a portable executable (PE) file from intermediate language |
| **Feedback:**  llasm.exe generates a portable executable (PE) file from intermediate language |
|  | None of the above |

***Score 1.00 of 1***

15. Every version of .NET has a final release but need not have a beta release

|  |  |
| --- | --- |
|  | Yes |
| Correct | No |
| **Feedback:**  Every version of .NET has a beta release and a final release |

Question Results

***Score 1.00 of 1***

1. What is the output of following set of code ?  
int a,b;  
  
a = (b = 10) + 5;

|  |  |
| --- | --- |
|  | b=10; a=5 |
|  | b=15; a=5 |
| Correct | a=15, b=10 |
| **Feedback:**  As () are used. First expression evaluates to (b=10). Second expression evaluates to a = 10 + 5 |
|  | a=10; b=10 |

***Score 1.00 of 1***

2. Predict the solution for following set of code.

**static void Main(string[] args)  
{  
int a, b, c, x;  
a = 90;  
b = 15;  
c = 3;  
x = a - b / 3 + c \* 2 - 1;  
Console.WriteLine(x);  
Console.ReadLine();  
}**

|  |  |
| --- | --- |
|  | 92 |
|  | 89 |
| Correct | 90 |
| **Feedback:**  The evaluation process followed is (\*, /, (-, +)-left to right): x = a - b / 3 + c \* 2 - 1; x = a - b / 3 + 6 - 1; x = a - 5 + 6 - 1; x = a - 5 + 6 - 1; x = 85 + 6 - 1; x = 91 - 1; x = 90; |
|  | 88 |

***Score 1.00 of 1***

3. The correct way of incrementing the operators are :

|  |  |
| --- | --- |
|  | ++ a ++ |
|  | b ++ 1 |
| Correct | c += 1 |
| **Feedback:**  This += is known as short hand operator which is same as variable = variable +1 .Similarly, a-= 1 is a = a-1, a\*=1 is a = a \* 1. They are used to make code short and efficient. |
|  | d =+ 1 |

***Score 1.00 of 1***

4. Which of the following is/are not Relational operators in C# ?

|  |  |
| --- | --- |
|  | >= |
|  | != |
| Correct | Not |
| **Feedback:**  C# doenot have keyword called Not, it support != operators |
|  | <= |

***Score 1.00 of 1***

5. Select the relevant values been assinged for set of code :  
m = 5;  
int y;  
y = m++;  
y = ++m;

|  |  |
| --- | --- |
|  | y = 5, m = 6 ; y = 5, m = 5 |
|  | y = 6, m = 6; y = 7, m = 6 |
| Correct | y = 5, m = 6; y = 7, m = 7 |
| **Feedback:**  step 1 : m = 5, y = m++ i.e y =5 ,m =6. step 2 : y = ++m , Since m = 6 .So, m = 7 on ++m and hence y = 7. Output : y = 5, m = 6; y =7 , m = 7. |
|  | y = 5, m = 6; y = 7, m = 8 |

***Score 1.00 of 1***

6. Select the output for following set of Code:

**static void Main(string[] args)  
{  
char ch = 'p';  
switch (ch)  
{  
case 'p':  
Console.WriteLine("coco" + "\t" + Convert.ToInt32(ch));  
break;  
default:  
Console.WriteLine("default");  
break;   
}  
Console.WriteLine("main");  
}**

|  |  |
| --- | --- |
|  | coco main |
|  | coco 112 |
| Correct | coco 112 main |
| **Feedback:**  ASCII value of ‘p’ is 112.Hence, coco 112 main. |
|  | compile time error |

***Score 1.00 of 1***

7. Select the output for following set of code :

**static void Main(string[] args)  
{  
int i;  
for (i = 0; ; )  
{  
Console.WriteLine("hello");  
}  
Console.ReadLine();  
}**

|  |  |
| --- | --- |
|  | No output |
|  | hello |
| Correct | hello printed infinite times |
| **Feedback:**  Testing condition for the loop is absent.So,loop will continue executing. |
|  | Syntax error |

***Score 1.00 of 1***

8. When a jump statement is used and the execution leaves a block, all local objects created in that scope are destroyed (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  A jump statement changes execution flow from its normal sequence and all the objects created in that scope are destroyed when the execution leaves the block |
|  | No |

***Score 1.00 of 1***

9. Select the output for following set of code :

**{  
int i; Console.WriteLine("Hi");  
for (i = 1; i <= 10; i++)  
Program.Main(args);  
Console.ReadLine();  
}**

|  |  |
| --- | --- |
|  | Prints ‘Hi’ for one time |
|  | Prints ‘Hi’ for infinite times |
| Correct | Stack overflow exception Condition generated |
| **Feedback:**  Calling of ‘Main()’ inside for loop generates Stack overflow exception |
|  | None of the above |

***Score 1.00 of 1***

10. Nullable type is a feature of which version of C#?

|  |  |
| --- | --- |
| Correct | 2.0 |
| **Feedback:**  Nullable types were first introduced in C# 2.0. apart from this the 2.0 also introduced generics, partial types, anonymous methods, iterators amonst others. |
|  | 3.0 |
|  | 4.0 |
|  | None of the above |

***Score 1.00 of 1***

11. Which of the following are the programming features of C#?

|  |  |
| --- | --- |
|  | Delegates and events |
|  | Properties and indexers |
|  | Custom Library creation |
| Correct | All of the above |
| **Feedback:**  C# supports all of these features |

***Score 1.00 of 1***

12. Select output of given set of Code

**static void Main(string[] args)  
{  
string name = "Dr.John";  
Console.WriteLine("Good Morning" + name);  
}**

|  |  |
| --- | --- |
|  | Dr. John |
|  | Good Morning |
| Correct | Good Morning Dr. John |
| **Feedback:**  Intialize a string variable using '=' and concatenate string using ‘+’ operator. Output:Good Morning Dr.John |
|  | Good Morning name |

***Score 1.00 of 1***

13. Which function supports conversion of value type to reference type with string type?

|  |  |
| --- | --- |
| Correct | valueType.ToString() |
| **Feedback:**  conversion of value type to reference type is called 'boxing' and is automatic when used with object type. However with the string type, the function ToString() achieves this purpose |
|  | valueType.ToInt() |
|  | valueType.ToChar |
|  | None of the above |

***Score 1.00 of 1***

14. String type is a reference type (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  String type resides on the heap and hence is areference data type. Apart from string, the other reference data types are object, class, interface and arrays |
|  | No |

***Score 1.00 of 1***

15. Which of the following operators are used for comparing two entities?

|  |  |
| --- | --- |
|  | Conditional |
|  | Logical |
| Correct | Equality |
| **Feedback:**  == and != are the two equality operators that are used to perform comparison between two entities |
|  | Logical |

***Score 1.00 of 1***

16. in C# a variable need not have a data type (Yes / No)

|  |  |
| --- | --- |
|  | Yes |
| Correct | No |
| **Feedback:**  Variables hold different values at different points of time in a program. Based on the value they hold, they should always have a datatype. |

***Score 1.00 of 1***

17. enum is a value data type (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  enum resides on the stack in the memory and hence is a value data type |
|  | No |

***Score 1.00 of 1***

18. The Default value of Boolean DataType is ?

|  |  |
| --- | --- |
|  | 2 |
|  | TRUE |
| Correct | FALSE |
| **Feedback:**  By definition, uninitialized member variables are automatically initialized to 0, null or false depending on the data type |
|  | 1 |

***Score 1.00 of 1***

19. Correct declaration and initialization of falues to variables ‘a’ and ‘b’?

|  |  |
| --- | --- |
|  | int a = 32, b = 40.6; |
|  | int a = 42; b = 40; |
| Correct | int a = 32; int b = 40; |
| **Feedback:**  This is the correct syntax fordecalation and initialization of a and b |
|  | int a = b = 42; |

***Score 1.00 of 1***

20. Which of the following is not infinite loop?

|  |  |
| --- | --- |
|  | for( ;’0′; ) |
| Correct | for(;;) |
| **Feedback:**  A for loop without any condition check results in a infinite loop |
|  | for( ;’1′; ) |
|  | for( ;1; ) |

Question Results

***Score 1.00 of 1***

1. What is the output of following set of code?

**static void Main(string[] args)  
{  
int a = 5;  
int s = 0, c = 0;  
Mul (a, ref s, ref c);  
Console.WriteLine(s + "t " +c);  
Console.ReadLine();  
}  
static void Mul (int x, ref int ss, ref int cc)  
{  
ss = x \* x;  
cc = x \* x \* x;  
}**

|  |  |
| --- | --- |
|  | 125 25 |
| Correct | 25t 125 |
| **Feedback:**  The value of variable a is passed by value while value of variable s and c is passed by reference. Output: 25 125. |
|  | Compile error |
|  | The value of variable a is passed by value while value of variable s and c is passed by reference. Output: 25 125. |

***Score 1.00 of 1***

2. In Method overloading, method signature is checked during

|  |  |
| --- | --- |
|  | Run time |
| Correct | Compile time |
| **Feedback:**  Method signature is checked at compile time during overloading as it undergoes early binding |
|  | Declaration of methods |
|  | Initiation of variables |

***Score 1.00 of 1***

3. Method signature does not consist of this

|  |  |
| --- | --- |
|  | Method name |
|  | Kind of parameters |
|  | No. of parameters |
| Correct | return type of the method |
| **Feedback:**  Method signature consists of the method name, type of parameters, kind of parameters and no. of parameters only and not the return type of the method |

***Score 1.00 of 1***

4. Method overloading is implemented using optional parameters (Yes / No)

|  |  |
| --- | --- |
|  | Yes |
| Correct | No |
| **Feedback:**  Method overloading is implemented without using optional parameters |

***Score 1.00 of 1***

5. Which of the following do not refer to Method overloading?

|  |  |
| --- | --- |
|  | Early binding |
|  | Static polymorphism |
| Correct | Late binding |
| **Feedback:**  Late binding is also called as dynamic binding or method overriding. |
|  | Static biding |

***Score 1.00 of 1***

6. A parameterized method provides reuse of functionality, processes data based on the arguments passed to it, and also returns the processed data (Yes/No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  According to defination ,the above statement is correct |
|  | No |

***Score 1.00 of 1***

7. To pass parameters by 'pass by out' method, the keyword used is

|  |  |
| --- | --- |
|  | ref |
|  | params |
| Correct | out |
| **Feedback:**  Use of out keyword when delcaring the function and calling the function ensures the parameters are passed by 'pass by out' method |
|  | val |

***Score 1.00 of 1***

8. The tehinque used to change value of the initialized parameters been passed to the function

|  |  |
| --- | --- |
|  | Pass by Value |
| Correct | Pass by Reference |
| **Feedback:**  Pass by reference passes the variable address to the calling function by the called function and hence uses the data of the parameters passed in the called function |
|  | Pass by Out |
|  | Pass by Params |

***Score 1.00 of 1***

9. The correct syntax of a parameterized method is

|  |  |
| --- | --- |
| Correct | <Access Modifier> <Return Type> <Method Name>(<Parameter1>, <Parameter2>, <…>) {  // Method Body for Processing data based on parameters  return <Processed Data>; } |
| **Feedback:**  According to syntax  <Access Modifier> <Return Type> <Method Name>(<Parameter1>, <Parameter2>, <…>) {  // Method Body for Processing data based on parameters  return <Processed Data>; } is the correct way to define the method |
|  | <Return Type> <Access Modifier> <Method Name>(<Parameter1>, <Parameter2>, <…>) {  // Method Body for Processing data based on parameters  return <Processed Data>; } |
|  | <Return Type> <Access Modifier> (<Parameter1>, <Parameter2>, <…><Method Name>) {  // Method Body for Processing data based on parameters  return <Processed Data>; } |
|  | None of the above |

***Score 1.00 of 1***

10. Overloading concept can also be applied to

|  |  |
| --- | --- |
| Correct | Operators |
| **Feedback:**  Overloading can be applied to operators to modify the functionality of the same operator |
|  | Interfaces |
|  | Assemblies |
|  | Namespaces |

Question Results

***Score 1.00 of 1***

1. Choose the correct statement about constructors in C#.

|  |  |
| --- | --- |
|  | Constructors cannot be overloaded |
|  | Constructors do not set default values |
|  | Constructors are explicitly called |
| Correct | Constructors have the same name as that of the class |
| **Feedback:**  By definition, constructors needs to have their name same as the class name, so the run time can identify the same. |

***Score 1.00 of 1***

2. Virtual is one of the valid C# modifier (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  The valid C# modifiers include Virtual, abstract, static, sealed, public, private, protected, internal and protected internal |
|  | No |

***Score 1.00 of 1***

3. When a member variable is declared as protected internal in the base class, the derived class can use this variable (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  When a member variable is declared as protected internal in the base class, this variable can be used by the containing class, the derived class and any program containing this class. |
|  | No |

***Score 0.00 of 1***

4. Properties can be declared in a namespace (Yes / No)

|  |  |
| --- | --- |
| Wrong | Yes |
| **Feedback:**  Properties can be declared in a class, struct or an interface only |
| Should have chosen | No |

***Score 1.00 of 1***

5. Choose the correct statement for properties

|  |  |
| --- | --- |
|  | Properties always need to have the get and set methods |
|  | Properties once set cannot be changed |
| Correct | Property functions cannot take any parameters |
| **Feedback:**  Properties are like a combination of a variable and a method and cannot take any parameters |
|  | Properties is the not a recommended way to access variable from inside the class |

***Score 1.00 of 1***

6. The modifier used to define a class whose objects cannot be created but acts as a base class for it’s derived once is?

|  |  |
| --- | --- |
|  | sealed |
|  | Static |
|  | new |
| Correct | abstract |
| **Feedback:**  Abstract class has atleast one abstract method(method with out body). Abstract class cannot be instantiated as the method is incomplete. |

***Score 1.00 of 1***

7. Choose the correct statement about interfaces

|  |  |
| --- | --- |
|  | Interfaces cannot be inherited |
|  | Interfaces consists of data static in nature and static methods |
|  | Interfaces consists of only method declaration |
| Correct | None of the above |
| **Feedback:**  All statemts are not correct w.r.t interfaces |

***Score 1.00 of 1***

8. A struct cannot declare a default constructor (constructor without parameters)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  Structs can declare constructors, but they must take parameters. It is an error to declare a default (parameterless) constructor for a struct. Struct members cannot have initializers. A default constructor is always provided to initialize the struct members to their default values. |
|  | No |

***Score 1.00 of 1***

9. The generation in GC where objects are long lived is

|  |  |
| --- | --- |
|  | Generation 0 |
|  | Generation 1 |
| Correct | Generation 2 |
| **Feedback:**  Generation 2 has long lived objects, Generation 1 is a buffer between short lived and long lived objects whereas Generation 0 is for short lived objects |
|  | None of the above |

***Score 1.00 of 1***

10. Which of these base class are accessible to the derived class members?

|  |  |
| --- | --- |
|  | static |
| Correct | protected |
| **Feedback:**  Protected members of classes are accessible to derived classes. |
|  | private |
|  | virtual |

***Score 1.00 of 1***

11. Which of the following cannot be specified for a destructor

|  |  |
| --- | --- |
|  | Name of destructor method |
|  | Body of destructor method |
| Correct | Return type |
| **Feedback:**  Destructors cannot have modifiers, parameters or return types |
|  | None of the above |

***Score 1.00 of 1***

12. The structure of a class contains which of the following

|  |  |
| --- | --- |
| Correct | methods, fields, properties |
| **Feedback:**  The structure of a class constains method which are the member functions, fields that are called member variables and properties that look like fields to the users |
|  | methods, fields, return type |
|  | methods, return type, operators |
|  | None of the above |

***Score 1.00 of 1***

13. How many objects of a same class can a program create

|  |  |
| --- | --- |
|  | 3 |
|  | 1 |
|  | 2 |
| Correct | Not Limited |
| **Feedback:**  A program can create as many objects of the same class as required depending upon the avaiable memory |

***Score 1.00 of 1***

14. Constructors are used to  
1. Set default and custom values  
2. Limit instantiation  
3. Write code that is flexible and easy to read  
4. Deallocate memory of variables

|  |  |
| --- | --- |
|  | 1, 2 only |
|  | 2, 3, 4 |
| Correct | 1,2, 3 |
| **Feedback:**  Constructors can be used to Stament 1, 2 , 3 |
|  | 1, 2, 4 |

***Score 1.00 of 1***

15. Following are the valid types of constructors:  
1. Instance  
2. Class  
3. Parameterized  
4. Virtual

|  |  |
| --- | --- |
| Correct | 1,2,3 |
| **Feedback:**  Constructors can be instance or class type. Instance constructors are further categorized as default or parameterized. |
|  | 1,2,4 |
|  | 2,3,4 |
|  | 2 and 4 |

***Score 1.00 of 1***

16. When an object of a class is created it calls which constructor

|  |  |
| --- | --- |
| Correct | default |
| **Feedback:**  A default constructor is triggered whenever an object of a class is created. |
|  | parameterized |
|  | instance |
|  | class |

***Score 1.00 of 1***

17. Special functions called by garbage collector used to clear unmanaged data of the class are called

|  |  |
| --- | --- |
|  | Constructors |
| Correct | Destructors |
| **Feedback:**  Destructors are the special functions that handle clearning of unmanaged data of the class |
|  | User defined methods |
|  | Parameterized functions |

***Score 1.00 of 1***

18. Desctructor is called by

|  |  |
| --- | --- |
|  | User |
|  | Constructors |
| Correct | Garbage collector |
| **Feedback:**  Destructors are called by garbage collectors to clear unmanaged data of the class. Destructors cannot be explicitly called |
|  | None of the above |

***Score 1.00 of 1***

19. Desctructors can be overloaded

|  |  |
| --- | --- |
|  | Yes |
| Correct | No |
| **Feedback:**  A class can have only one destructor and cannot overload the destructor, while a constructor can be overloaded |

***Score 1.00 of 1***

20. This process enables interactive applications to be more responsive by minimizing pauses for a garbage collection

|  |  |
| --- | --- |
| Correct | Ephemeral garbage collection |
| **Feedback:**  Concurrent garbage collection enables interactive applications to be more responsive by minimizing pauses for a collection. Managed threads can continue to run most of the time while the concurrent garbage collection thread is running. This results in shorter pauses while a garbage collection is occurring. |
|  | Concurrent garbage collection |
|  | Simultaneous garbage collection |
|  | None of the above |

Question Results

***Score 1.00 of 1***

1. A struct cannot declare a default constructor (constructor without parameters)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  Structs can declare constructors, but they must take parameters. It is an error to declare a default (parameterless) constructor for a struct. Struct members cannot have initializers. A default constructor is always provided to initialize the struct members to their default values. |
|  | No |

***Score 1.00 of 1***

2. The phase of GC that reclaims the space occupied by the dead objects and compacts the surviving objects is

|  |  |
| --- | --- |
| Correct | Marking phase |
|  | Relocating phase |
|  | Compacting phase |
|  | None of the above |

***Score 1.00 of 1***

3. One of the area of memory that CLR reservers when a process is initialized is

|  |  |
| --- | --- |
| Correct | Managed heap |
| **Feedback:**  Common Language Run-time reserves two areas of memory when a process is initialized - managed heap and application roots. |
|  | Managed stack |
|  | Application nodes |
|  | None of the above |

***Score 1.00 of 1***

4. Managed heap is logically segregated as

|  |  |
| --- | --- |
| Correct | Heaps |
|  | Generations |
|  | Stacks |
|  | None of the above |

***Score 1.00 of 1***

5. The generation in GC where objects are long lived is

|  |  |
| --- | --- |
|  | Generation 0 |
|  | Generation 1 |
| Correct | Generation 2 |
| **Feedback:**  Generation 2 has long lived objects, Generation 1 is a buffer between short lived and long lived objects whereas Generation 0 is for short lived objects |
|  | None of the above |

***Score 1.00 of 1***

6. Collect is a method of GC class (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  Collect is a method of the GC class and is used to free unused memory of objects. |
|  | No |

***Score 1.00 of 1***

7. Generations 0 and 1 are also known as

|  |  |
| --- | --- |
| Correct | Ephemeral generations |
| **Feedback:**  Because objects in generations 0 and 1 are short-lived, these generations are known as the ephemeral generations. |
|  | Peripheral generations |
|  | Rectangular generations |
|  | None of the above |

***Score 1.00 of 1***

8. This process enables interactive applications to be more responsive by minimizing pauses for a garbage collection

|  |  |
| --- | --- |
| Correct | Ephemeral garbage collection |
| **Feedback:**  Concurrent garbage collection enables interactive applications to be more responsive by minimizing pauses for a collection. Managed threads can continue to run most of the time while the concurrent garbage collection thread is running. This results in shorter pauses while a garbage collection is occurring. |
|  | Concurrent garbage collection |
|  | Simultaneous garbage collection |
|  | None of the above |

Question Results

***Score 1.00 of 1***

1. Variables in an array are called \_\_\_\_\_\_\_\_ of the array

|  |  |
| --- | --- |
| Correct | elements |
| **Feedback:**  Variables in an array are called elements and are referred to by the index number |
|  | literals |
|  | constants |
|  | None of the above |

***Score 1.00 of 1***

2. What is the core difference between an indexer and a property?

|  |  |
| --- | --- |
| Correct | The accessors for indexers take parameters |
| **Feedback:**  Indexers are similar to properties except that their accessors take parameters |
|  | The indexers can be read-write only |
|  | Indexers cannot be overloaded |
|  | None of the above |

***Score 1.00 of 1***

3. Choose the correct statement for an indexer  
a. Indexers can have more than one formal parameter  
b. Indexers can be overloaded  
c. Can be read-only, write-only, or read/write  
d. All of the above

|  |  |
| --- | --- |
| Correct | a |
|  | b |
|  | c |
|  | All of the above |

***Score 0.00 of 1***

4. The keyword used to define the value being assigned to the set indexer is

|  |  |
| --- | --- |
| Wrong | get |
|  | set |
| Should have chosen | value |
|  | this |

***Score 0.00 of 1***

5. Choose the correct syntax for declaring an indexer

|  |  |
| --- | --- |
| Wrong | this [ ] |
|  | this [ ] |
| Should have chosen | this [ ] |
|  | this |

***Score 0.00 of 1***

6. Choose the correct syntax for using an indexer

|  |  |
| --- | --- |
| Wrong | < ObjectName>[] = “value” |
|  | < ObjectName>[] = “value” |
|  | < ObjectName>[] = “” |
| Should have chosen | < ObjectName>[] = “value” |

***Score 1.00 of 1***

7. ArrayList is a collection class that has a fixed size (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
|  | No |

***Score 1.00 of 1***

8. Choose the correct statement for an arraylist class

|  |  |
| --- | --- |
| Correct | The ArrayList is not guaranteed to be sorted |
| **Feedback:**  By definition, an arraylist can sort if all the elements been added are default and similar types |
|  | The ArrayList implementation is provided by the System.Collections.Array class |
|  | ArrayList is used to store data of fixed size |
|  | All of the above |

***Score 1.00 of 1***

9. The function that adds an object to the end of the arraylist is

|  |  |
| --- | --- |
| Correct | Append() |
|  | Add() |
|  | Copy() |
|  | Insert() |

***Score 1.00 of 1***

10. Arraylist support what type of arrays as elements in its collection

|  |  |
| --- | --- |
| Correct | Single dimensional array |
| **Feedback:**  Arraylist does not support multi-dimensional and jagged arrays as its elements |
|  | Multi dimensional array |
|  | Jagged array |
|  | none of the above |

***Score 1.00 of 1***

11. ArrayList allows duplicate values (Yes/No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  ArrayList accepts null as a valid value and hence allows duplicate values in its collection. |
|  | No |

***Score 0.00 of 1***

12. In Events, the class that receives the event is

|  |  |
| --- | --- |
| Wrong | Publisher |
| Should have chosen | Subscriber |
|  | Receiver |
|  | Sender |

***Score 1.00 of 1***

13. Indexers allow instances of a class or struct to be used/indexed just like arrays (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  Indexers involve attaching a property to the class itself and when an indexer is defined for a class, the class behaves like a virtual array. |
|  | No |

***Score 1.00 of 1***

14. Choose the correct statement for an indexer

|  |  |
| --- | --- |
| Correct | Provides secure access to the private variables |
| **Feedback:**  Indexers provides secure access to the private variables via property that is connected to the class, instead of a member |
|  | Indexers are implemented using underlying public variables |
|  | When an indexer is defined for a class, the class behaves like a 2-d array |
|  | None of the above |

***Score 1.00 of 1***

15. Choose the operator/operators which is/are used to access the [] operator in indexers?  
a. get  
b. set  
c. this  
d. val

|  |  |
| --- | --- |
|  | b, c |
| Correct | a, b |
| **Feedback:**  The indexer is implemented through the get and set accessors for the [] operator |
|  | d, a |
|  | a, c |

***Score 1.00 of 1***

16. Choose the correct statement for array

|  |  |
| --- | --- |
| Correct | Elements in an array are allocated contiguously in the memory |
| **Feedback:**  Is a technic when help in refering multiple variables of the same type with a single name and using the index location. |
|  | Index of array elements starts with 1 |
|  | Array always has a fixed number of elements |
|  | All of the above |

***Score 1.00 of 1***

17. The elements of an array can be accessed using

|  |  |
| --- | --- |
|  | switch construct |
| Correct | loop |
| **Feedback:**  Array elements are sequential and is accessed using the index location, to access all the elements in an array an loop with a counter can be used. |
|  | conditional operator |
|  | if construct |

***Score 1.00 of 1***

18. Choose the incorrect statement for an array

|  |  |
| --- | --- |
|  | Individual elements of the array can be accessed using the indices |
|  | The elements can also be accessed using a foreach loop |
| Correct | An array always has one dimension |
| **Feedback:**  An array can have one or more dimensions. The dimension of the array is also known as rank of the array |
|  | All of the above |

***Score 1.00 of 1***

19. Choose the correct delcaration for a single dimensional array

|  |  |
| --- | --- |
| Correct | int[] numbers; |
| **Feedback:**  By definition, in C# the array syntax has the [] brackets before the variable name; |
|  | int numbers[]; |
|  | numbers[]; |
|  | int numbers; |

***Score 1.00 of 1***

20. An array with a rank of one is called

|  |  |
| --- | --- |
| Correct | Single dimensional array |
| **Feedback:**  For a single dimensional array, the dimension or rank is always 1 |
|  | multi dimensional array |
|  | jagged array |
|  | None of the above |

***Score 1.00 of 1***

21. Choose the correct statement for a multi dimensional array

|  |  |
| --- | --- |
|  | Also called rectangular arrays |
|  | The elements can be accessed using a foreach loop |
|  | Individual elements of the array can be accessed using the indices |
| Correct | All of the above |
| **Feedback:**  It can store details as simple records having multiple rows and columns. The elements in the array can be access using their respective row and column number. Sequential access of the rows and columns can be achieved using loops |

***Score 1.00 of 1***

22. In the declaration [x,y] , x and y specify?

|  |  |
| --- | --- |
|  | rank of the array |
| Correct | size of the array |
| **Feedback:**  By syntax, [] specifies the rank of the array, x and y specify the size of the array |
|  | dimensions of the array |
|  | none of the above |

***Score 1.00 of 1***

23. What is the output of the following code:  
string[,] address = new string[3,2];  
address[0,0] = “Bangalore”;  
address[0,1] = “560043”;  
  
address[1,0] = “Mumbai”;  
address[1,1] = “400001”;  
  
address[2,0] = “Chennai”;  
address[2,1] = “600001”;  
  
Console.WriteLine(“Value of Address[0,0]: ” + address[0,0]);

|  |  |
| --- | --- |
| Correct | Value of Address[0,0]: Bangalore |
| **Feedback:**  The value of the array element address[0,0] refers to 'Bangalore' and hence the output |
|  | Value of Address[0,0]: Mumbai |
|  | Value of Address[0,0]: 560043 |
|  | Value of Address[0,0]: 400001 |

***Score 1.00 of 1***

24. Multi-dimensional arrays are used to store data as multiple rows and columns in memory for ease of access and iteration (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  Is a technic when help in refering multiple variables of the same type with a single name and using the indexs. First index location helps in refering to the row location, second index helps in refering the column location. |
|  | No |

***Score 1.00 of 1***

25. Choose the correct statement for a jagged array

|  |  |
| --- | --- |
| Correct | The elements of a jagged array can be of different dimensions and sizes |
| **Feedback:**  A jagged array is an array of arrays and hence the elements can be of different dimensions and sizes |
|  | Before you can use jagged array, its elements need not be individually initialized |
|  | Individual elements of the array can be accessed using post fix operators like ++ |
|  | The elements cannot be accessed using loop |

***Score 0.00 of 1***

26. Before you can use jagged array, its elements must be individually initialized (Yes / No)

|  |  |
| --- | --- |
| Wrong | Yes |
| **Feedback:**  Each of the elements of the individual arrays in a jagged array should be initialized before using the jagged array else the compiler will return an error |
| Should have chosen | No |

***Score 1.00 of 1***

27. A public event cannot be raised from

|  |  |
| --- | --- |
| Correct | Outside a class declaration |
| **Feedback:**  A public event can not be raised from outside a class declaration whereas A public delegate can be raised from outside a class declaration |
|  | Inside a class declaration |
|  | Either of the two |
|  | Neither of the two |

Question Results

***Score 1.00 of 1***

1. Choose the incorrect statement for exceptions:

|  |  |
| --- | --- |
|  | Provides a way to transfer control from one part of a program to another when runtime errors occure |
| Correct | Synchronizes compile time errors |
| **Feedback:**  Exceptions synchronize run time errors and not compile time ones |
|  | Enables in building robust and more fault-tolerant programs |
|  | All of the above |

***Score 1.00 of 1***

2. Exception is an erroneous situation that occurs during \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  | program compilation |
|  | program creation |
| Correct | program execution |
| **Feedback:**  Exceptions is a technique to detect and respond to an unexpected circumstance that arises while a program is running |
|  | None of the above |

***Score 1.00 of 1***

3. Exception handling provides a structured way of handling\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  | system level errors |
|  | application level errors |
| Correct | Both of the above |
| **Feedback:**  Exception handling provides a structured and uniform way of handling system-level and application-level errors |
|  | None of the above |

***Score 1.00 of 1***

4. An exception can be generated by\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  | CLR |
|  | third party libraries |
|  | using throw keyword |
| Correct | All of the above |
| **Feedback:**  An exception can be generated either by the CLR, or by third-party libraries, or by the application code using the throw keyword |

***Score 1.00 of 1***

5. Which of the following is a benefit of exception

|  |  |
| --- | --- |
|  | Error-handling code is seperated from regular code |
|  | Propagates errors up the call stack |
|  | Error handling machanisim groups and differentiates diffrent error types |
| Correct | All of the above |
| **Feedback:**  Exception handling helps in designing a separate and a structured error handling code, and gives the application to exit gracefully in the event of error. |

***Score 1.00 of 1***

6. Choose the correct syntax for a custom exception

|  |  |
| --- | --- |
| Correct | class : System.Exception { ….} |
| **Feedback:**  According to syntax rule class : System.Exception { ….} is the correct syntax for defining custom exception class |
|  | : System.Exception { ….} |
|  | class { ….} |
|  | class System.Exception { ….} |

***Score 1.00 of 1***

7. A custom exception is thrown using a throw statement and handled using try catch statement (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  By definition, all user defined exceptions are raised using throw statement |
|  | No |

Question Results

***Score 1.00 of 1***

1. Choose the correct statement for Input / Output

|  |  |
| --- | --- |
| Correct | I/O is used for reading and writing to be performed on files, directories, or streams |
|  | I/O helps to get and set properties for files and directories |
|  | I/O helps to retrieve collections of files and directories based on search criteria |
|  | All of the above |

***Score 1.00 of 1***

2. System.IO namespace contain types that enable reading and writing to a storage medium in a \_\_\_\_\_ manner

|  |  |
| --- | --- |
| Correct | synchronously |
|  | asynchronously |
|  | secure manner |
|  | All of the above |

***Score 1.00 of 1***

3. Readers and Writers are one of the I/O categories (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  IO categories include Files and Directories, Streams, Readers and Writers, Compression, Isolated storage etc., |
|  | No |

***Score 1.00 of 1***

4. Stream is a class of I/O that

|  |  |
| --- | --- |
| Correct | Supports reading and writing as bytes to the storage media |
| **Feedback:**  Streams is an abstract class that supports reading and writing to the storage media and provides a common view for different data sources |
|  | Provide types for reading and writing encoded characters from streams |
|  | Handle the conversion of encoded characters to and from bytes |
|  | None of the above |

***Score 1.00 of 1***

5. The category of I/O that provides types for condensing and de-condensing of storage file is

|  |  |
| --- | --- |
| Correct | Files and Directories |
|  | Streams |
|  | Readers and Writers |
|  | Compression |

***Score 1.00 of 1***

6. Streamwriter is a type of File Writer (Yes / No)

|  |  |
| --- | --- |
| Correct | Yes |
|  | No |

***Score 1.00 of 1***

7. Choose the correct statement with respect to Serialization

|  |  |
| --- | --- |
| Correct | A saved object state can be converted back into active object using deserialization. |
|  | Selective serialization of object contents can also be done. |
|  | An object state can be converted to a stream of bytes that can be saved into memory buffer, file or even a database. |
|  | All of the above |

***Score 1.00 of 1***

8. In Selective Serialization, the attribute can be applied to specific fields that needs to be skipped in the serialized output is

|  |  |
| --- | --- |
| Correct | Serialized attribute |
|  | Non-seriazlied attribute |
|  | Both of the above |
|  | None of the above |

**Question Results**

**Score 1.00 of 1**

**Question:**

**All .NET languages are compiled as assemblies in the form of**

**Response:**

|  |  |
| --- | --- |
|  | dll only |
|  | exe only |
| Correct | both of the above |
| **Feedback:**  All .NET languages are compiled as assemblies (exe/dll) by their respective .NET language compiler and they are in MSIL format that is only understood by the CLR |
|  | None of the above |

**Score 1.00 of 1**

**Question:**

**Choose the correct statement for .NET assemblies**

**Response:**

|  |  |
| --- | --- |
| Correct | Provides type information for the CLR |
| **Feedback:**  .NET assemblies are in MSIL format and they are a collection of types and resources that are built to work together and form a logical unit of functionality while they also provide type information for the CLR |
|  | All .NET languages are compiled as classes |
|  | Assemblies are in decimal format |
|  | None of the above |

**Score 1.00 of 1**

**Question:**

**Assemblies can contain only one module (Yes / No)**

**Response:**

|  |  |
| --- | --- |
|  | Yes |
| Correct | No |
| **Feedback:**  Assemblies can contain more than one module |

**Score 1.00 of 1**

**Question:**

**Choose the correct statement for a private assembly**

**Response:**

|  |  |
| --- | --- |
| Correct | Must be designed to work side-by-side with other versions of the assembly on the system |
| **Feedback:**  Thye reside at the location where the client assembly resides. They help in quicker access to the resources by the client assembly. They do not conflict with assemblies present in other location under the same machine. |
|  | Assembly manifest will not be included in the DLL as a resource (default) |
|  | Needs to be assigned a strong name |
|  | Cannot be used to create isolated applications |

**Score 1.00 of 1**

**Question:**

**Version number is a part of the manifest structure in an assembly (Yes / No)**

**Response:**

|  |  |
| --- | --- |
| Correct | Yes |
| **Feedback:**  The Manifest structure in an assembly contains the assembly name, version number, culture and strong name |
|  | No |

Question Results

***Score 1.00 of 1***

1. Correct statement w.r.t the following code snippet:  
List l1=new List

|  |  |
| --- | --- |
|  | Above code reports compile time error |
|  | Above code generates run time error |
| Correct | Code works fine, No error in the code |
| **Feedback:**  That's correct! At runtime, string is converted to object type. |
|  | Cannot predict |

***Score 1.00 of 1***

2. State whether the following statement is true or false.  
IEnumerable is a parent interface of all the classes in System.Collections.Generic.

|  |  |
| --- | --- |
| Correct | TRUE |
| **Feedback:**  That's correct! All classes in System.Collection.Generic will implement the interface IEnamurable. |
|  | FALSE |

***Score 1.00 of 1***

3. A SortedList maintains a collection of names of states and capital city of each state. Which of the following is used to find out whether the state of “Kashmir” is present in the collection?

|  |  |
| --- | --- |
|  | t.HasValue(“Kashmir”); |
| Correct | t.ContainsKey(“Kashmir”); |
| **Feedback:**  That's correct! ContainsKey() method of the SortedList class returns either true or false depending on the availibility of the key. |
|  | t.HasKey(“Kashmir”); |
|  | t.ContainsValue(“Kashmir”); |

***Score 1.00 of 1***

4. \_\_\_\_\_\_\_\_\_\_ is used to set the capacity to the actual number of elements in the SortedList.

|  |  |
| --- | --- |
|  | Trim() |
|  | Clear() |
| Correct | TrimToSize() |
| **Feedback:**  That's correct! TrimToSize() is used to set the capacity to the actual number of elements in the SortedList. |
|  | SetCapacity() |

***Score 1.00 of 1***

5. Each Element in the BitArray is of \_\_\_\_\_\_\_\_\_\_ data type.

|  |  |
| --- | --- |
|  | int |
| Correct | bool |
| **Feedback:**  That's correct! BitArray is a collection of bits. |
|  | byte |
|  | object |

***Score 1.00 of 1***

6. The method used to set all bits in the BitArray is \_\_\_\_\_\_\_\_\_\_.

|  |  |
| --- | --- |
|  | Set() |
| Correct | SetAll() |
| **Feedback:**  That's correct! SetAll(0) sets all bits in the BitArray to false. |
|  | No such method is availible |
|  | SetBits() |

***Score 1.00 of 1***

7. If we want to compare two objects of the same class then \_\_\_\_\_\_\_\_\_ interface must be implemented by the class.

|  |  |
| --- | --- |
|  | IEnamurable |
|  | IEnamurator |
|  | ICompare |
| Correct | IComparar |
| **Feedback:**  That's correct! Compare() method of IComparar must be implemented by the class to compare two objects. |

***Score 1.00 of 1***

8. Dictionary class implements \_\_\_\_\_\_\_\_ interface.

|  |  |
| --- | --- |
|  | IGenerics |
|  | ICollections |
| Correct | IDictionary |
| **Feedback:**  That's correct! Dictionary class implements IDictionary interface. |
|  | IKeyValue |