1. Will the following code compile?

double d = 9999.11;

int i = d;

**Ans: Compiler will show error since the type does not match.**

1. For the following C# code what is the relevant solution for conversion of data type.

static void Main(string[] args)

{

int num1 = 20000;

int num2 = 50000;

long total;

total = num1 + num2;

Console.WriteLine($"Total is : {total}");

Console.ReadLine();

}

**Ans : Here the output will be shown since here the explicit conversion takes place because int is a subset of long hence the compiler will automatically convert the datatype and will give the output.**

1. For the given set of C# code, is conversion possible?

static void Main(string[] args)

{

int a = 76;

char b;

b = (char)a;

Console.WriteLine(b);

Console.ReadLine();

}

**Ans: Yes , It is an explicit conversion in C# since a cast is used where an int value is casted into char type using cast option.**

1. What will be the output of the following C# code?

static void Main(string[] args)

{

float sum;

int i;

sum = 0.0F;

for (i = 1; i <= 10; i++)

{

sum = sum + 1 /(float)i;

}

Console.WriteLine($"sum = {sum}");

Console.ReadLine();

}

**Ans: sum = 2.9289684, float value will be displayed.**

1. Refactor the following code (apply explicit conversion) to get an output without any error?

static void Main(string[] args)

{

int a = 22;

long b = 44;

double c = 1.406;

b = a;

c = a;

a = b;

b = c;

}

**Ans : Here we can use explicit type conversion using cast operation.**

**using System;**

**public class MyClass**

**{**

**static void Main(string[] args)**

**{**

**int a = 22;**

**long b = 44;**

**double c = 1.406;**

**b = a;**

**c = a;**

**a = (int)b;**

**b = (long)c;**

**Console.WriteLine(a);**

**Console.WriteLine(b);**

**Console.WriteLine(c);**

**}**

**}**

1. Write a program to print the current day. Eg: Today is Tuesday.
2. Write a C# Sharp program that takes two numbers as input and perform an operation (+,-,\*,x,/,%) on them using switch statement and displays the result of that operation.
3. Write a program in C# Sharp to read 10 numbers from the keyboard and find their sum and average.

Test Data :

Input the 10 numbers :

Number-1 :2

...

Number-10 :2

Expected Output :

The sum of 10 no is : 51

The Average is : 5.100000

1. Write a C# Sharp program that takes a number and a width also a number, as input and then displays a triangle of that width, using that number.

Test Data

Enter a number: 6

Enter the desired width: 6

Expected Output :

666666

66666

6666

666

66

6

1. Write a program in C# Sharp to sort a string array in ascending order. Go to the editor

Test Data :

Input the string : this is a string

Expected Output :

1. Find how many prime numbers lying between the given range using loops and conditional statements
2. Write a function to parse the DateTime string and returns true if it is successfully parsed else return false using TryParse method
3. What will be the correct order of execution of function func1(), func2() & func3() in the given code snippet?

class baseclass

{

public void func1() {}

public virtual void func2() {}

public virtual void func3() {}

}

class derivedclass :baseclass

{

new public void func1() {}

public override void func2() {}

public new void func3() {}

}

class Program

{

static void Main(string[] args)

{

baseclass b = new derivedclass();

b.func1 ();

b.func2 ();

b.func3 ();

}

}

**Ans: func1() of base class get executed**

**func2() of derived class get executed as override method is used.**

**func3() of base class get executed**

1. What will be the output of the following code snippet?

class BaseClass

{

public virtual void Print()

{

System.Console.WriteLine("BaseClass");

}

}

class DerivedClass : BaseClass

{

public override void Print()

{

System.Console.WriteLine("DerivedClass");

}

}

class Derived2Class : BaseClass

{

public void Print()

{

System.Console.WriteLine("Derived2Class");

}

}

class Program

{

public static void Main()

{

BaseClass b;

b = new DerivedClass();

b.Print();

BaseClass c= new Derived2Class();

c.Print();

}

}

**Ans: derivedclass**

**baseclass**

1. If we call the below statements in main (), what is the wings’ colour of Parrot? Give reason.

Bird b = new Parrot();

b.wings();

In case of unknown colour, complete the source code of classes to get correct colour of Parrot’s wings.

class Bird

{

public void legs() { Console.WriteLine("legs: 2"); }

public void wings() { Console.WriteLine("colour: unknown"); }

}

class Parrot : Bird

{

public void wings() { Console.WriteLine("colour: Green"); }

}

**Ans: Color:unknown**

**Because here in this class it uses method hiding characteristic of C# .**

1. What is output of the following program? When we create 2 objects of the class A e.g.

A obj1 = new A();

A obj2 = new A();

class A

{

static A()//Static constructor

{

Console.WriteLine("Static Constructor");

}

public A()// Constructor

{

Console.WriteLine("Class Constructor");

}

}

**Ans: static constructor**

**class constructor**

**class constructor**

**Here the static constructor will be executed before object creation and it will be executed only once.**

1. Write a program in C# Sharp to create a nested struct (Student and DateOfBirth) and store two records of students in an array.
2. Write complete program for below scenario in C#.

Create a class Travel Insurance using below Interface and call GetInsurance() method in main() program.

interface Insurance

{

void GetInsurance();

}

1. What type of implementation is shown below:

interface IFlyable

{

void fly();

}

interface IEatable

{

void eat();

}

class Bird : IFlyable, IEatable

{

public void fly()

{

Console.WriteLine("Bird flying");

}

public void eat()

{

Console.WriteLine("Bird eats");

}

}

class Program

{

static void Main(string[] args)

{

Bird b = new Bird();

b.eat();

b.fly();

}

}

**Ans : Here the multiple interface are implemented implicitly the class Bird implements both IFlyable, IEatable interfaces. It defines all the members of the IFlyable, IEatable interfaces with public access modifier.**

1. How to implement two interface with same method in C#?

Eg:

interface ILoanCustomer

{

void GetCostomerInfo();

}

interface IBankCustomer

{

void GetCostomerInfo();

}

Implement the above interfaces in a class.

1. Create a text file in your local Machine using StreamWriter
2. Read a text file from your local Machine using Stream Reader
3. What is the Use of Digital Separator

**Ans : The concept of digit separator is introduced in C# 7.0. With the help of digit separator, you can separate the large number into small parts which makes your code more readable. Underscore(\_) is used as a digit separator. When you use digit separator in your code they are simply ignored by the compiler, so the compiler does not print digit separators in the output**

1. What is the Output of the following code snippet {long x = 100000022200000202;

long z = 10000000020;

Console.WriteLine("X: {0}", x);

Console.WriteLine("Z: {0}", z);

// Using digit separators

long num1 = 1\_00\_10\_00\_00\_00;

var num2 = 0b\_010\_000\_000\_000\_000\_000\_000\_000\_000;

long num3 = 1\_00\_00\_00\_00\_00\_00;

var num4 = 0b\_1\_1000\_0000\_1000\_0000\_0011\_0000\_0000\_1000\_0001;

Console.WriteLine("Num1: {0}", num1);

Console.WriteLine("Num2: {0}", num2);

Console.WriteLine("Num3: {0}", num3);

Console.WriteLine("Num4: {0}", num4);

**Ans**

**X: 100000022200000202**

**Z: 10000000020**

**Num1: 10010000000**

**Num2: 33554432**

**Num3: 1000000000000**

**Num4: 103213629569**

1. What is the Output Console.WriteLine(default(Int32));

Console.WriteLine(default(Boolean));

Console.WriteLine(default(String));

**Ans:**

**Output Console.WriteLine(default(Int32)); null**

**Console.WriteLine(default(Boolean)); false**

**Console.WriteLine(default(String)); null**

1. Write a generic class with a function to swap any 2 variables of same datatype( eg. integer, string).
2. What is the output var My\_Tuple = Tuple.Create(13, "Geeks", 67, 89.90,

'g', 39939, "geek", Tuple.Create(12, 30, 40, 50));

// Accessing the element of Tuple

// Using Item property

// And accessing the elements of nested tuple

// Using Rest property

Console.WriteLine("Element of My\_Tuple: "+My\_Tuple.Item1);

Console.WriteLine("Element of My\_Tuple: "+My\_Tuple.Item2);

Console.WriteLine("Element of My\_Tuple: "+My\_Tuple.Item3);

Console.WriteLine("Element of My\_Tuple: "+My\_Tuple.Item4);

Console.WriteLine("Element of My\_Tuple: "+My\_Tuple.Item5);

Console.WriteLine("Element of My\_Tuple: "+My\_Tuple.Item6);

Console.WriteLine("Element of My\_Tuple: "+My\_Tuple.Item7);

Console.WriteLine("Element of Nested tuple: "+My\_Tuple.Rest);

Console.WriteLine("Element of Nested tuple: "+My\_Tuple.Rest.Item1.Item1);

Console.WriteLine("Element of Nested tuple: "+My\_Tuple.Rest.Item1.Item2);

Console.WriteLine("Element of Nested tuple: "+My\_Tuple.Rest.Item1.Item3);

Console.WriteLine("Element of Nested tuple: "+My\_Tuple.Rest.Item1.Item4);

**Ans:**

**Element of My\_Tuple: 13**

**Element of My\_Tuple: Geeks**

**Element of My\_Tuple: 67**

**Element of My\_Tuple: 89.9**

**Element of My\_Tuple: g**

**Element of My\_Tuple: 39939**

**Element of My\_Tuple: geek**

**Element of Nested tuple: ((12, 30, 40, 50))**

**Element of Nested tuple: 12**

**Element of Nested tuple: 30**

**Element of Nested tuple: 40**

**Element of Nested tuple: 50**

1. What is the Output Queue myQueue = new Queue();

// Inserting the elements into the Queue

myQueue.Enqueue("C#");

myQueue.Enqueue("PHP");

myQueue.Enqueue("Perl");

myQueue.Enqueue("Java");

myQueue.Enqueue("C");

// Displaying the count of elements

// contained in the Queue

Console.Write("Total number of elements present in the Queue are: ");

Console.WriteLine(myQueue.Count);

// Displaying the beginning element of Queue

Console.WriteLine("Beginning Item is: " + myQueue.Peek());

**Ans:**

**Total number of elements present in the Queue are: 5**

**Beginning Item is: C#**

1. Write an extension method to read the DOB and check whether the current month is your birthday month otherwise show how many months left for your birthday
2. Write a function to do the following:

1. should accepts an Action delegate (to pass a function) and count (number of execution of that function)

2. write code to execute the passed function to 'count' times and find the average execution time using Stopwatch

1. Write a program in C# Sharp to find the negative numbers from a list of numbers using LINQ Query.
2. Write a program in C# Sharp to find the number in the array and the square of each number which is more than 30 using LINQ query.
3. Can one process run multiple threads in C#? If yes, provide an example

**Ans : yes C# allows multithreading as it allows multiple threads with in a single process all the threads will execute in the same time hence it will give a faster response .**

**Eg:**

**using System;**

**using System.Threading;**

**public class example {**

**public static void method1()**

**{**

**/**

**for (int I = 0; I <= 10; I++) {**

**Console.WriteLine("Method1 is : {0}", I);**

**if (I == 5) {**

**Thread.Sleep(6000);**

**}**

**}**

**}**

**public static void method2()**

**{**

**for (int J = 0; J <= 10; J++) {**

**Console.WriteLine("Method2 is : {0}", J);**

**}**

**}**

**// Main Method**

**static public void Main()**

**{**

**// Creating and initializing threads**

**Thread thr1 = new Thread(method1);**

**Thread thr2 = new Thread(method2);**

**thr1.Start();**

**thr2.Start();**

**}**

**}**

1. What is the output of the following code?

public static void Main()

{

foreach (int num in Numbers())

{

Console.WriteLine(num.ToString());

}

}

static IEnumerable<int> Numbers()

{

yield return 6;

yield return 2 + 4;

yield return 15;

yield return 0;

yield return -9;

yield return 264;

}

**Ans :**

**6**

**6**

**15**

**0**

**-9**

**264**

1. Write an alternate code with Using statement

string manyLines = @"This is line one

This is line two

Here is line three

The second last line is line four

This is the final, fifth line.";

var reader = new StringReader(manyLines);

try

{

string item;

do

{

item = reader.ReadLine();

Console.WriteLine(item);

} while (item != null);

Console.ReadKey();

}

finally

{

reader?.Dispose();

}

**Ans :**

**class MyClass**

**{**

**public static void Main()**

**{**

**string manyLines = @"This is line one**

**This is line two**

**Here is line three**

**The second last line is line four**

**This is the final, fifth line.";**

**using (var reader = new StringReader(manyLines))**

**{**

**string item;**

**do**

**{**

**item = reader.ReadLine();**

**Console.WriteLine(item);**

**} while (item != null);**

**Console.ReadKey();**

**}**

**}**

**}**