

# RAJSHAHI MOHILA POLYTECHNIC INSTITUTE

## OBJECT ORIENTED PROGRAMMING(66841)

Name: Most. shahanur Sultana

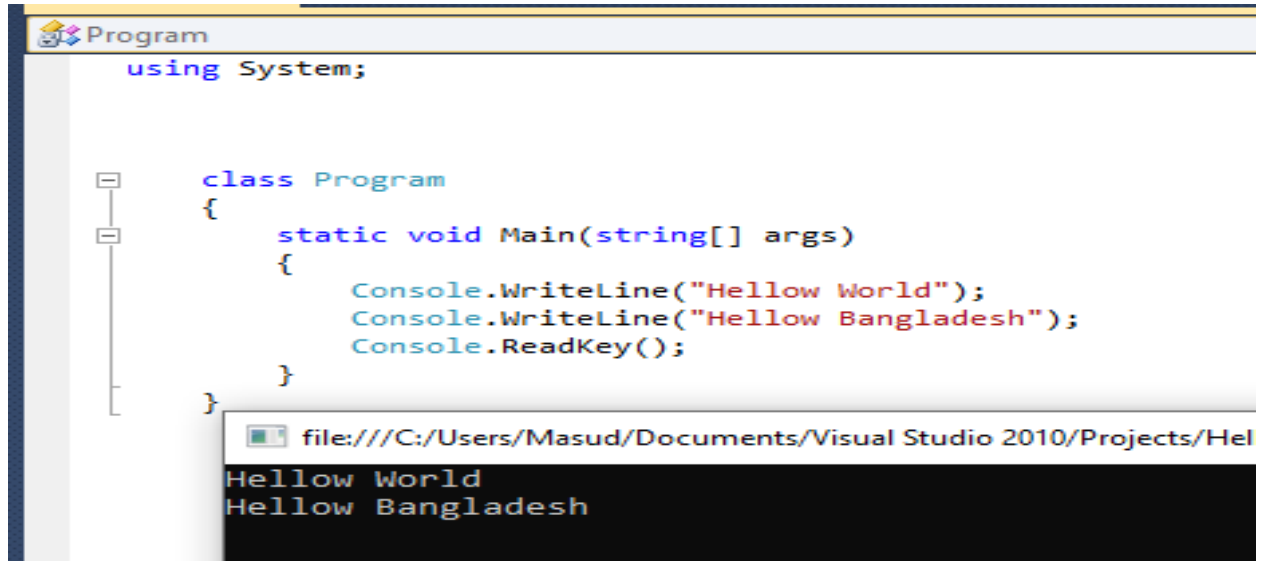
Roll: 420674

Semester: 4<sup>th</sup>

Shift: 2nd

Tech: CMT

## HELLO WORLD



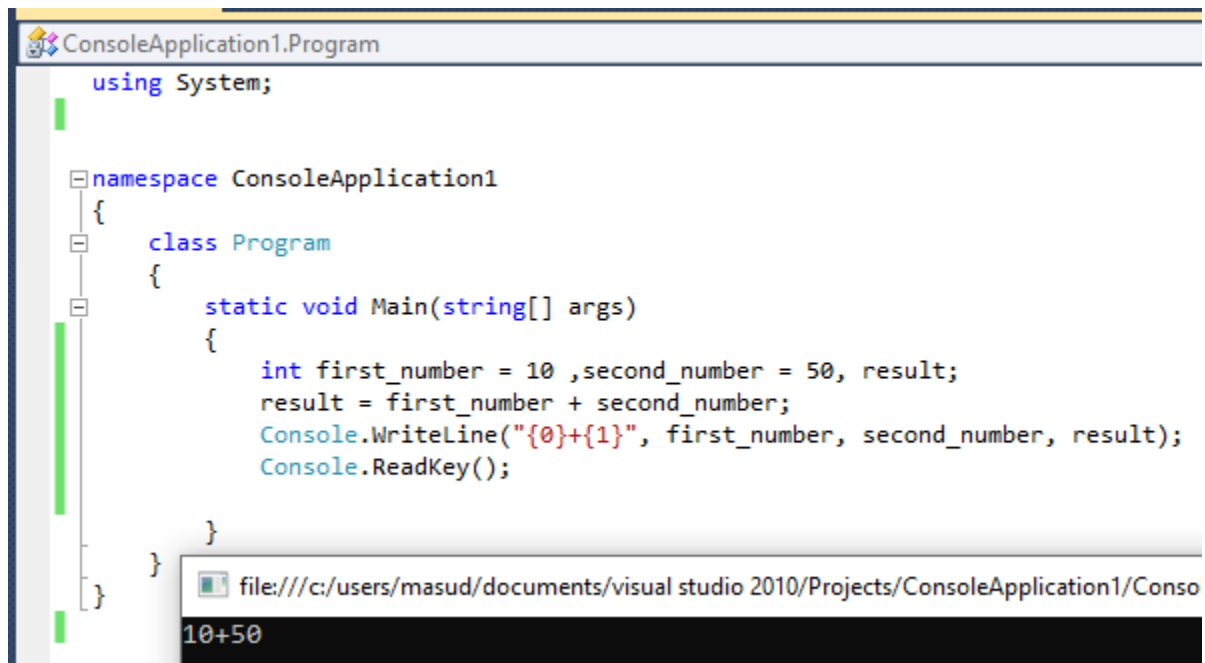
```
using System;

class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("Hello World");
        Console.WriteLine("Hello Bangladesh");
        Console.ReadKey();
    }
}
```

file:///C:/Users/Masud/Documents/Visual Studio 2010/Projects/Hel

Hello World  
Hello Bangladesh

## SUM OF TWO NUMBERS



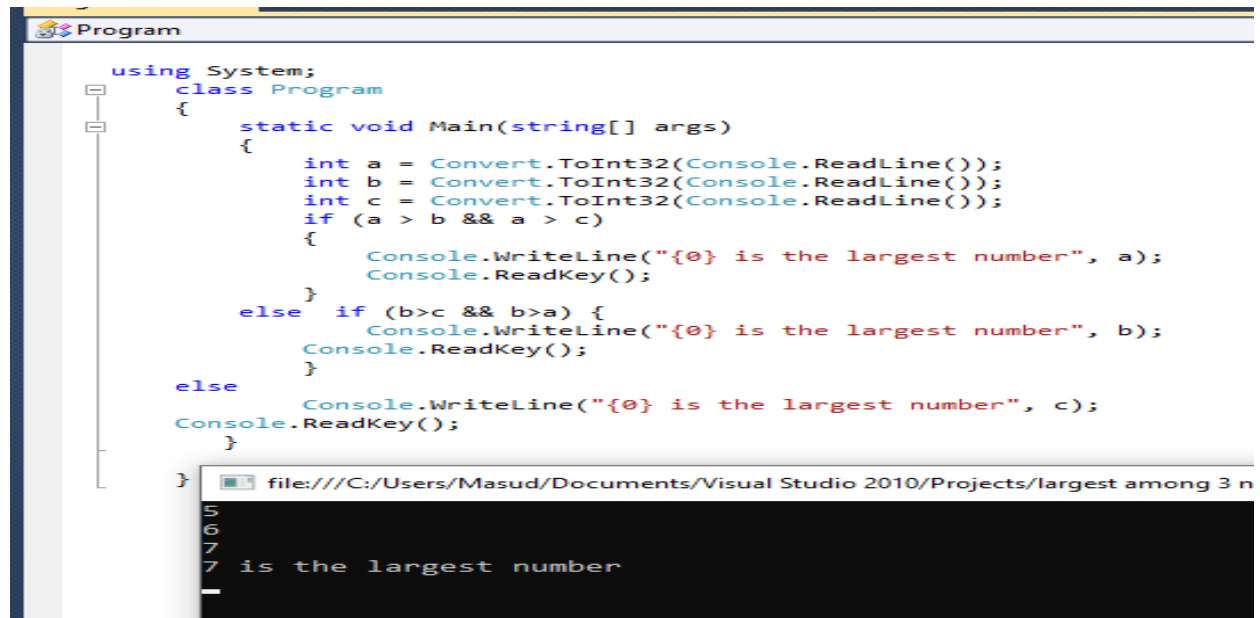
```
using System;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            int first_number = 10 ,second_number = 50, result;
            result = first_number + second_number;
            Console.WriteLine("{0}+{1}", first_number, second_number, result);
            Console.ReadKey();
        }
    }
}
```

file:///c:/users/masud/documents/visual studio 2010/Projects/ConsoleApplication1/Conso

10+50

## LARGE NUMBER AMONG THREE

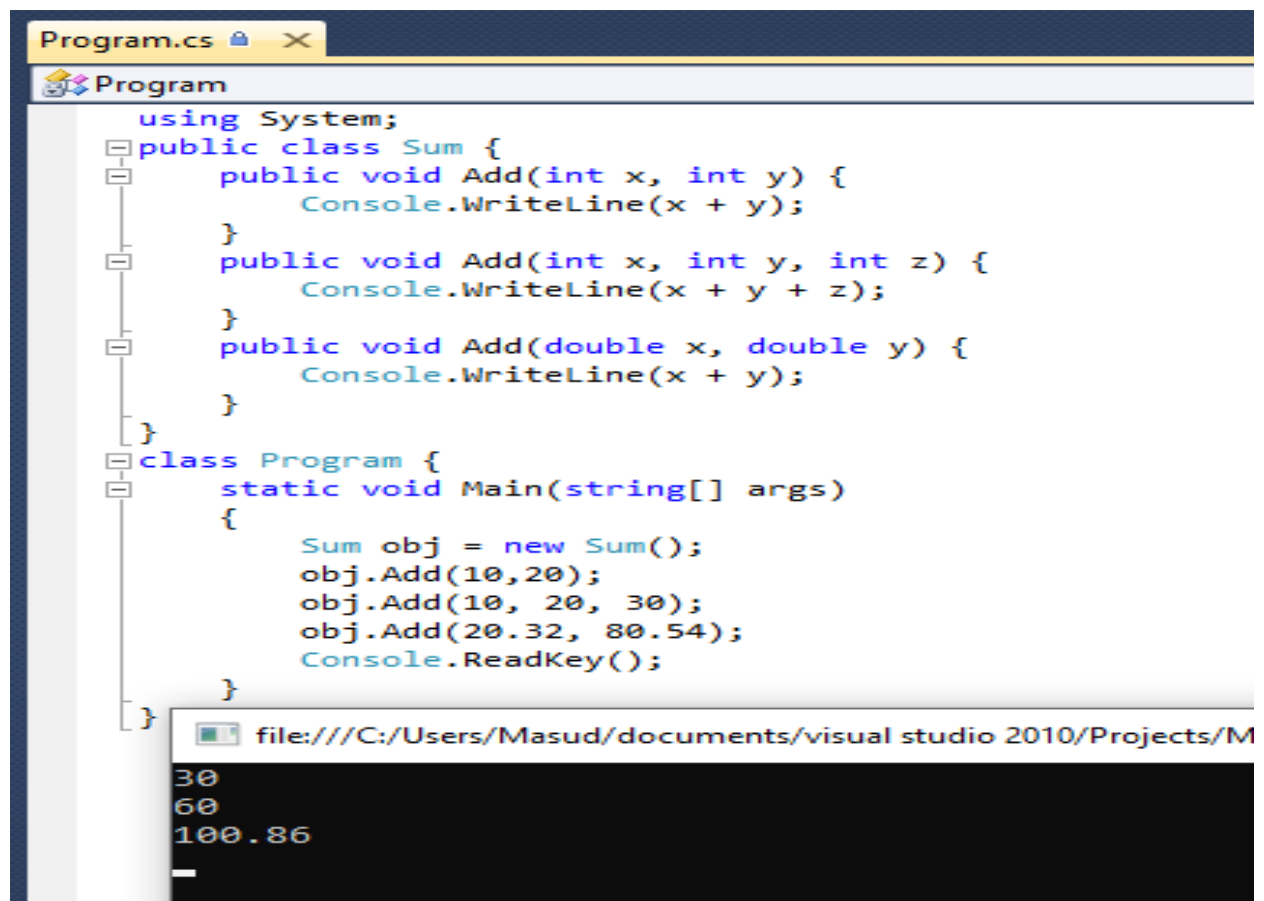


```
using System;
class Program
{
    static void Main(string[] args)
    {
        int a = Convert.ToInt32(Console.ReadLine());
        int b = Convert.ToInt32(Console.ReadLine());
        int c = Convert.ToInt32(Console.ReadLine());
        if (a > b && a > c)
        {
            Console.WriteLine("{0} is the largest number", a);
            Console.ReadKey();
        }
        else if (b > c && b > a) {
            Console.WriteLine("{0} is the largest number", b);
            Console.ReadKey();
        }
        else
            Console.WriteLine("{0} is the largest number", c);
        Console.ReadKey();
    }
}
```

file:///C:/Users/Masud/Documents/Visual Studio 2010/Projects/largest among 3 n

5  
6  
7  
7 is the largest number

## POLYMORPHISM

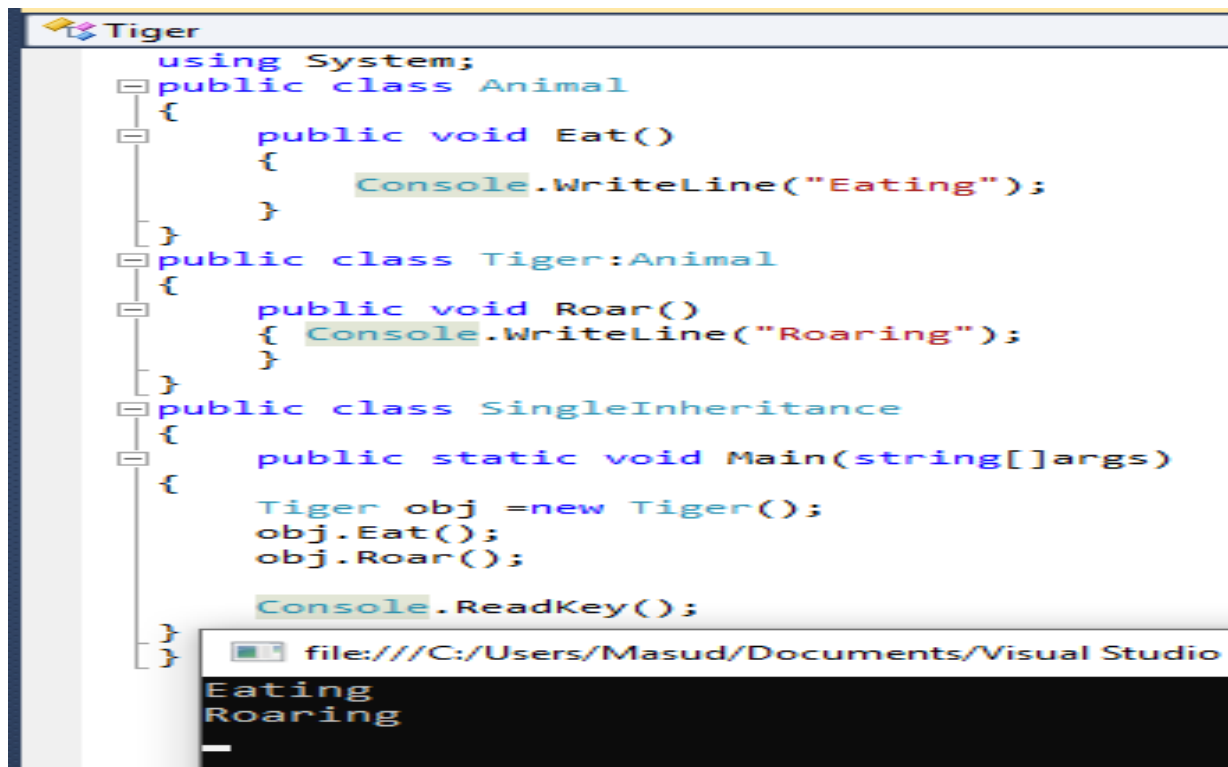


```
Program.cs
using System;
public class Sum {
    public void Add(int x, int y) {
        Console.WriteLine(x + y);
    }
    public void Add(int x, int y, int z) {
        Console.WriteLine(x + y + z);
    }
    public void Add(double x, double y) {
        Console.WriteLine(x + y);
    }
}
class Program {
    static void Main(string[] args)
    {
        Sum obj = new Sum();
        obj.Add(10, 20);
        obj.Add(10, 20, 30);
        obj.Add(20.32, 80.54);
        Console.ReadKey();
    }
}
```

file:///C:/Users/Masud/documents/visual studio 2010/Projects/M

30  
60  
100.86

## SINGLE INHERITANCE



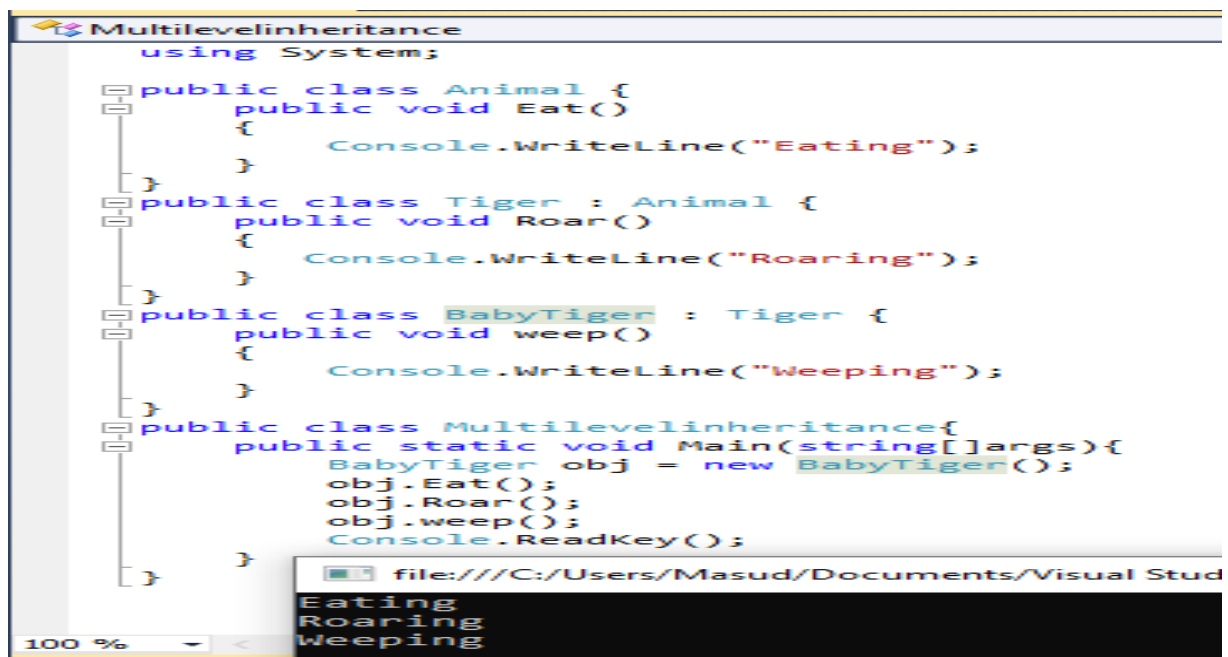
The screenshot shows a Visual Studio window titled "Tiger" containing C# code for single inheritance. The code defines an `Animal` class with an `Eat()` method, a `Tiger` class that inherits from `Animal` and adds a `Roar()` method, and a `SingleInheritance` class with a `Main` method that creates a `Tiger` object and calls its methods. The console output shows "Eating" and "Roaring".

```
using System;
public class Animal
{
    public void Eat()
    {
        Console.WriteLine("Eating");
    }
}
public class Tiger:Animal
{
    public void Roar()
    {
        Console.WriteLine("Roaring");
    }
}
public class SingleInheritance
{
    public static void Main(string[]args)
    {
        Tiger obj =new Tiger();
        obj.Eat();
        obj.Roar();

        Console.ReadKey();
    }
}
```

file:///C:/Users/Masud/Documents/Visual Studio  
Eating  
Roaring

## MILTILEVEL INHERIANCE

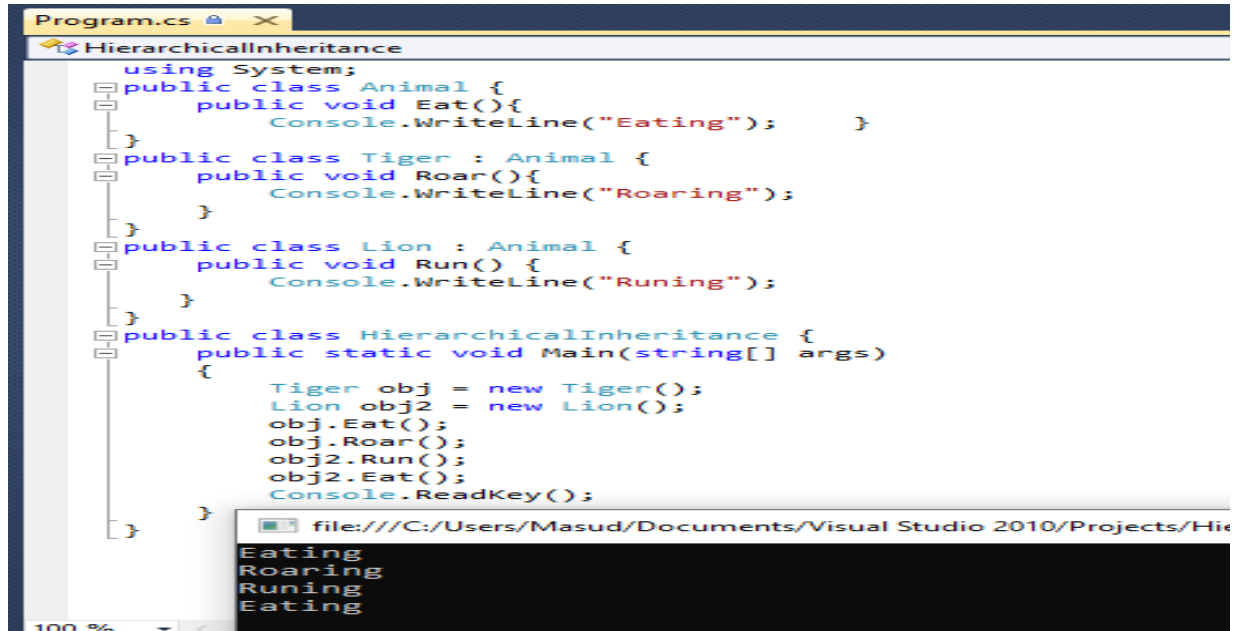


The screenshot shows a Visual Studio window titled "Multilevelinheritance" containing C# code for multilevel inheritance. The code defines an `Animal` class with an `Eat()` method, a `Tiger` class that inherits from `Animal` and adds a `Roar()` method, and a `BabyTiger` class that inherits from `Tiger` and adds a `weep()` method. The `Multilevelinheritance` class has a `Main` method that creates a `BabyTiger` object and calls its methods. The console output shows "Eating", "Roaring", and "Weeping".

```
using System;
public class Animal {
    public void Eat()
    {
        Console.WriteLine("Eating");
    }
}
public class Tiger : Animal {
    public void Roar()
    {
        Console.WriteLine("Roaring");
    }
}
public class BabyTiger : Tiger {
    public void weep()
    {
        Console.WriteLine("Weeping");
    }
}
public class Multilevelinheritance{
    public static void Main(string[]args){
        BabyTiger obj = new BabyTiger();
        obj.Eat();
        obj.Roar();
        obj.weep();
        Console.ReadKey();
    }
}
```

file:///C:/Users/Masud/Documents/Visual Stud  
Eating  
Roaring  
Weeping

## HIERARCHICAL

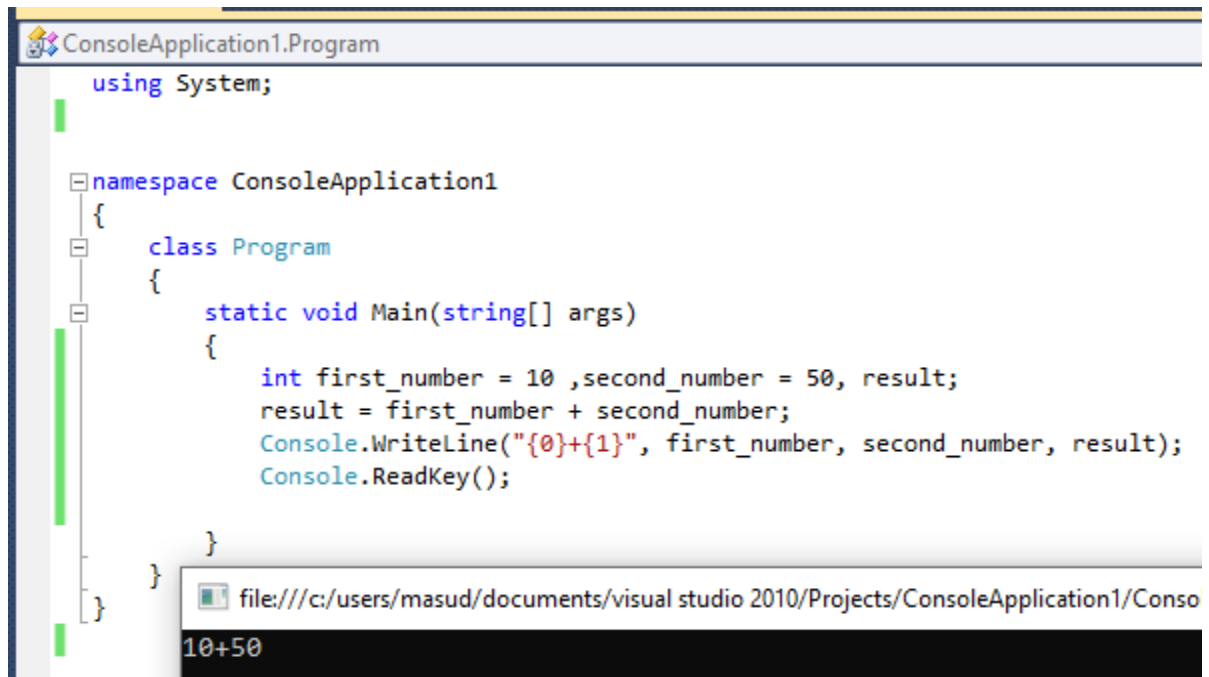


```
Program.cs
HierarchicalInheritance
using System;
public class Animal {
    public void Eat(){
        Console.WriteLine("Eating");    }
}
public class Tiger : Animal {
    public void Roar(){
        Console.WriteLine("Roaring");
    }
}
public class Lion : Animal {
    public void Run() {
        Console.WriteLine("Runing");
    }
}
public class HierarchicalInheritance {
    public static void Main(string[] args)
    {
        Tiger obj = new Tiger();
        Lion obj2 = new Lion();
        obj.Eat();
        obj.Roar();
        obj2.Run();
        obj2.Eat();
        Console.ReadKey();
    }
}
```

file:///C:/Users/Masud/Documents/Visual Studio 2010/Projects/Hie

Eating  
Roaring  
Runing  
Eating

## SUM OF TWO NUMBERS TAKING INPUT FROM KEYBOARD



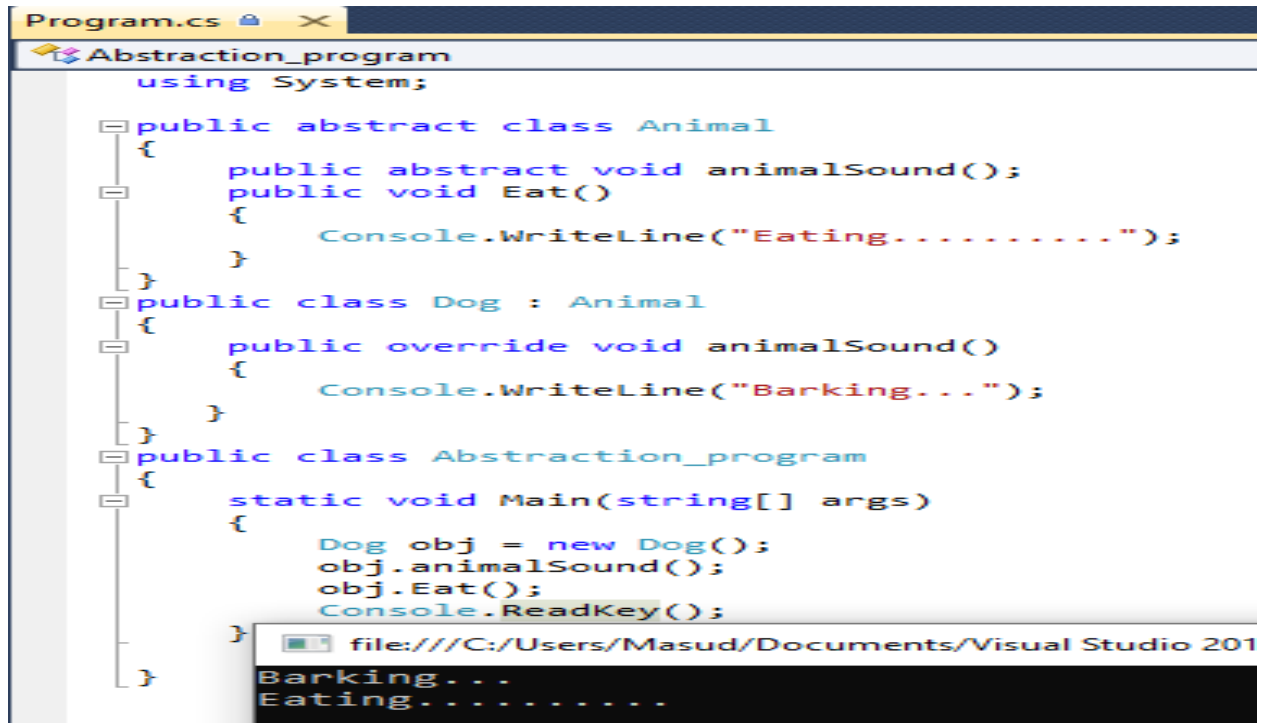
```
ConsoleApplication1.Program
using System;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            int first_number = 10 ,second_number = 50, result;
            result = first_number + second_number;
            Console.WriteLine("{0}+{1}", first_number, second_number, result);
            Console.ReadKey();
        }
    }
}
```

file:///c:/users/masud/documents/visual studio 2010/Projects/ConsoleApplication1/Conso

10+50

## DATA ABSTRACTION



```
Program.cs
Abstraction_program
using System;

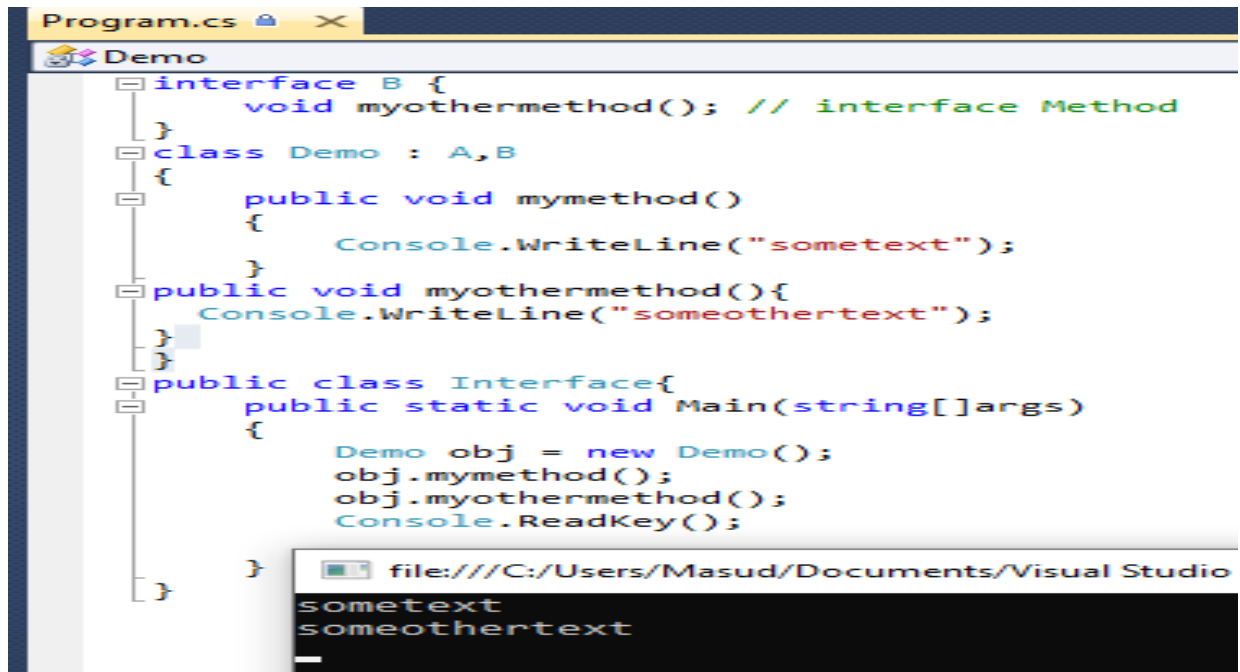
public abstract class Animal
{
    public abstract void animalSound();
    public void Eat()
    {
        Console.WriteLine("Eating.....");
    }
}

public class Dog : Animal
{
    public override void animalSound()
    {
        Console.WriteLine("Barking...");
    }
}

public class Abstraction_program
{
    static void Main(string[] args)
    {
        Dog obj = new Dog();
        obj.animalSound();
        obj.Eat();
        Console.ReadKey();
    }
}
```

file:///C:/Users/Masud/Documents/Visual Studio 201...  
Barking...  
Eating.....

## INTERFACE



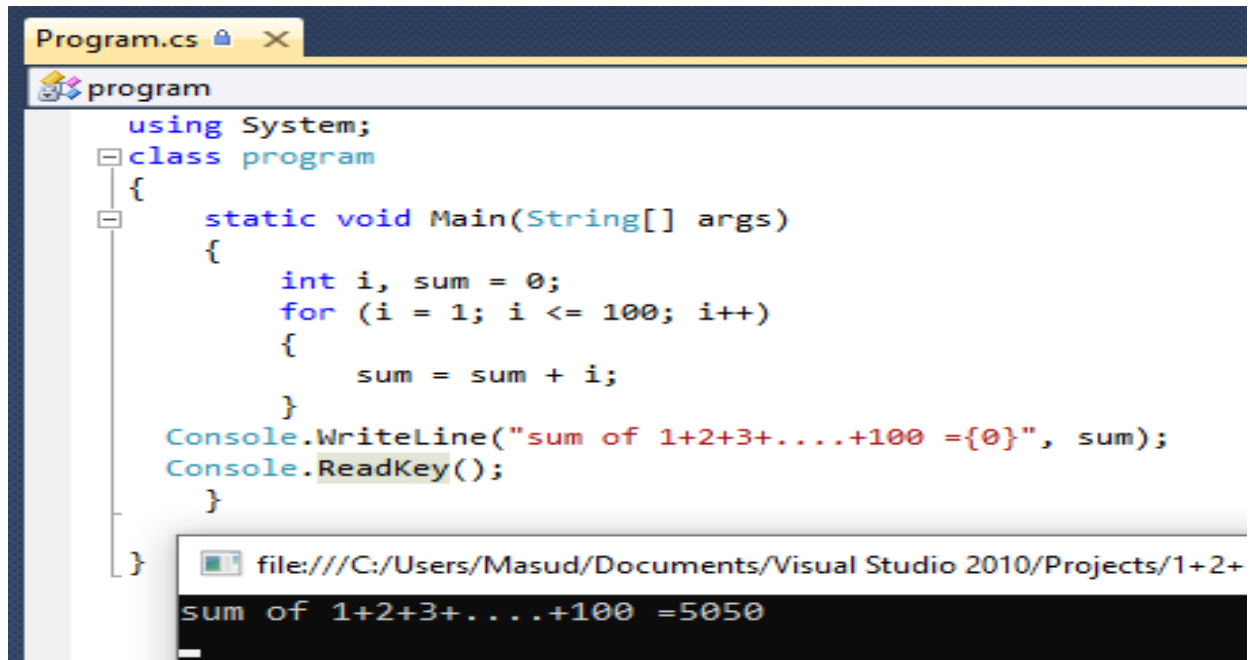
```
Program.cs
Demo
interface B {
    void myothermethod(); // interface Method
}

class Demo : A,B
{
    public void mymethod()
    {
        Console.WriteLine("sometext");
    }
    public void myothermethod(){
        Console.WriteLine("someothertext");
    }
}

public class Interface{
    public static void Main(string[]args)
    {
        Demo obj = new Demo();
        obj.mymethod();
        obj.myothermethod();
        Console.ReadKey();
    }
}
```

file:///C:/Users/Masud/Documents/Visual Studio...  
sometext  
someothertext

## SERIES SUM

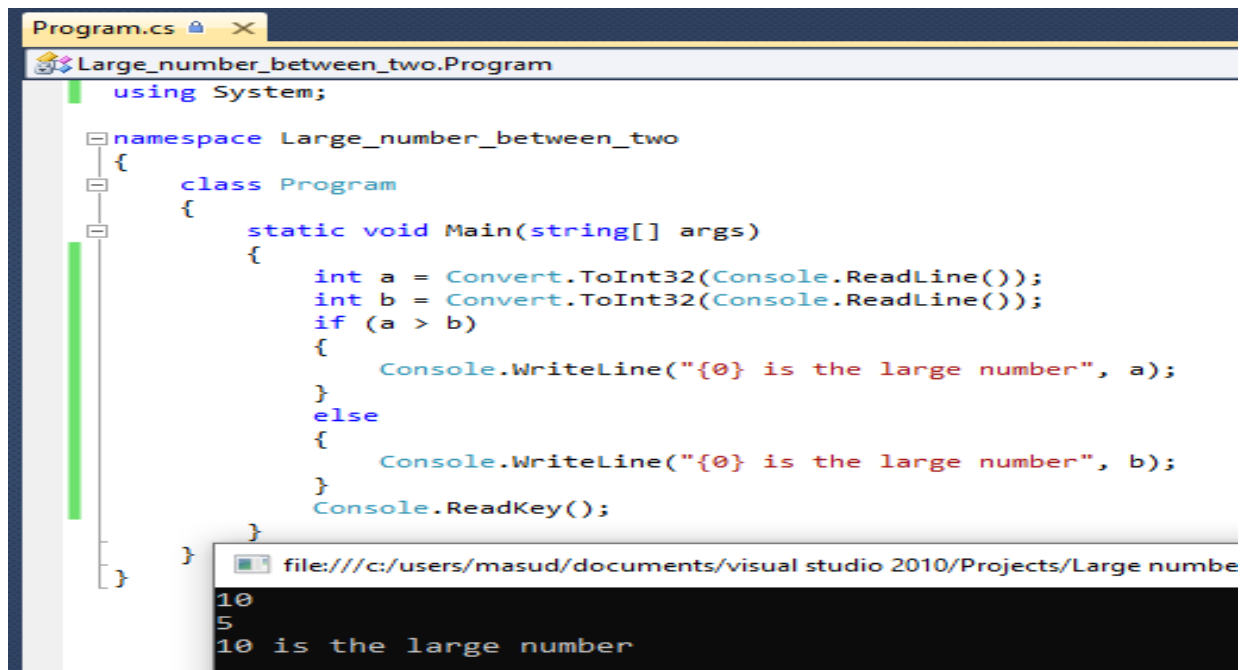


The screenshot shows a C# program in Visual Studio. The code calculates the sum of integers from 1 to 100. The output window displays the result: "sum of 1+2+3+....+100 =5050".

```
Program.cs
program
using System;
class program
{
    static void Main(String[] args)
    {
        int i, sum = 0;
        for (i = 1; i <= 100; i++)
        {
            sum = sum + i;
        }
        Console.WriteLine("sum of 1+2+3+....+100 ={0}", sum);
        Console.ReadKey();
    }
}
```

file:///C:/Users/Masud/Documents/Visual Studio 2010/Projects/1+2+3+...+100  
sum of 1+2+3+....+100 =5050

## LARGE NUMBER BETWEEN TWO

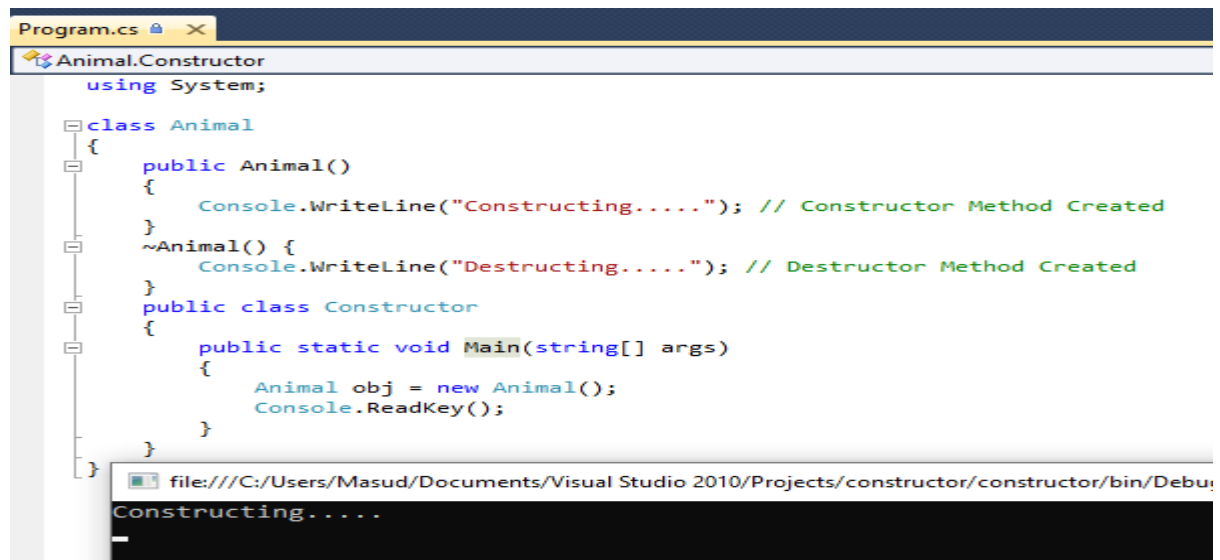


The screenshot shows a C# program in Visual Studio. The code prompts the user to enter two numbers, compares them, and prints the larger one. The output window shows the input "10" and "5", and the result "10 is the large number".

```
Program.cs
Large_number_between_two.Program
using System;
namespace Large_number_between_two
{
    class Program
    {
        static void Main(string[] args)
        {
            int a = Convert.ToInt32(Console.ReadLine());
            int b = Convert.ToInt32(Console.ReadLine());
            if (a > b)
            {
                Console.WriteLine("{0} is the large number", a);
            }
            else
            {
                Console.WriteLine("{0} is the large number", b);
            }
            Console.ReadKey();
        }
    }
}
```

file:///c:/users/masud/documents/visual studio 2010/Projects/Large number  
10  
5  
10 is the large number

## CONSTRUCTOR

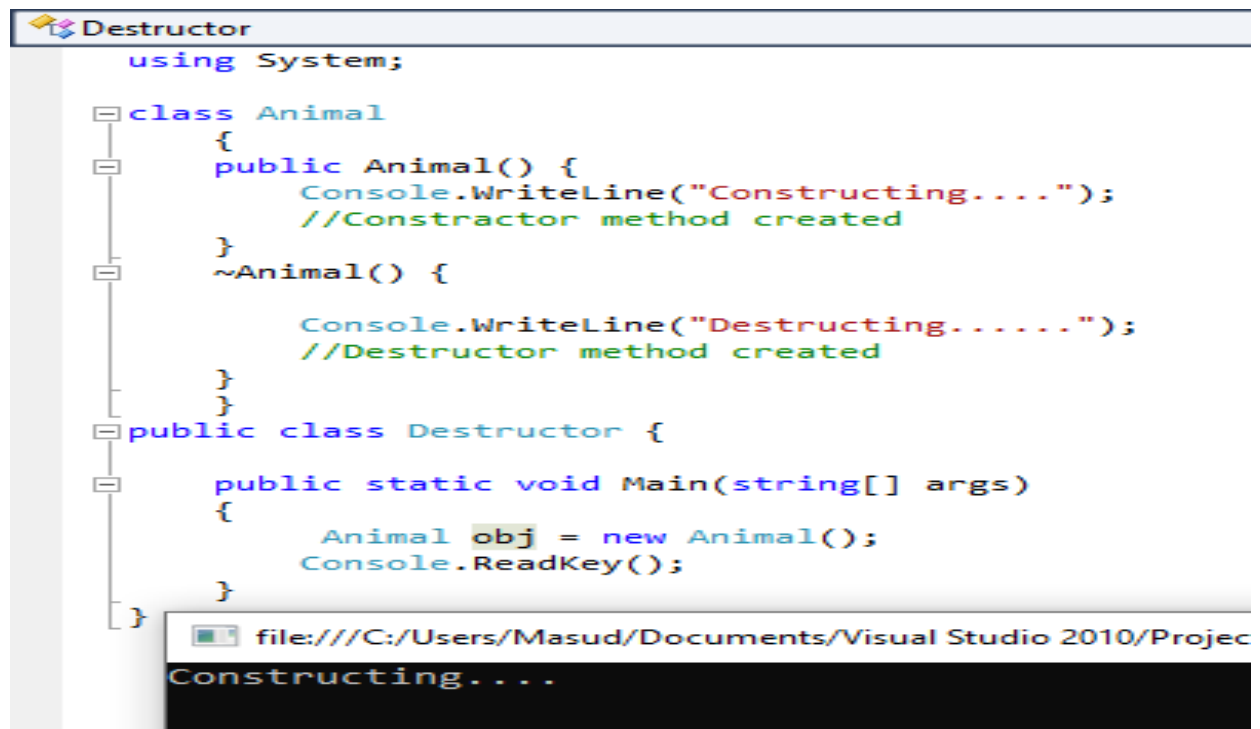


```
Program.cs
Animal.Constructor
using System;

class Animal
{
    public Animal()
    {
        Console.WriteLine("Constructing....."); // Constructor Method Created
    }
    ~Animal() {
        Console.WriteLine("Destructing....."); // Destructor Method Created
    }
    public class Constructor
    {
        public static void Main(string[] args)
        {
            Animal obj = new Animal();
            Console.ReadKey();
        }
    }
}
```

file:///C:/Users/Masud/Documents/Visual Studio 2010/Projects/constructor/constructor/bin/Debug/
Constructing.....

## DESTRUCTOR



```
Destructor
using System;

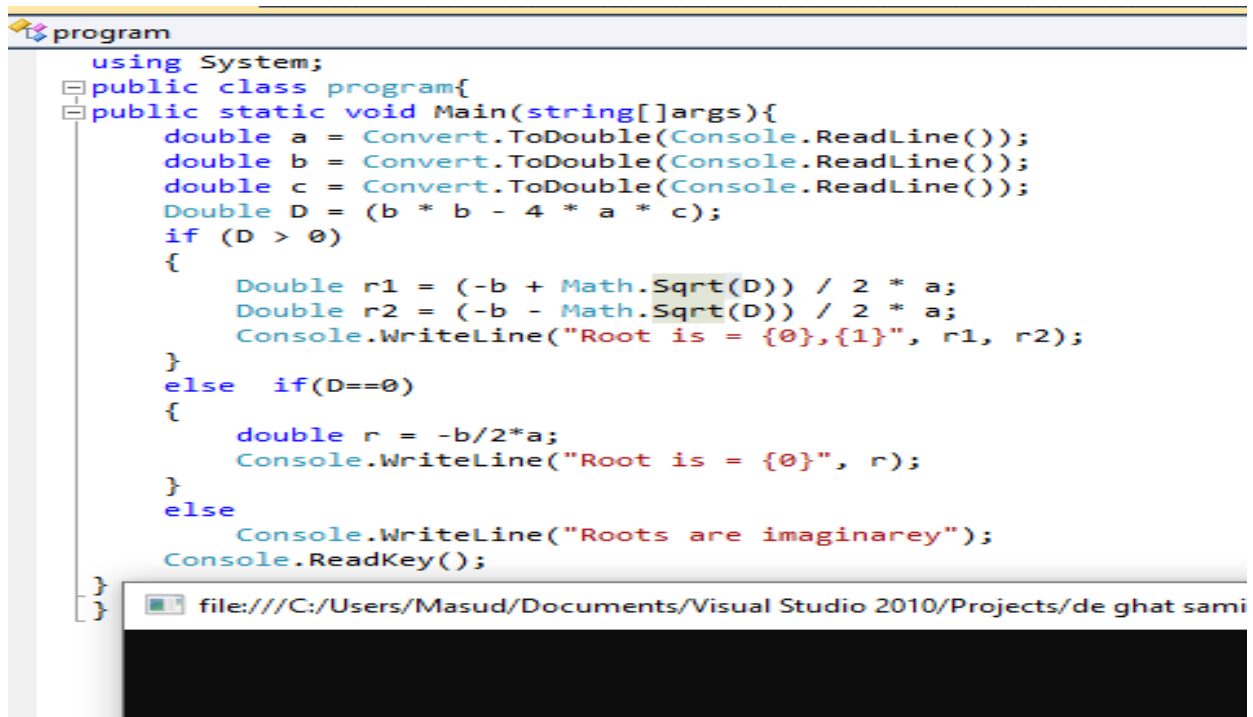
class Animal
{
    public Animal() {
        Console.WriteLine("Constructing....");
        //Constructor method created
    }
    ~Animal() {
        Console.WriteLine("Destructing.....");
        //Destructor method created
    }
}

public class Destructor {
    public static void Main(string[] args)
    {
        Animal obj = new Animal();
        Console.ReadKey();
    }
}
```

file:///C:/Users/Masud/Documents/Visual Studio 2010/Projec
Constructing.....



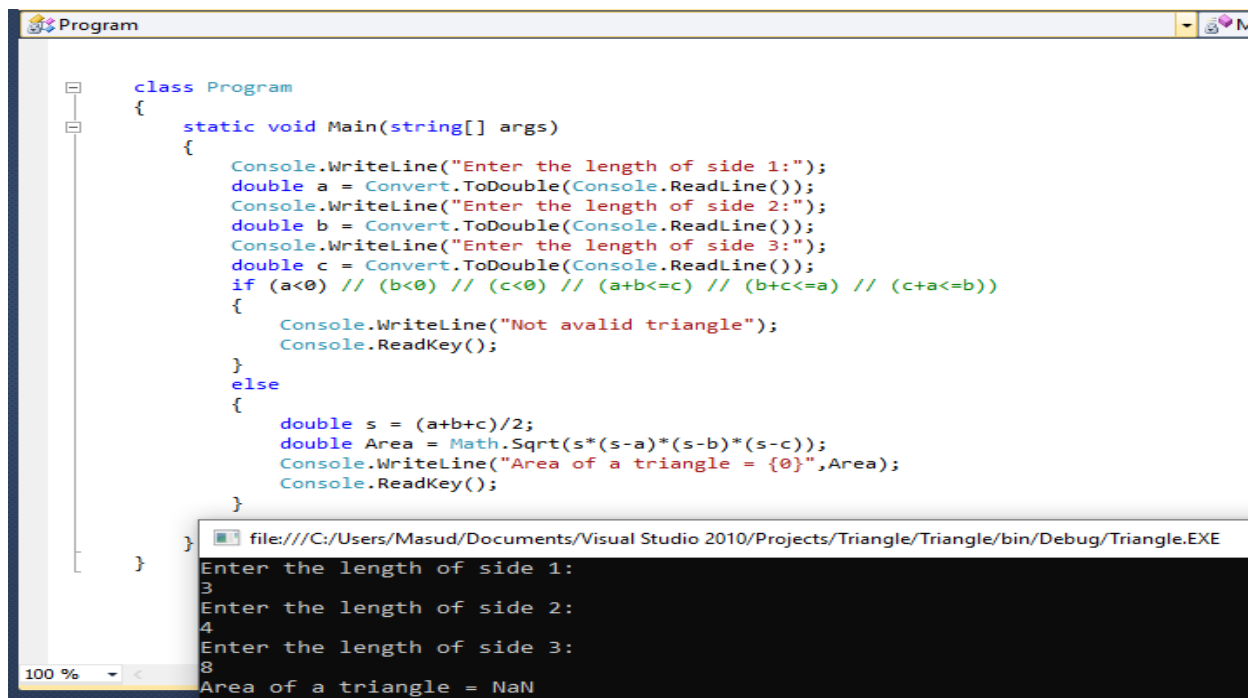
## DE GHAT SOMIKORON



```
using System;
public class program{
public static void Main(string[]args){
double a = Convert.ToDouble(Console.ReadLine());
double b = Convert.ToDouble(Console.ReadLine());
double c = Convert.ToDouble(Console.ReadLine());
Double D = (b * b - 4 * a * c);
if (D > 0)
{
Double r1 = (-b + Math.Sqrt(D)) / 2 * a;
Double r2 = (-b - Math.Sqrt(D)) / 2 * a;
Console.WriteLine("Root is = {0},{1}", r1, r2);
}
else if(D==0)
{
double r = -b/2*a;
Console.WriteLine("Root is = {0}", r);
}
else
Console.WriteLine("Roots are imaginarey");
Console.ReadKey();
}
}
```

file:///C:/Users/Masud/Documents/Visual Studio 2010/Projects/de ghat sami

## AREA OF TRIANGLE



```
class Program
{
static void Main(string[] args)
{
Console.WriteLine("Enter the length of side 1:");
double a = Convert.ToDouble(Console.ReadLine());
Console.WriteLine("Enter the length of side 2:");
double b = Convert.ToDouble(Console.ReadLine());
Console.WriteLine("Enter the length of side 3:");
double c = Convert.ToDouble(Console.ReadLine());
if (a<0) // (b<0) // (c<0) // (a+b<=c) // (b+c<=a) // (c+a<=b))
{
Console.WriteLine("Not avalid triangle");
Console.ReadKey();
}
else
{
double s = (a+b+c)/2;
double Area = Math.Sqrt(s*(s-a)*(s-b)*(s-c));
Console.WriteLine("Area of a triangle = {0}",Area);
Console.ReadKey();
}
}
}
```

file:///C:/Users/Masud/Documents/Visual Studio 2010/Projects/Triangle/Triangle/bin/Debug/Triangle.EXE

Enter the length of side 1:  
3  
Enter the length of side 2:  
4  
Enter the length of side 3:  
8  
Area of a triangle = NaN