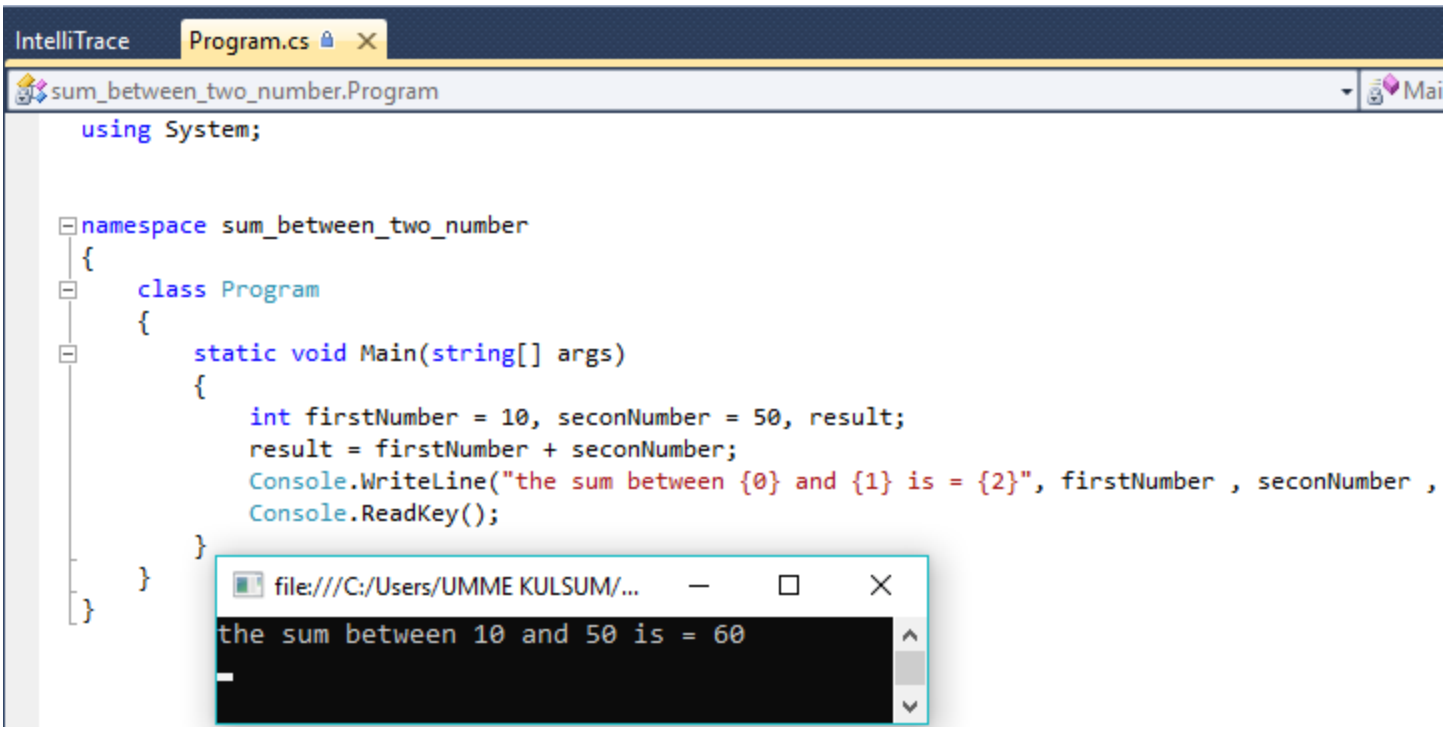
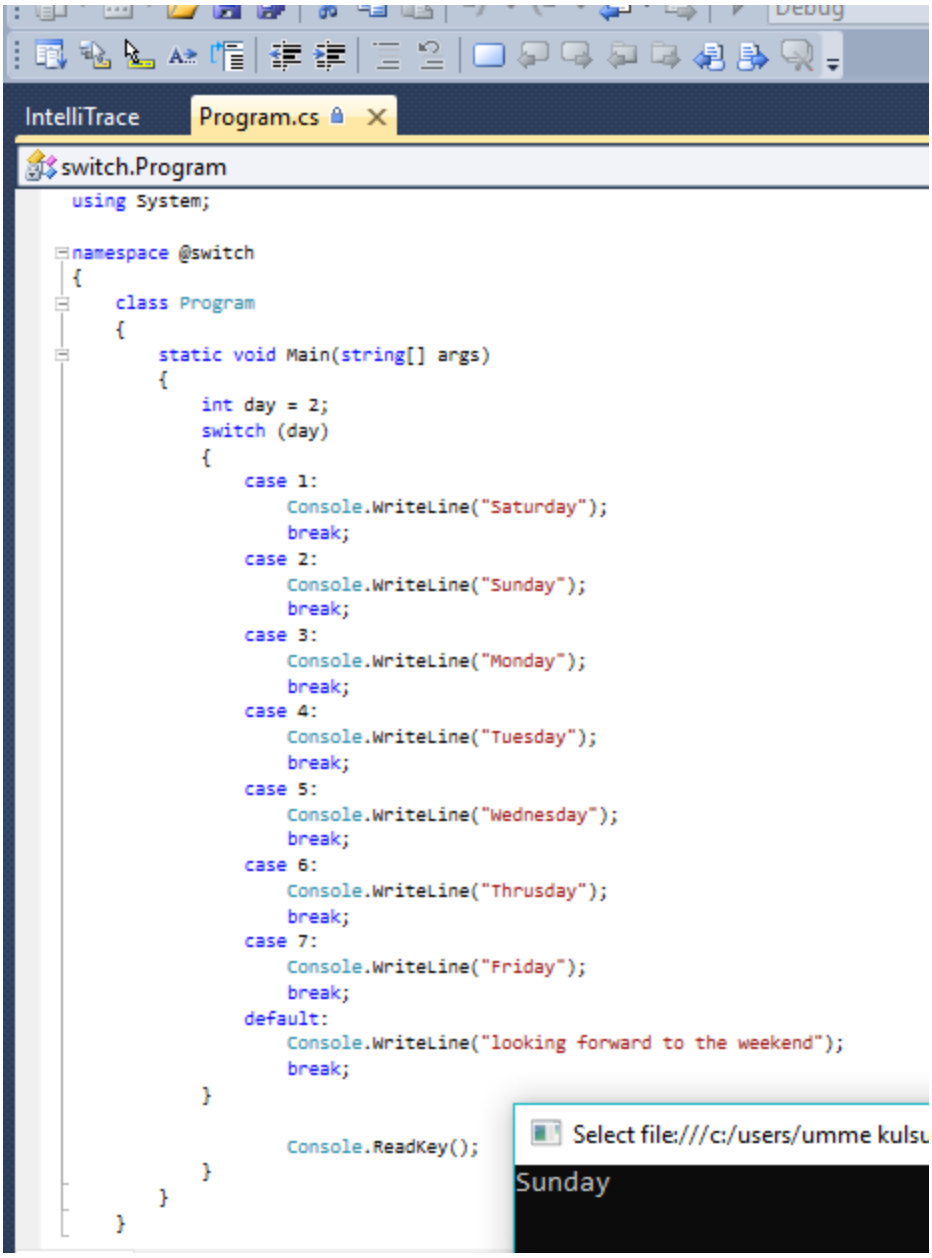
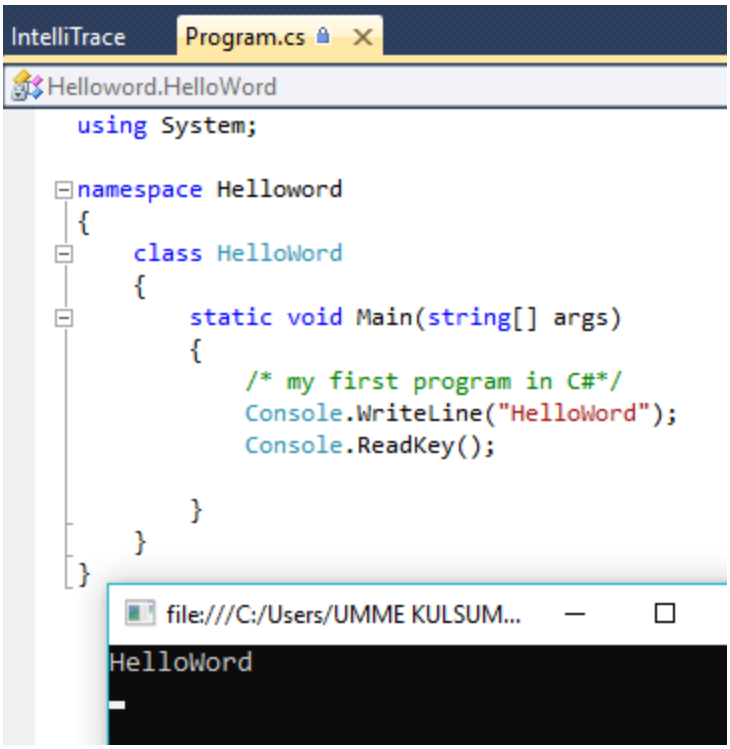
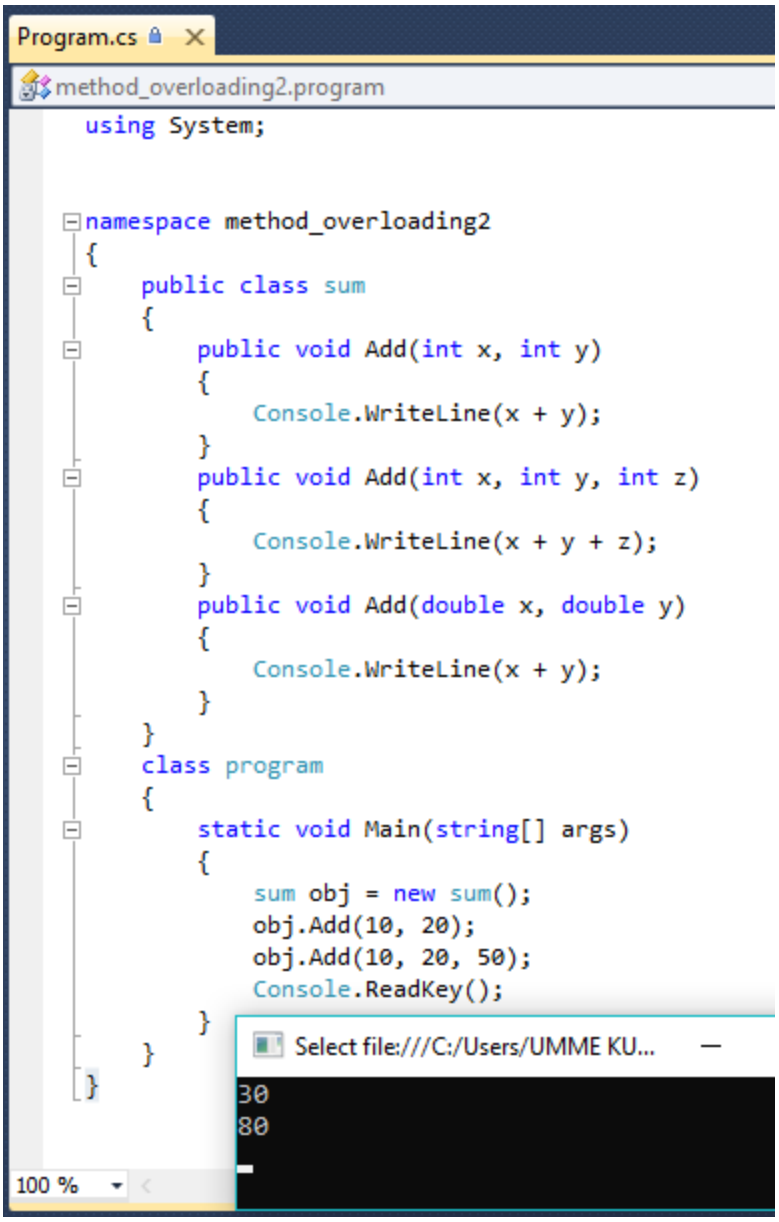
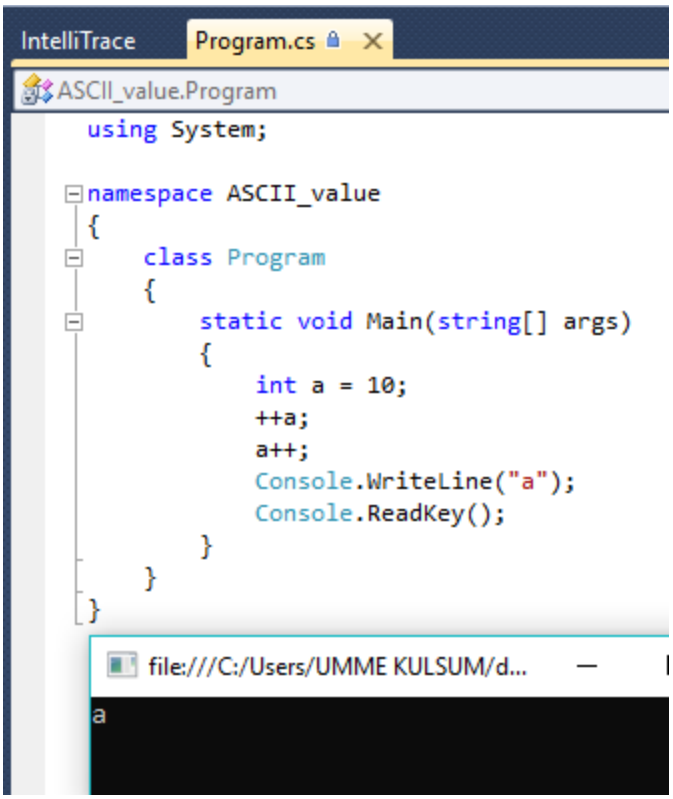
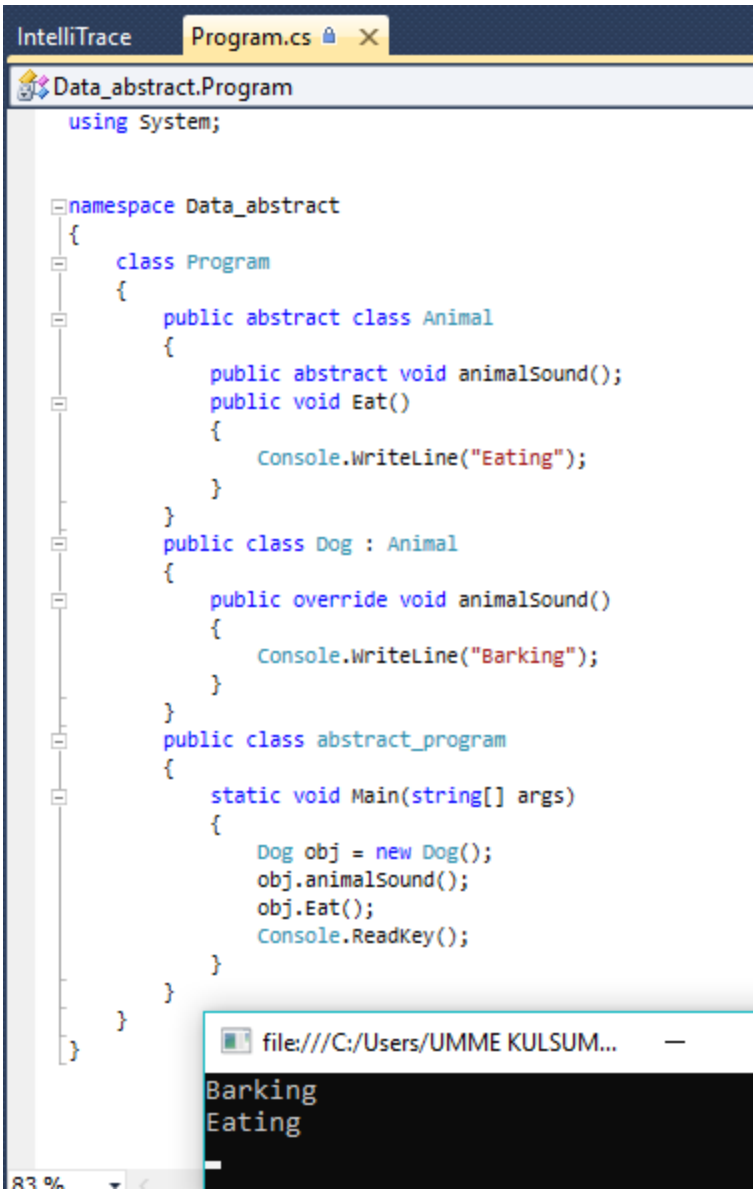

IntelliTrace
Program.cs
sum_between_two_number.Program
using System;
namespace sum_between_two_number
{
class Program
{
static void Main(string[] args)
{
int firstNumber = 10, seconNumber = 50, result;
result = firstNumber + seconNumber;
Console.WriteLine("the sum between {0} and {1} is = {2}", firstNumber , seconNumber ,
Console.ReadKey();
}
}
}
file:///C:/Users/UMME KULSUM/...
the sum between 10 and 50 is = 60

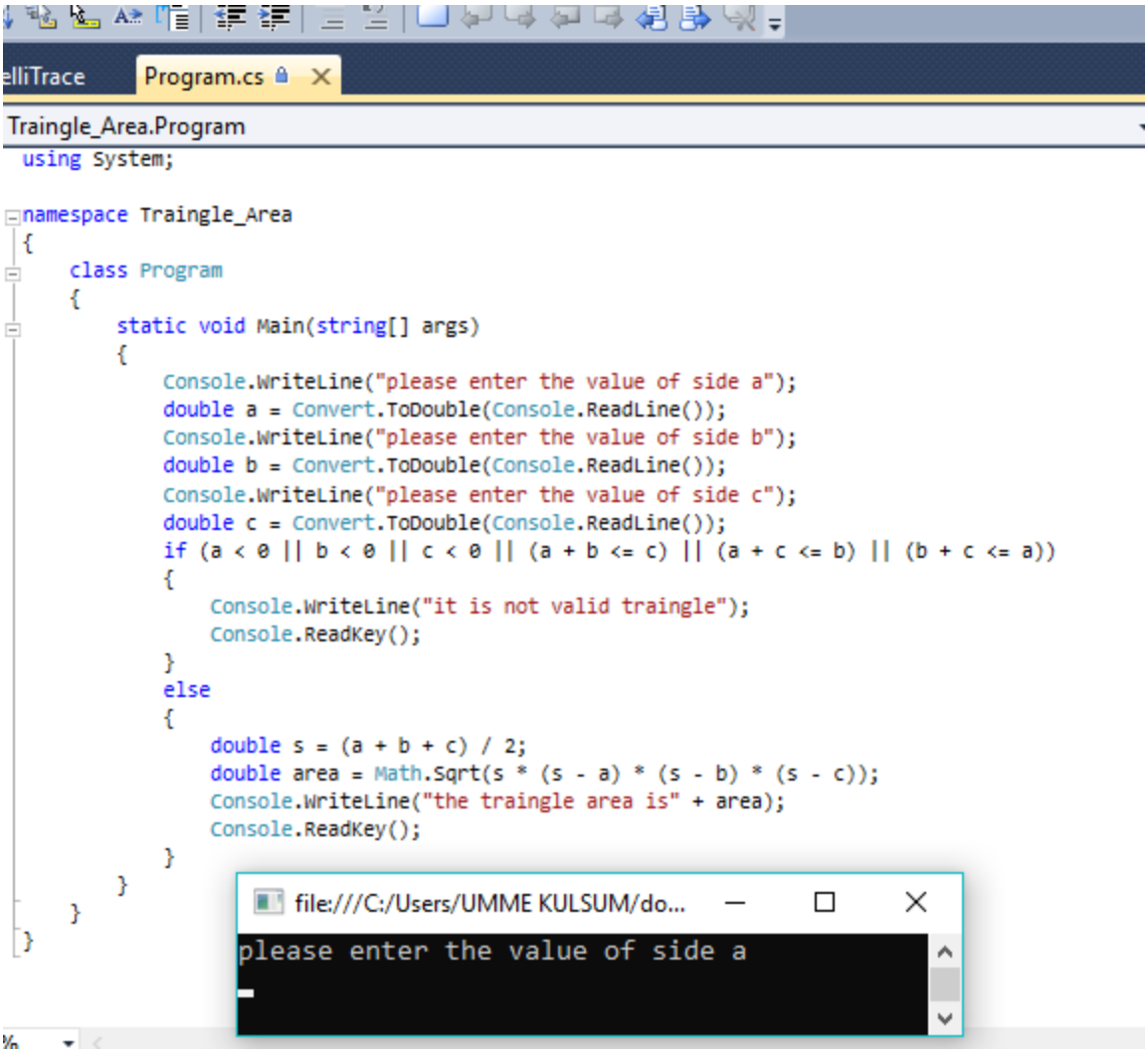

IntelliTrace
Program.cs
switch.Program
using System;
namespace @switch
{
class Program
{
static void Main(string[] args)
{
int day = 2;
switch (day)
{
case 1:
Console.WriteLine("Saturday");
break;
case 2:
Console.WriteLine("Sunday");
break;
case 3:
Console.WriteLine("Monday");
break;
case 4:
Console.WriteLine("Tuesday");
break;
case 5:
Console.WriteLine("Wednesday");
break;
case 6:
Console.WriteLine("Thrusday");
break;
case 7:
Console.WriteLine("Friday");
break;
default:
Console.WriteLine("looking forward to the weekend");
break;
}
Console.ReadKey();
}
}
}
Select file:///c:/users/umme kulsu
Sunday

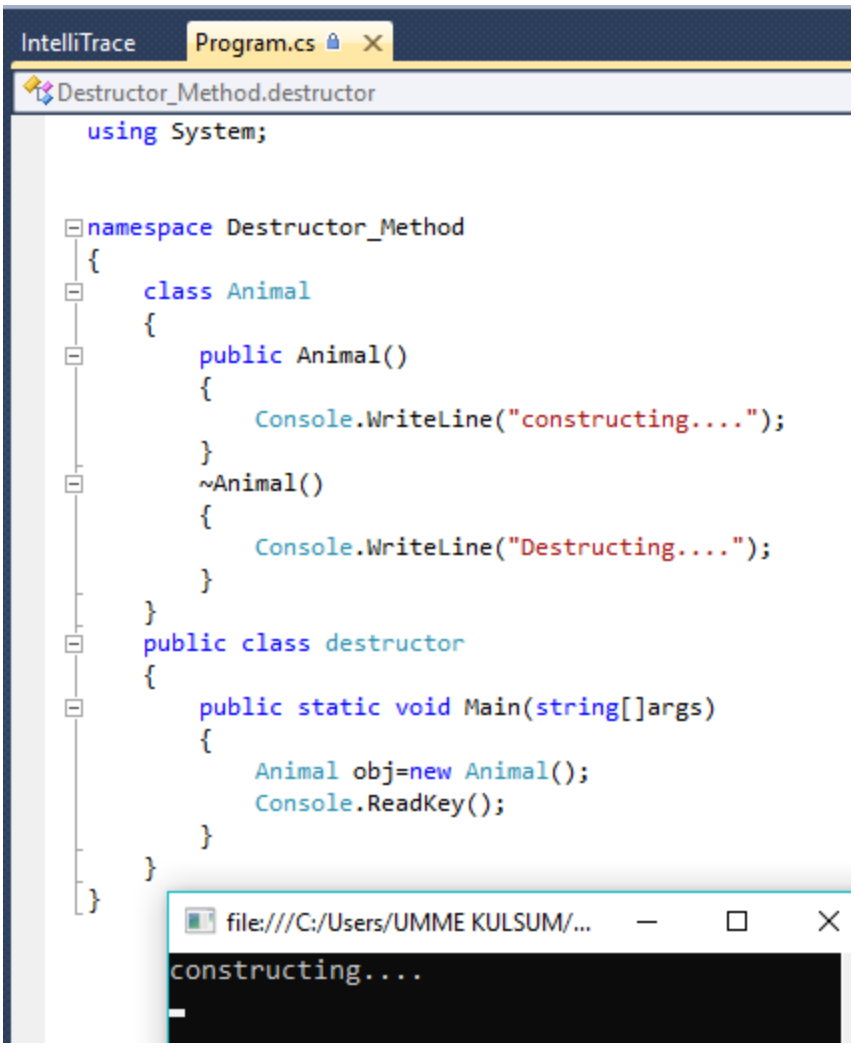

IntelliTrace
Program.cs
Helloword.HelloWord
using System;
namespace Helloword
{
class HelloWord
{
static void Main(string[] args)
{
/* my first program in C#*/
Console.WriteLine("HelloWord");
Console.ReadKey();
}
}
}
file:///C:/Users/UMME KULSUM...
HelloWord

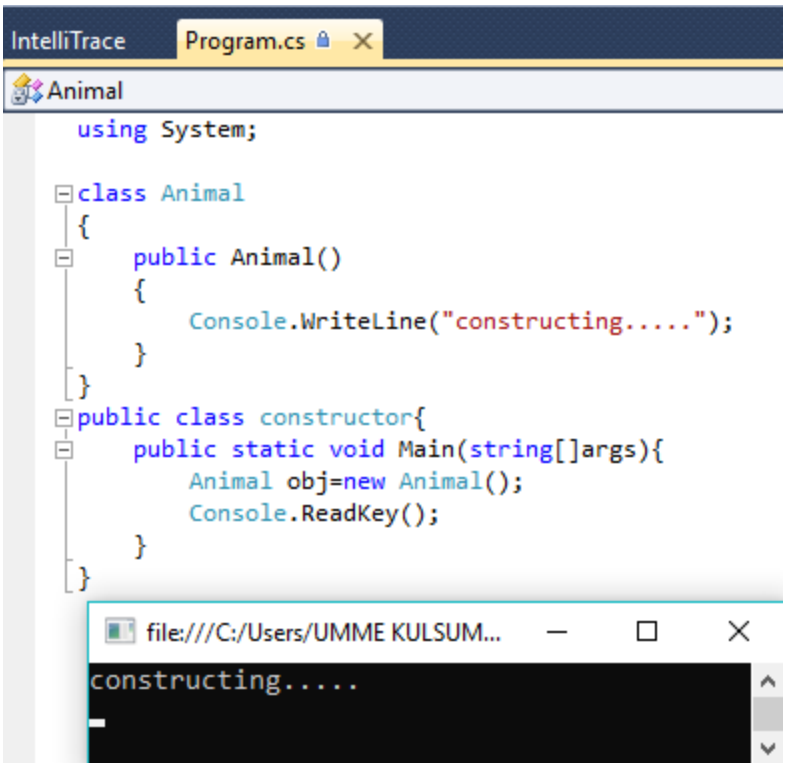

Program.cs
method_overloading2.program
using System;
namespace method_overloading2
{
public class sum
{
public void Add(int x, int y)
{
Console.WriteLine(x + y);
}
public void Add(int x, int y, int z)
{
Console.WriteLine(x + y + z);
}
public void Add(double x, double y)
{
Console.WriteLine(x + y);
}
}
class program
{
static void Main(string[] args)
{
sum obj = new sum();
obj.Add(10, 20);
obj.Add(10, 20, 50);
Console.ReadKey();
}
}
}
100 %
Select file:///C:/Users/UMME KU...
30
80

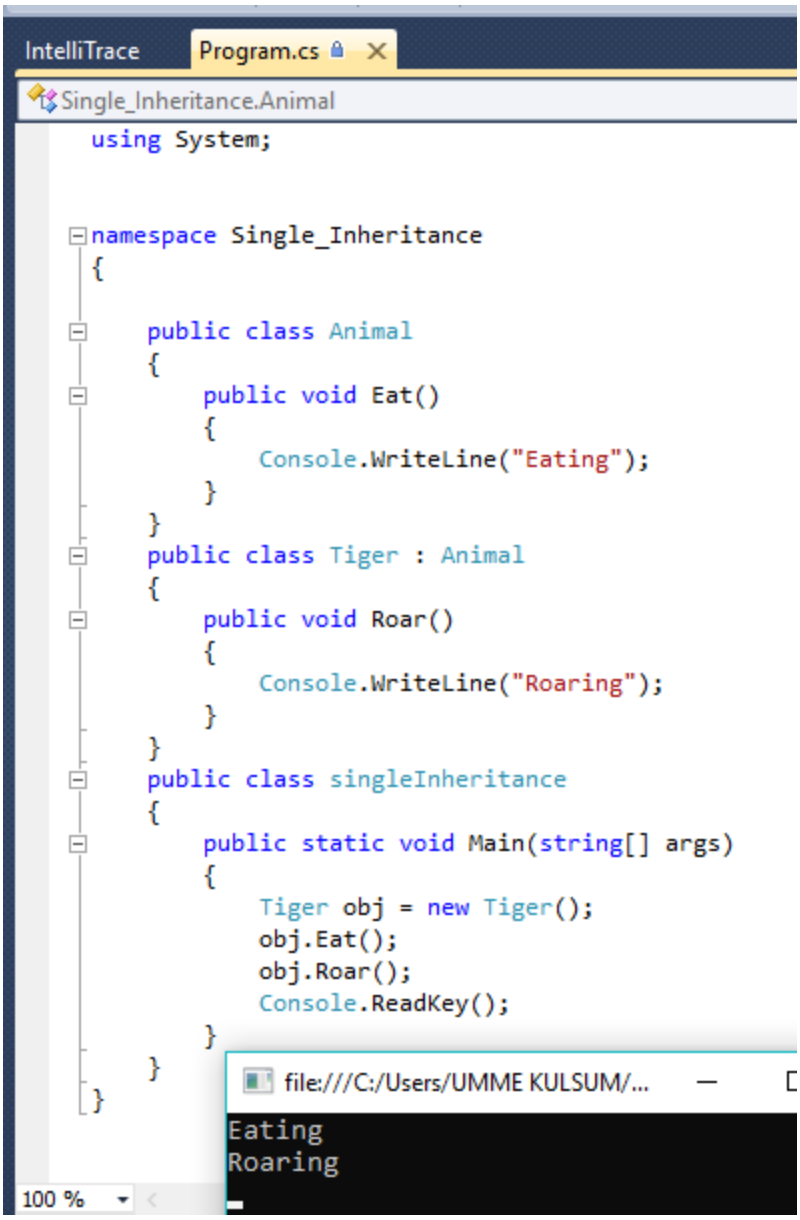

IntelliTrace
Program.cs
ASCII_value.Program
using System;
namespace ASCII_value
{
class Program
{
static void Main(string[] args)
{
int a = 10;
++a;
a++;
Console.WriteLine("a");
Console.ReadKey();
}
}
}
file:///C:/Users/UMME KULSUM/d...
a

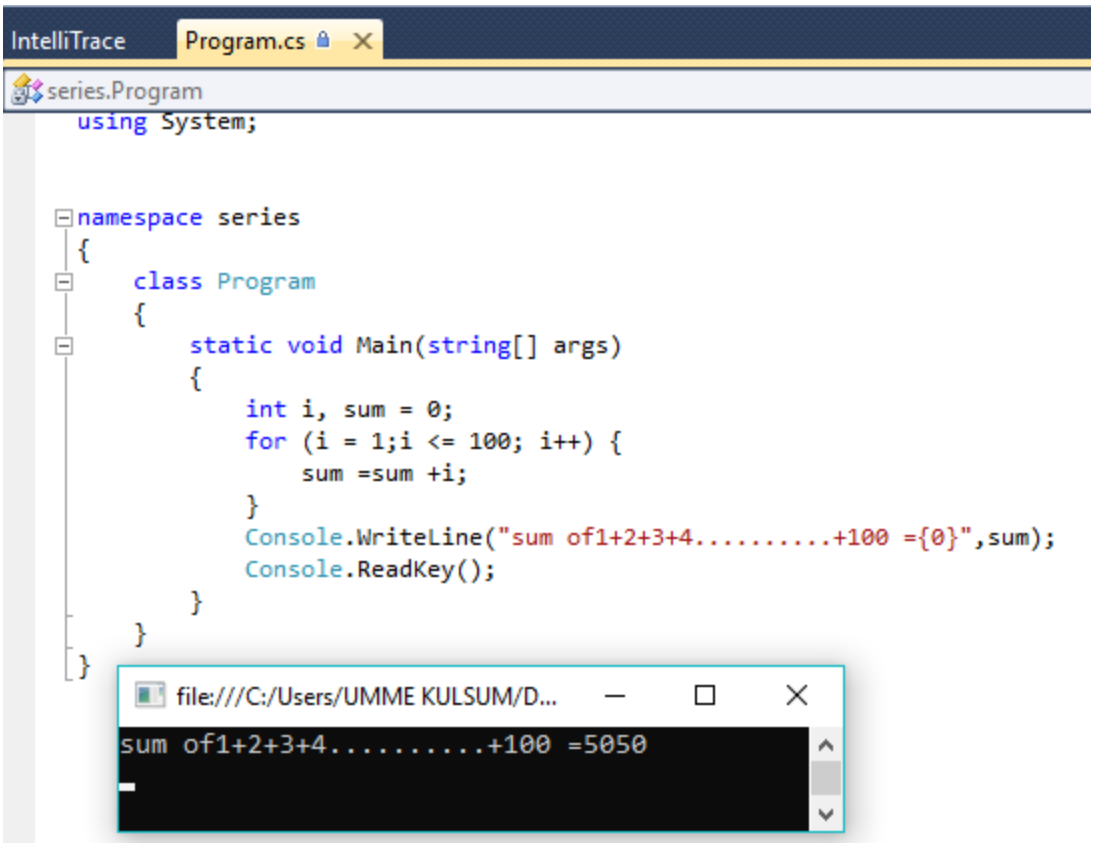

IntelliTrace
Program.cs
Data_abstract.Program
using System;
namespace Data_abstract
{
class Program
{
public abstract class Animal
{
public abstract void animalSound();
public void Eat()
{
Console.WriteLine("Eating");
}
}
public class Dog : Animal
{
public override void animalSound()
{
Console.WriteLine("Barking");
}
}
public class abstract_program
{
static void Main(string[] args)
{
Dog obj = new Dog();
obj.animalSound();
obj.Eat();
Console.ReadKey();
}
}
}
}
file:///C:/Users/UMME KULSUM...
Barking
Eating
83 %

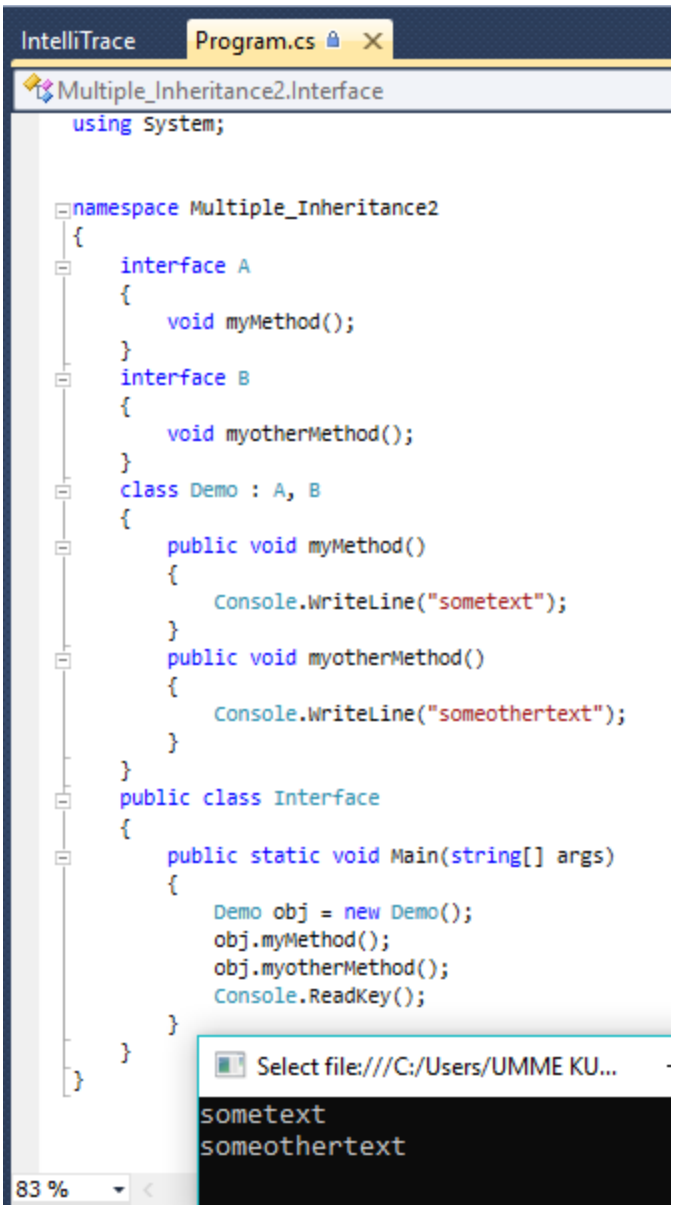

elliTrace
Program.cs
Traingle_Area.Program
using System;
namespace Traingle_Area
{
class Program
{
static void Main(string[] args)
{
Console.WriteLine("please enter the value of side a");
double a = Convert.ToDouble(Console.ReadLine());
Console.WriteLine("please enter the value of side b");
double b = Convert.ToDouble(Console.ReadLine());
Console.WriteLine("please enter the value of side c");
double c = Convert.ToDouble(Console.ReadLine());
if (a < 0 || b < 0 || c < 0 || (a + b <= c) || (a + c <= b) || (b + c <= a))
{
Console.WriteLine("it is not valid traingle");
Console.ReadKey();
}
else
{
double s = (a + b + c) / 2;
double area = Math.Sqrt(s * (s - a) * (s - b) * (s - c));
Console.WriteLine("the traingle area is" + area);
Console.ReadKey();
}
}
}
}
file:///C:/Users/UMME KULSUM/do...
please enter the value of side a

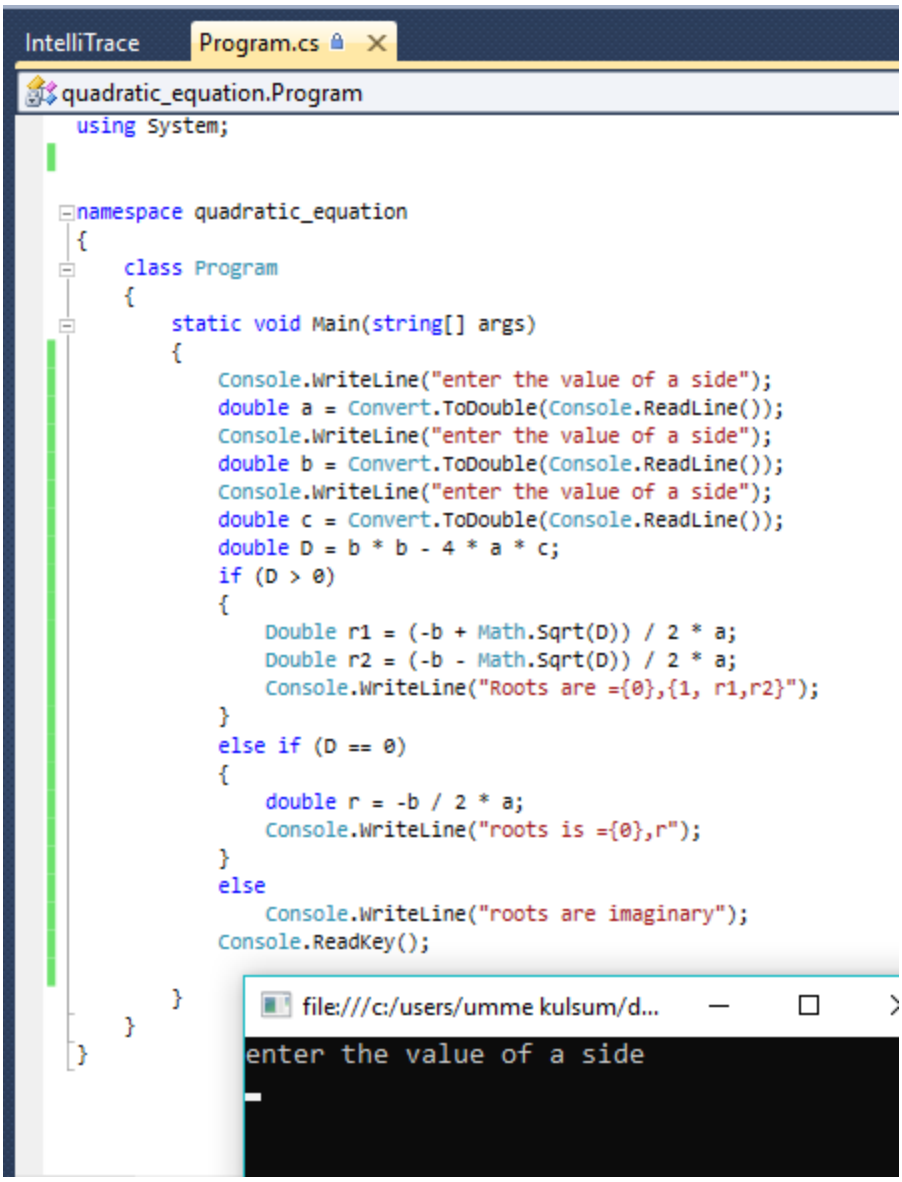

IntelliTrace
Program.cs
Destructor_Method.destructor
using System;
namespace Destructor_Method
{
class Animal
{
public Animal()
{
Console.WriteLine("constructing....");
}
~Animal()
{
Console.WriteLine("Destructing....");
}
}
public class destructor
{
public static void Main(string[]args)
{
Animal obj=new Animal();
Console.ReadKey();
}
}
}
file:///C:/Users/UMME KULSUM/...
constructing....

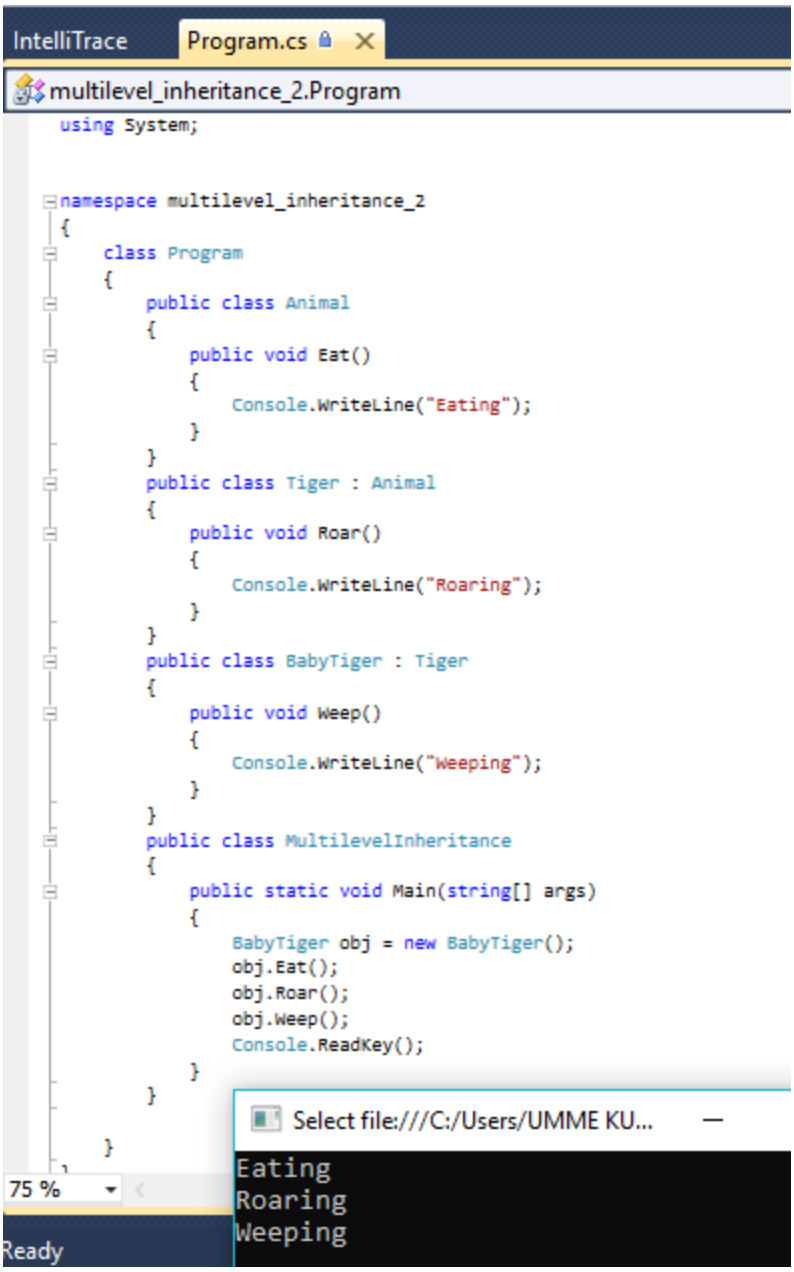

IntelliTrace
Program.cs
Animal
using System;
class Animal
{
public Animal()
{
Console.WriteLine("constructing.....");
}
}
public class constructor{
public static void Main(string[]args){
Animal obj=new Animal();
Console.ReadKey();
}
}
file:///C:/Users/UMME KULSUM...
constructing.....


IntelliTrace
Program.cs
Single_Inheritance.Animal
using System;
namespace Single_Inheritance
{
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();
}
}
}
100 %
file:///C:/Users/UMME KULSUM/...
Eating
Roaring


IntelliTrace
Program.cs
series.Program
using System;
namespace series
{
class Program
{
static void Main(string[] args)
{
int i, sum = 0;
for (i = 1;i <= 100; i++) {
sum =sum +i;
}
Console.WriteLine("sum of1+2+3+4.....+100 ={0}",sum);
Console.ReadKey();
}
}
}
file:///C:/Users/UMME KULSUM/D...
sum of1+2+3+4.....+100 =5050


IntelliTrace
Program.cs
Multiple_Inheritance2.Interface
using System;
namespace Multiple_Inheritance2
{
interface A
{
void myMethod();
}
interface B
{
void myotherMethod();
}
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();
}
}
}
Select file:///C:/Users/UMME KU...
sometext
someothertext
83 %

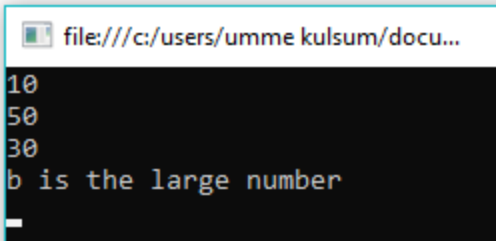

IntelliTrace
Program.cs
quadratic_equation.Program
using System;
namespace quadratic_equation
{
class Program
{
static void Main(string[] args)
{
Console.WriteLine("enter the value of a side");
double a = Convert.ToDouble(Console.ReadLine());
Console.WriteLine("enter the value of a side");
double b = Convert.ToDouble(Console.ReadLine());
Console.WriteLine("enter the value of a side");
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("Roots are ={0},{1, r1,r2}");
}
else if (D == 0)
{
double r = -b / 2 * a;
Console.WriteLine("roots is ={0},r");
}
else
Console.WriteLine("roots are imaginary");
Console.ReadKey();
}
}
}
file:///c:/users/umme kulsum/d...
enter the value of a side

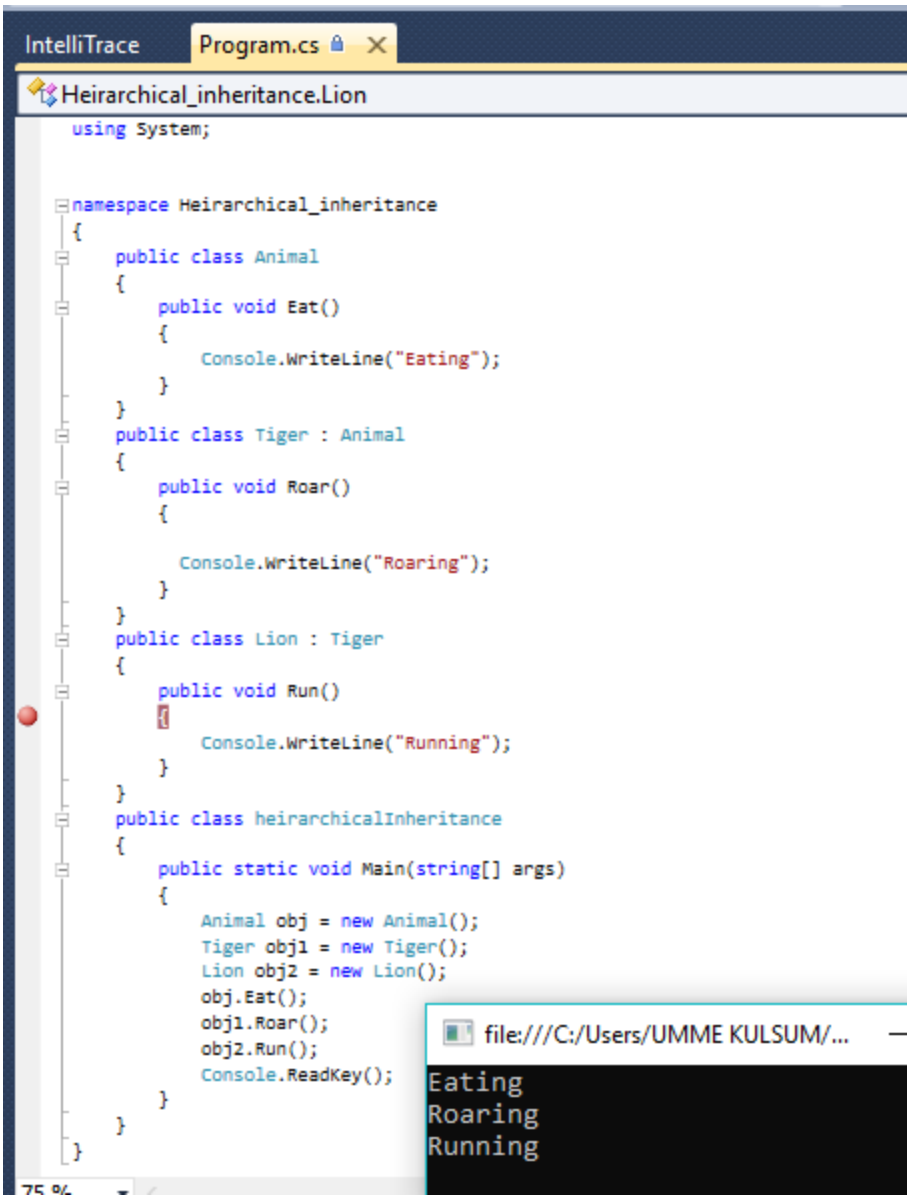

IntelliTrace
Program.cs
multilevel_inheritance_2.Program
using System;
namespace multilevel_inheritance_2
{
class Program
{
public class Animal
{
public void Eat()
{
Console.WriteLine("Eating");
}
}
public class Tiger : Animal
{
public void Roar()
{
Console.WriteLine("Roaring");
}
}
public class BabyTiger : 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();
}
}
}
75 %
Ready
Select file:///C:/Users/UMME KU...
Eating
Roaring
Weeping

largenumber__2.Program

```
using System;


namespace largenumber__2
{
    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 > b)
            {
                Console.WriteLine("a is the large number");
            }
            else if (b > a && b > c)
            {
                Console.WriteLine("b is the large number");
            }
            else
            {
                Console.WriteLine("c is the large number");
            }
            Console.ReadKey();
        }
    }
}
```




IntelliTrace
Program.cs
Heirarchical_inheritance.Lion
using System;
namespace Heirarchical_inheritance
{
public class Animal
{
public void Eat()
{
Console.WriteLine("Eating");
}
}
public class Tiger : Animal
{
public void Roar()
{
Console.WriteLine("Roaring");
}
}
public class Lion : Tiger
{
public void Run()
{
Console.WriteLine("Running");
}
}
public class heirarchicalInheritance
{
public static void Main(string[] args)
{
Animal obj = new Animal();
Tiger obj1 = new Tiger();
Lion obj2 = new Lion();
obj.Eat();
obj1.Roar();
obj2.Run();
Console.ReadKey();
}
}
}
file:///C:/Users/UMME KULSUM/...
Eating
Roaring
Running