DAY 10 MORNING ASSIGNMENT

-- BY SARATH KASIMSETTY

1) Write the two points of discussed about of inheritance.

- Inheritance is a process of reusing parent class method in child class method.
- Inheritance main goals is:
  Re-Usability and remove the duplicate code.

- **Types of Inheritance:**
  - Single inheritance
  - Multilevel inheritance
  - Multiple inheritance

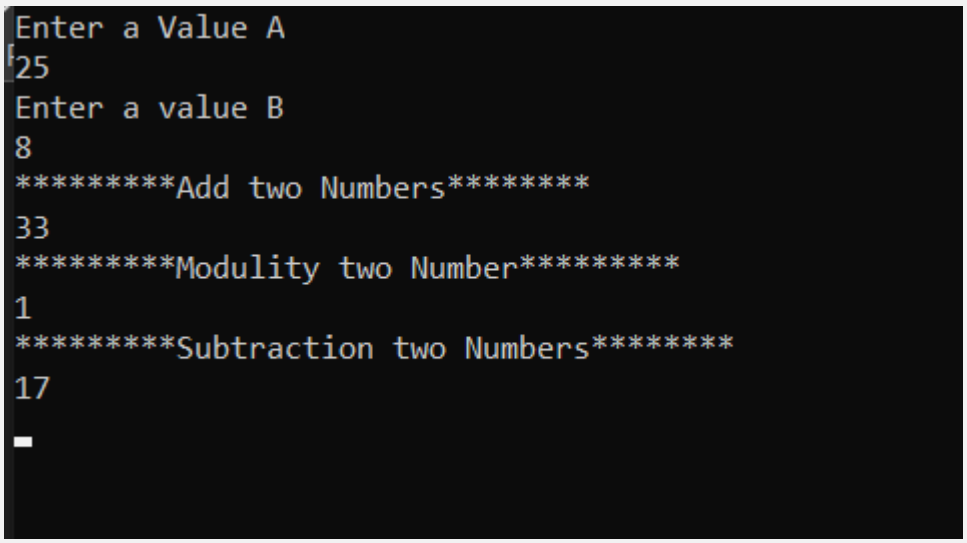
2) Write example code for:
a. Single inheritance
b. Multi level inheritance

SINGLE INHERITANCE CODE:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
//sarath kasimsetty
//Write example code for:
//a.Single inheritance b. Multi level inheritance
namespace Day10MorningProject2
{
    /// <summary>
    /// single inheritance
    /// </summary>
    internal class Week1
    {
        public int a;
        public int b;
        /// <summary>
        /// read input from input
        /// </summary>
        public void Readinput()
        {
            Console.WriteLine("Enter a Value A");
            a = Convert.ToInt32(Console.ReadLine());

            Console.WriteLine("Enter a value B");
            b = Convert.ToInt32(Console.ReadLine());
        }
        public int Add()
        {
            return a + b;
        }
        public int Sub()
        {
            return a - b;
        }
    }

    class Week2 :Week1
    {

        public int product()
        {
            return a * b;
        }
        public int Modulity()
        {
            return a % b;
        }
    }

    internal class Program
    {
        static void Main(string[] args)
        {
```

```
            Week2 sol = new Week2();

            sol.Readinput();
            Console.WriteLine("*********Add two Numbers********");
            Console.WriteLine(sol.Add());

            Console.WriteLine("*********Modulity two Number*********");
            Console.WriteLine(sol.Modulity());

            Console.WriteLine("*********Subtraction two Numbers********");
            Console.WriteLine(sol.Sub());

            Console.ReadLine();
        }
    }
}
```

OUTPUT:



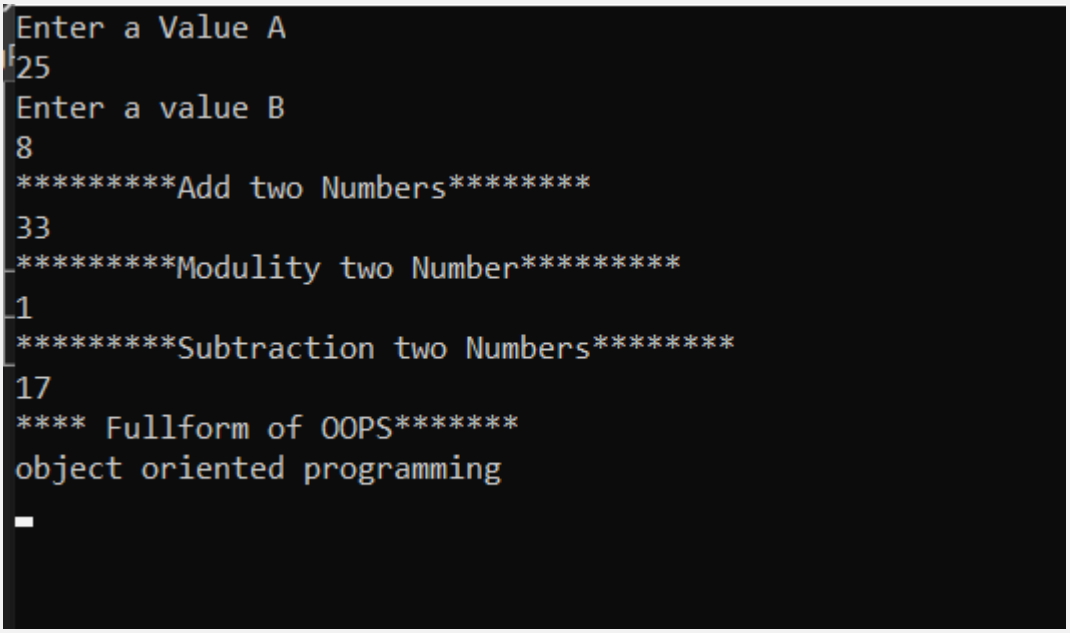
MULTILEVEL INHERITANCE CODE:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
//sarath kasimsetty
//Write example code for:
//a.Single inheritance b. Multi level inheritance
namespace Day10MorningProject2
{
    /// <summary>
    /// Multilevel inheritance
    /// </summary>
    internal class Week1
    {
```

```
        public int a;
        public int b;
        /// <summary>
        /// read input from input
        /// </summary>
        public void Readinput()
        {
            Console.WriteLine("Enter a Value A");
            a = Convert.ToInt32(Console.ReadLine());

            Console.WriteLine("Enter a value B");
            b = Convert.ToInt32(Console.ReadLine());
        }
        public int Add()
        {
            return a + b;
        }
        public int Sub()
        {
            return a - b;
        }
    }
    /// <summary>
    /// week1 is parent class and week2 is chils class
    /// </summary>
    class Week2 :Week1
    {

        public int product()
        {
            return a * b;
        }
        public int Modulity()
        {
            return a % b;
        }
    }
    /// <summary>
    /// week3 is child class and week2 is parent class
    /// </summary>
    class Week3 : Week2
    {
        /// <summary>
        /// FULL FORM OF OOPS
        /// </summary>
        /// <returns>Oops</returns>
        public string Oops()
        {
            return "object oriented programming";
        }
    }

    internal class Program
    {
        static void Main(string[] args)
```

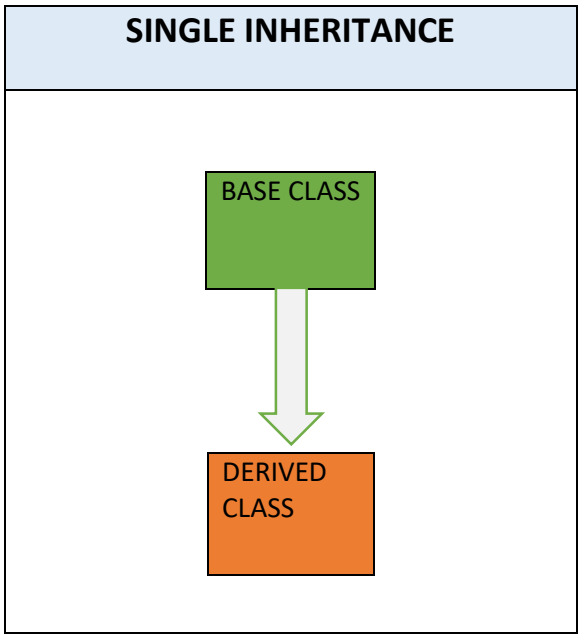
```
        {
            Week3 sol = new Week3();

            sol.Readinput();
            Console.WriteLine("*********Add two Numbers********");
            Console.WriteLine(sol.Add());

            Console.WriteLine("*********Modulity two Number*********");
            Console.WriteLine(sol.Modulity());

            Console.WriteLine("*********Subtraction two Numbers********");
            Console.WriteLine(sol.Sub());

            Console.WriteLine("**** Fullform of OOPS*******");
                Console.WriteLine(sol.Oops());

            Console.ReadLine();
        }
    }
}
```

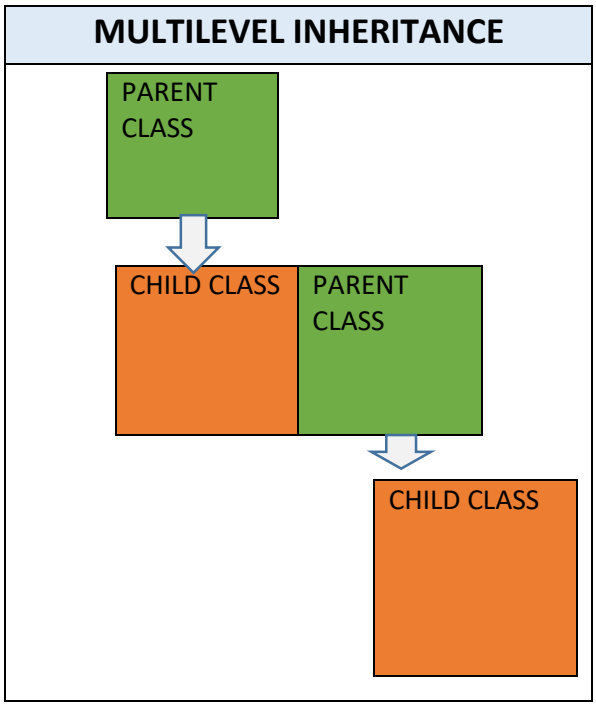
OUTPUT:

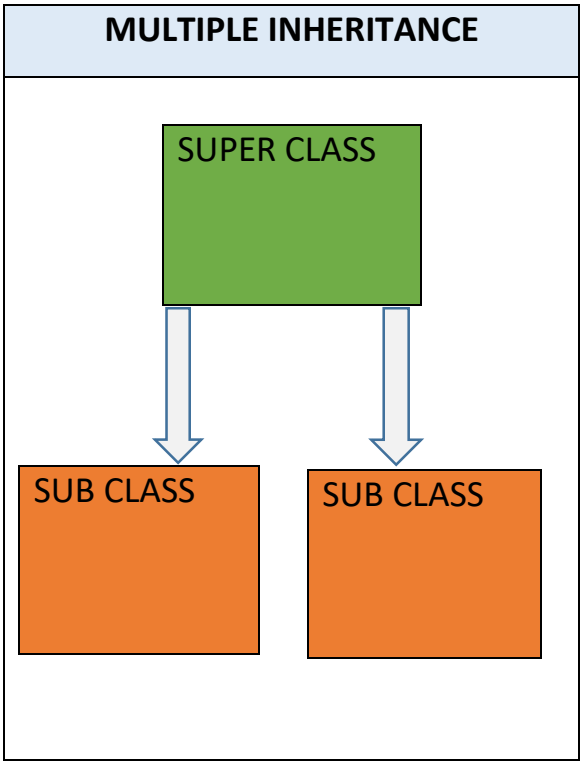


3) Pictorially represent 3 types of inheritance discussed in the class.

- Single Inheritance
- Multilevel Inheritance
- Multiple Inheritance







4) Why multiple inheritance is not supported for classes in C#

- C# does not support multiple class inheritance. To overcome this problem we use interfaces to achieve multiple class inheritance
- We don't consider Multiple Inheritance in c# because it causes ambiguity of methods from different base class
- This Multiple Inheritance causes DIAMOND PROBLEMS. • The diamond problem is an ambiguity that arises when two classes B and C inherit from A, and class D inherits from both B and C. It is called the diamond problem.

5) What is Polymorphism .

- A Ability of an object to take on many forms.
- These are of two types. They are
  - **METHOD OVERLOADING**
  - **METHOD OVERRIDING**

METHOD OVERLOADING :

Method overloading is to use same multiple methods within the same class with different parameters irrespective of return type.

METHOD OVERRIDING :

Method overriding is used to modify or re-write the data in the same class when it is inherited.

• Method overriding is only possible in derived classes, not within the same class where the method is declared

• Base class must use the **NEW** keywords to declare a method. Then only can a method be overridden

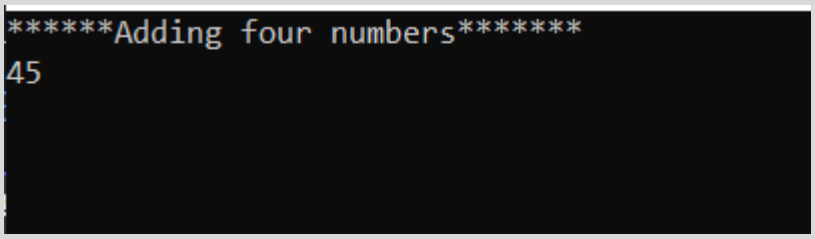
6) Write sample code for method overloading .

CODE :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
//sarath kasimsetty
//Write sample code for method overloading
namespace Day6MorningProject6
{
    internal class Program
    {
        /// <summary>
        /// overloading in polymorphism
        /// </summary>
        internal class Mathstask
        {
            public int Add(int a,int b )
```

```
            {
                return a + b;
            }
            public int Add(int a,int b,int c)
            {
                return a + b + c;
            }
            /// <summary>
            /// this is method is same but parameter are difference
            /// </summary>
            /// <param name="a"></param>
            /// <param name="b"></param>
            /// <param name="c"></param>
            /// <param name="d"></param>
            /// <returns>Add</returns>
            public int Add(int a ,int b ,int c,int d) { return a + b + c + d; }
        }
        static void Main(string[] args)
        {
            Mathstask obj = new Mathstask();

            Console.WriteLine("******Adding four numbers*******");
            Console.WriteLine(obj.Add(25, 10,5,5));

            Console.ReadLine();

        }
    }
}
```

OUTPUT:



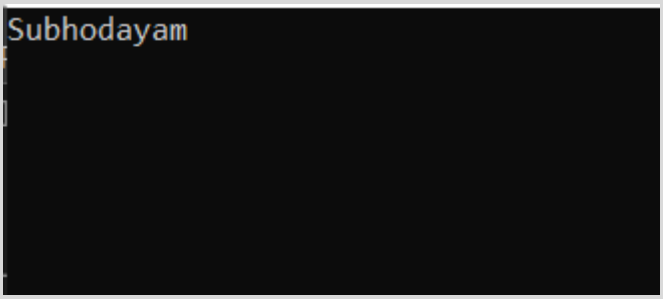
7) Write sample code for method overriding[using new key word]

CODE:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
//sarath kasimsetty
//Write sample code for method overriding[ using new key word ]
```

```
namespace Day10MorningProject7
{
    internal class Program
    {
        class Englishmessage
        {
            public void PrintHi()
            {
                Console.WriteLine("HI");
            }
            public void Printsarath()
            {
                Console.WriteLine("sarath");
            }
            public void PrintGM()
            {
                Console.WriteLine("GOOD MORNING");
            }
        }
        class Telugumessage : Englishmessage
        {
            public new void PrintGM()
            {
                Console.WriteLine("Subhodayam");
            }
        }
        static void Main(string[] args)
        {
            Telugumessage msg = new Telugumessage();

            msg.PrintGM();

            Console.ReadLine();

        }
    }
}
```

OUTPUT:



8) Research and write sample code for method overriding using virual, override keyword.

CODE:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
//SARATH KASIMSETTY
//Research and write sample code for method overriding virual,
override keyword.
namespace Day10MorningProject8
{
    internal class Program
    {
        class Englishmessage
        {
            public virtual void PrintHi()
            {
                Console.WriteLine("HI");
            }
            public virtual void PrintGM()
            {
                Console.WriteLine("GOOD MORNING");
            }
        }
        class Telugumessage : Englishmessage
        {
            public override void PrintGM()
            {
                Console.WriteLine("Namaskaaram");
            }
        }
        static void Main(string[] args)
        {
            Telugumessage msg = new Telugumessage();

            msg.PrintGM();

            Console.ReadLine();

        }
    }
```

```
}
```

OUTPUT: