Program.cs

```csharp
using System;

namespace CounterTask
{
    internal class Program
    {
        private static void PrintCounters(Counter[] myCounters)
        {
            foreach (Counter c in myCounters)
            {
                Console.WriteLine("{0} is {1}", c.Name, c.Ticks);
            }
        }

        static void Main(string[] args)
        {
            // Step 1: Let myCounters be an array of three Counter objects
            Counter[] myCounters = new Counter[3];

            // Step 2: myCounters[0] := new Counter with name "Counter 1"
            myCounters[0] = new Counter("Counter 1");

            // Step 3: myCounters[1] := new Counter with name "Counter 2"
            myCounters[1] = new Counter("Counter 2");

            // Step 4: myCounters[2] :=
            myCounters[2] = myCounters[0];
```

```csharp
        // Step 5: for i := 1 to 9
        for (int i = 1; i <= 9; i++)
        {
            // Step 6: Tell myCounters[0] to Increment
            myCounters[0].Increment();
        }

        // Step 7: for i := 1 to 14
        for (int i = 1; i <= 14; i++)
        {
            // Step 8: Tell myCounters[1] to Increment
            myCounters[1].Increment();
        }

        // Step 9: Tell Program to PrintCounters(myCounters)
        PrintCounters(myCounters);

        // Step 10: Tell myCounters[2] to Reset
        myCounters[2].Reset(); // This will reset Counter 3

        // Step 11: Tell Program to PrintCounters(myCounters)
        PrintCounters(myCounters);
    }
  }
}
```

Counter.cs

```csharp
using System;
```

```csharp
using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;


namespace CounterTask

{

    public class Counter

    {


        private int _count;

        private string _name;



        public Counter(string name)

        {

            _name = name;

            _count = 0;

        }

        public void Increment()

        { _count++; }


        public void Reset()

        { _count = 0; }


        public string Name

        {

            get

            {
```

```csharp
                return _name;
        }
        set
        {
            _name = value;


        }



    }
    public int Ticks
    {
        get
        {
            return _count;


        }
    }



    }
}
```

```
Counter 1 is 9
Counter 2 is 14
Counter 1 is 9
Counter 1 is 0
Counter 2 is 14
Counter 1 is 0

C:\Users\joshu\OneDrive\Desktop\COS20007 OOP\CounterTask\CounterTask\bin\Debug\net8.0\CounterTask.exe (process 17500) ex
ited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```