

## TP Modul 12

Muhammad Daniel Anugrah Pratama  
2311104063  
SE07-02

### A. Source Code Class *Helper.cs* :

```
namespace TP12
{
    public class Helper
    {
        public static string CariTandaBilangan(int a)
        {
            if (a < 0) return "Bilangan Negatif";
            else if (a > 0) return "Bilangan Positif";
            else return "Bilangan Nol";
        }
    }
}
```

### Penjelasannya:

Codingan pada class *Helper.cs* berfungsi untuk menentukan tanda dari sebuah bilangan bulat dengan mengembalikan nilai berupa string yang menunjukkan apakah bilangan tersebut negatif, positif, atau nol. Fungsi *CariTandaBilangan* dideklarasikan sebagai metode statis dalam kelas *Helper*, sehingga dapat dipanggil langsung tanpa membuat objek dari kelas tersebut. Jika parameter bilangan (*a*) bernilai kurang dari nol, maka akan dikembalikan string "Bilangan Negatif"; jika lebih dari nol, akan dikembalikan "Bilangan Positif"; dan jika sama dengan nol, akan dikembalikan "Bilangan Nol". Kode ini berguna sebagai fungsi bantu (utility) dalam proses analisis bilangan, validasi input, atau tampilan informasi kepada pengguna.

*Source Code Class Program.cs :*

```
using System;
using System.Windows.Forms;

namespace TP12
{
    static class Program
    {
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false); Application.Run(new
            Form1());
        }
    }
}
```

Penjelasan :

Class Program.cs adalah program utama (entry point) dari aplikasi Windows Forms berbasis .NET, yang ditulis dalam bahasa C#. Program ini berada dalam namespace TP12 dan didefinisikan di dalam kelas statis Program. Metode Main() adalah titik awal eksekusi aplikasi, yang diberi atribut [STAThread] untuk menunjukkan bahwa aplikasi ini menggunakan model threading STA (Single Threaded Apartment), yang diperlukan untuk menjalankan aplikasi Windows Forms. Di dalam Main(), baris Application.EnableVisualStyles() mengaktifkan tampilan visual modern sesuai tema Windows, sedangkan Application.SetCompatibleTextRenderingDefault(false) mengatur metode default untuk merender teks (menggunakan GDI+ daripada GDI). Akhirnya, Application.Run(new Form1()) memulai aplikasi dan membuka form utama bernama Form1, yang menjadi antarmuka pengguna dari program. Kode ini merupakan kerangka dasar untuk menjalankan aplikasi GUI menggunakan Windows Forms di Visual Studio.

### Source Code Class Unittest1.cs

```
using Microsoft.VisualStudio.TestTools.UnitTesting; using
TP12;

namespace TP12Tests
{
    [TestClass]
    public class UnitTest1
    {
        [TestMethod]
        public void TestNegatif()
        {
            string result = Helper.CariTandaBilangan(-10); Assert.AreEqual("Bilangan
            Negatif", result);
        }

        [TestMethod]
        public void TestPositif()
        {
            string result = Helper.CariTandaBilangan(10);
            Assert.AreEqual("Bilangan Positif", result);
        }

        [TestMethod]
        public void TestNol()
        {
            string result = Helper.CariTandaBilangan(0);
            Assert.AreEqual("Bilangan Nol", result);
        }
    }
}
```

### Penjelasan :

Pada class Unittest1.cs merupakan bagian dari unit testing dalam bahasa C# menggunakan framework MSTest, yang digunakan untuk menguji kebenaran fungsi CariTandaBilangan() dari kelas Helper di dalam proyek TP12. Kode ini berada dalam namespace TP12Tests dan didefinisikan dalam kelas UnitTest1, yang ditandai dengan atribut [TestClass] sebagai penanda bahwa kelas ini berisi metode-metode pengujian. Setiap metode pengujian ditandai dengan atribut [TestMethod], yang menandakan bahwa metode tersebut akan dijalankan sebagai bagian dari pengujian

otomatis. Ada tiga metode pengujian: TestNegatif() menguji input -10 dan memastikan hasilnya adalah "Bilangan Negatif", TestPositif() menguji input 10 dan memverifikasi hasilnya adalah "Bilangan Positif", serta TestNol() menguji input 0 dan memeriksa bahwa hasilnya adalah "Bilangan Nol". Semua pengujian menggunakan Assert.AreEqual() untuk membandingkan hasil aktual dengan hasil yang diharapkan. Tujuan utama dari kode ini adalah memastikan bahwa fungsi CariTandaBilangan() bekerja dengan benar untuk ketiga kondisi bilangan (negatif, positif, dan nol).

*Source Code Form1.cs*

```
using System;
using System.Windows.Forms;

namespace TP12
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        private void Form1_Load(object sender, EventArgs e)
        {
            labelOutput.Text = "Hasil Output";
        }

        public string CariTandaBilangan(int a)
        {
            if (a < 0) return "Negatif";
            else if (a > 0) return "Positif";
            else return "Nol";
        }

        private void buttonCek_Click(object sender, EventArgs e)
        {
            if (int.TryParse(textBoxInput.Text, out int input))
            {
                string hasil = Helper.CariTandaBilangan(input);
                labelOutput.Text = $"Hasil Output : {hasil}";
            }
            else
            {

```

```

        labelOutput.Text = "Hasil Output: Input tidak valid!";
    }
}

private void labelOutput_Click(object sender, EventArgs e)
{

}

private void pictureBoxSign_Click(object sender, EventArgs e)
{

}
}
}

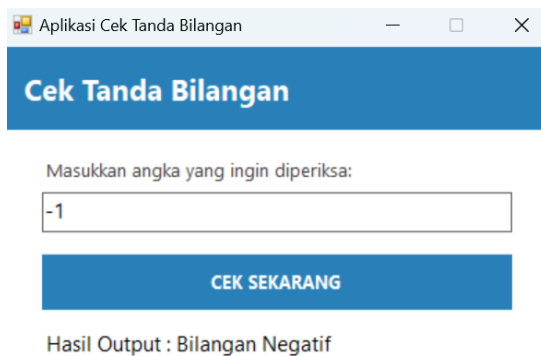
```

Penjelasan :

Pada Class Form1.cs merupakan bagian dari aplikasi Windows Forms dalam bahasa C#, yang mengatur tampilan dan logika interaksi pada Form1, yaitu jendela utama dari aplikasi. Kelas Form1 diturunkan dari kelas Form dan menggunakan pendekatan event- driven programming. Pada saat form dimuat (Form1\_Load), label labelOutput akan menampilkan teks default “Hasil Output”. Method CariTandaBilangan(int a) di dalam form ini berfungsi untuk menentukan apakah sebuah bilangan bertanda negatif, positif, atau nol. Namun, dalam proses utama pengecekan bilangan, digunakan method Helper.CariTandaBilangan() dari kelas luar. Ketika tombol buttonCek diklik (buttonCek\_Click), program akan mencoba membaca nilai dari textBoxInput. Jika input tersebut valid dan bisa dikonversi menjadi bilangan bulat (int.TryParse), maka fungsi Helper.CariTandaBilangan() dipanggil, dan hasilnya ditampilkan di labelOutput. Jika input tidak valid, label akan menampilkan pesan kesalahan. Dua event lainnya (labelOutput\_Click dan pictureBoxSign\_Click) tidak memiliki implementasi khusus dan hanya disediakan sebagai placeholder. Kode ini bertujuan memberikan antarmuka grafis bagi pengguna untuk mengecek tanda suatu bilangan dengan cara yang interaktif dan mudah digunakan.

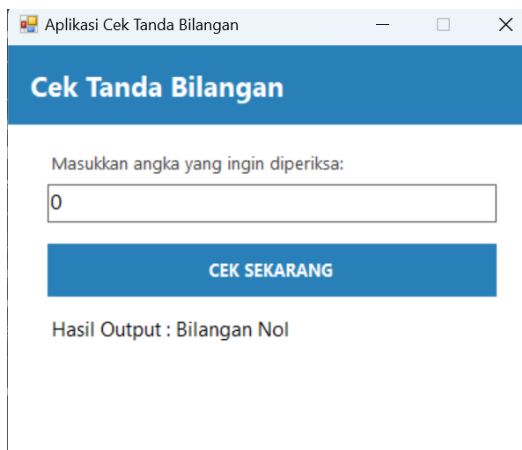
Output dari program :

1. Apabila menginput angka negatif



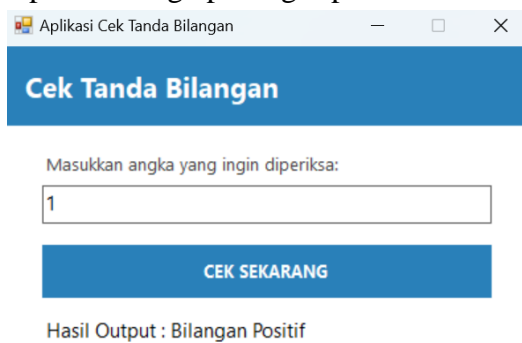
The screenshot shows a window titled "Aplikasi Cek Tanda Bilangan". The main heading is "Cek Tanda Bilangan". Below it, the instruction "Masukkan angka yang ingin diperiksa:" is followed by a text input field containing "-1". A blue button labeled "CEK SEKARANG" is positioned below the input field. At the bottom, the output text reads "Hasil Output : Bilangan Negatif".

2. Apabila menginput angka 0 "Nol"



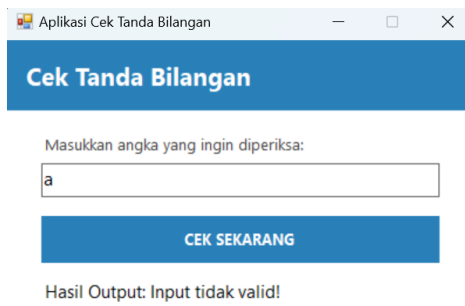
The screenshot shows the same application window. The text input field now contains "0". The "CEK SEKARANG" button remains below it. The output text at the bottom has changed to "Hasil Output : Bilangan Nol".

3. Apabila Menginput angka positif



The screenshot shows the application window with the text input field containing "1". The "CEK SEKARANG" button is still present. The output text at the bottom now displays "Hasil Output : Bilangan Positif".

#### 4. Apabila menginput selain angka



The screenshot shows a web browser window with the title 'Aplikasi Cek Tanda Bilangan'. The page has a blue header with the text 'Cek Tanda Bilangan'. Below the header, there is a label 'Masukkan angka yang ingin diperiksa:' followed by a text input field containing the letter 'a'. Below the input field is a blue button with the text 'CEK SEKARANG'. At the bottom of the page, the text 'Hasil Output: Input tidak valid!' is displayed.

Aplikasi Cek Tanda Bilangan

### Cek Tanda Bilangan

Masukkan angka yang ingin diperiksa:

CEK SEKARANG

Hasil Output: Input tidak valid!