

**JURNAL MODUL 12**  
**PERFORMANCE ANALYSIS UNIT TESTING DAN DEBUGGING**



Nama :

Alya Rabani (2311104076)

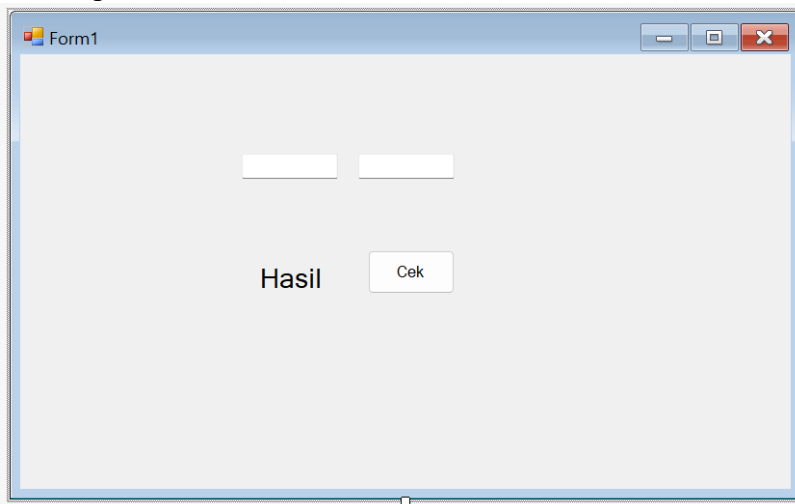
Dosen :

YUDHA ISLAMI SULISTYA

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK**  
**TELKOM UNIVERSITY PURWOKERTO**  
**2025**

## 1. Membuat project GUI

Pertama-tama membuat tampilan antarmuka, dengan mengambil TextBox untuk input angka, kemudian button yang akan menjadi tempat pengecekan, lalu label untuk menampilkan hasil.



Pada UI tersebut terdapat beberapa logika seperti `buttonHitung_Click` untuk menjalankan button agar saat nilai di inputkan pada textbox dan di cek button hasilnya bisa keluar pada label.

```
1 reference
private void buttonHitung_Click(object sender, EventArgs e)
{
    if (int.TryParse(textBox1.Text, out int a) && int.TryParse(textBox2.Text, out int b))
    {
        int hasil = CariNilaiPangkat(a, b);
        Hasil.Text = $"Hasil: {hasil}";
    }
    else
    {
        Hasil.Text = "Input tidak valid.";
    }
}

1 reference
public int CariNilaiPangkat(int a, int b)
{
    if (b < 0)
        return -2;

    try
    {
        long hasil = 1;

        for (int i = 0; i < b; i++)
        {
            hasil *= a;
            if (hasil > int.MaxValue || hasil < int.MinValue)
                return -3;
        }

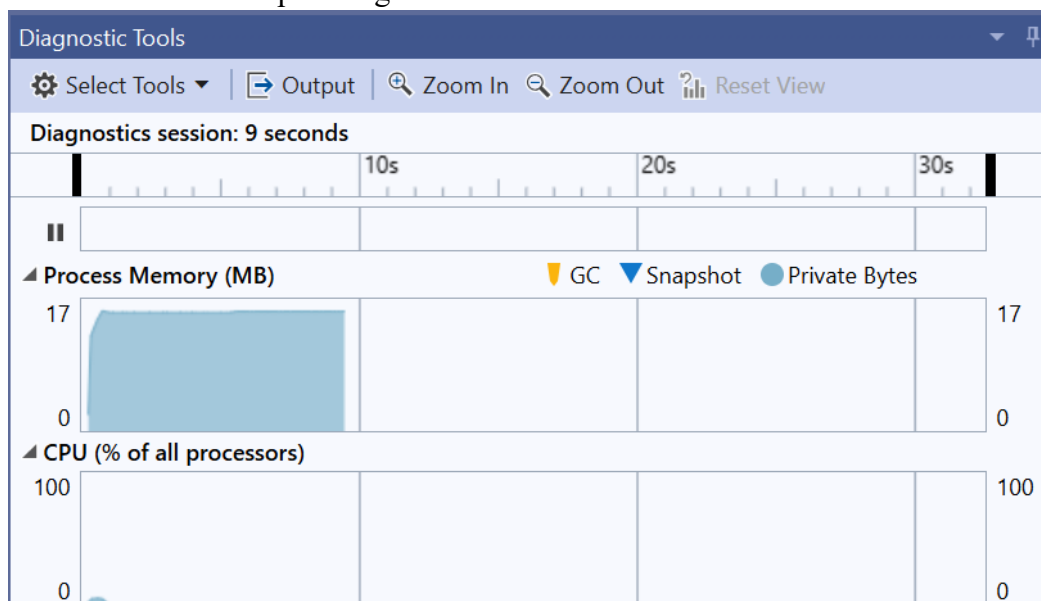
        return (int)hasil;
    }
    catch
    {
        return -1;
    }
}
```

Form1

8 2

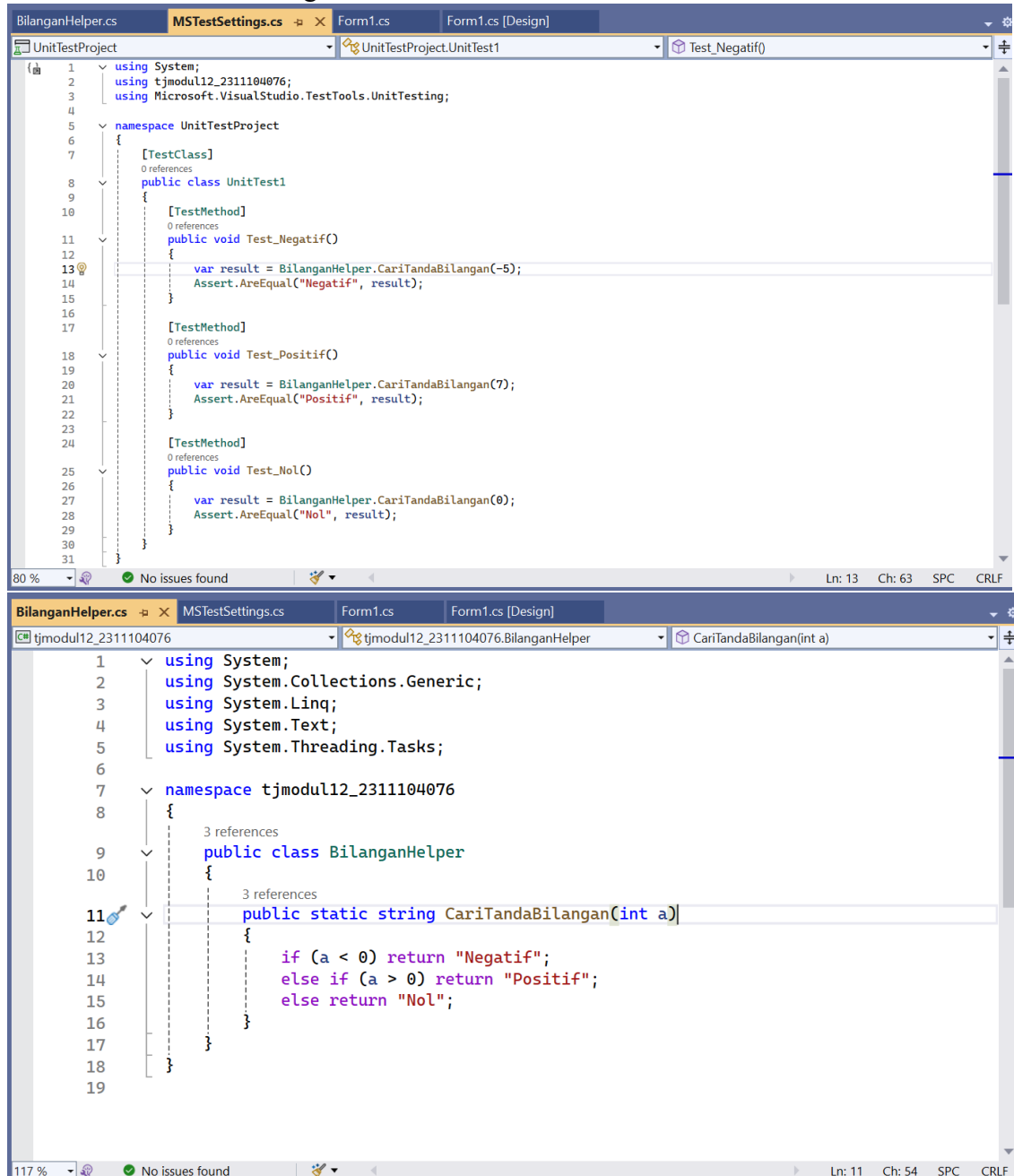
Hasil: 64 Cek

## 2. Melakukan software profiling



### 3. Menambahkan unit testing

Pada testing saya membuat sebuah kelas sendiri dulu untuk mengecek logika tersebut, kemudian pada MSTestSetting terdapat beberapa method test untuk memanggil method CariTandaBilangan dengan berbagai input. Disini sudah tidak terdapat error yang menandakan bahwa testing berhasil.



The image consists of two screenshots of the Visual Studio code editor. The top screenshot shows the file `UnitTestProject\UnitTest1\Test_Negatif()` being edited. It contains three unit tests: `Test_Negatif()`, `Test_Positif()`, and `Test_No()`. Each test calls `BilanganHelper.CariTandaBilangan()` with a specific input and asserts the result. The bottom screenshot shows the file `tjmodul12_2311104076\BilanganHelper` being edited. It defines a `BilanganHelper` class with a static method `CariTandaBilangan(int a)` that returns "Negatif" for negative numbers, "Positif" for positive numbers, and "No" for zero.

```
1 using System;
2 using tjmodul12_2311104076;
3 using Microsoft.VisualStudio.TestTools.UnitTesting;
4
5 namespace UnitTestProject
6 {
7     [TestClass]
8     public class UnitTest1
9     {
10         [TestMethod]
11         public void Test_Negatif()
12         {
13             var result = BilanganHelper.CariTandaBilangan(-5);
14             Assert.AreEqual("Negatif", result);
15         }
16
17         [TestMethod]
18         public void Test_Positif()
19         {
20             var result = BilanganHelper.CariTandaBilangan(7);
21             Assert.AreEqual("Positif", result);
22         }
23
24         [TestMethod]
25         public void Test_No()
26         {
27             var result = BilanganHelper.CariTandaBilangan(0);
28             Assert.AreEqual("No", result);
29         }
30     }
31 }
```

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace tjmodul12_2311104076
8 {
9     public class BilanganHelper
10     {
11         public static string CariTandaBilangan(int a)
12         {
13             if (a < 0) return "Negatif";
14             else if (a > 0) return "Positif";
15             else return "No";
16         }
17     }
18 }
```