

Nama : Muhammad Fariz Nur Hidayat

Kelas : SE063

NIM : 2211104069

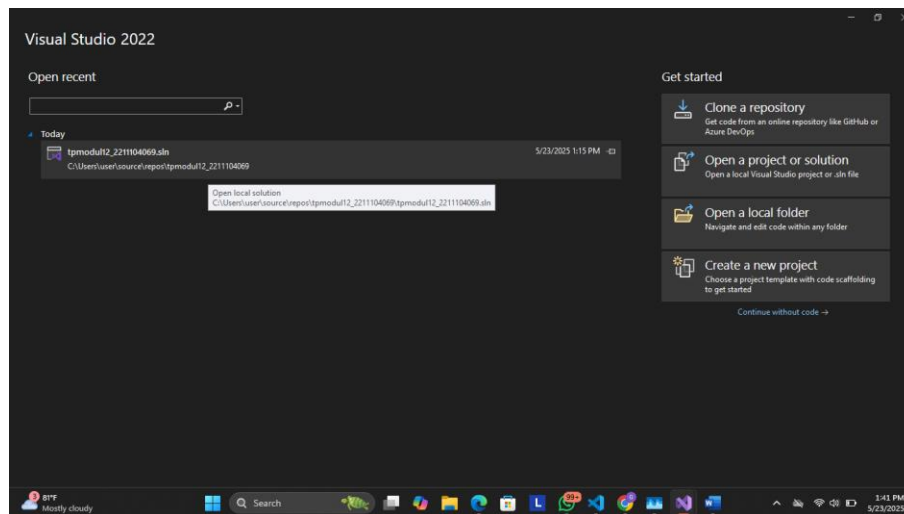
1. MEMBUAT PROJECT GUI BARU

Buka IDE misalnya dengan Visual Studio

A. Misalnya menggunakan Visual Studio, buatlah project baru dengan nama tpmodul12_NIM

B. Pastikan project yang dibuat dapat menggunakan GUI (misalnya tipe Windows Form pada Visual Studio).

Bukti :

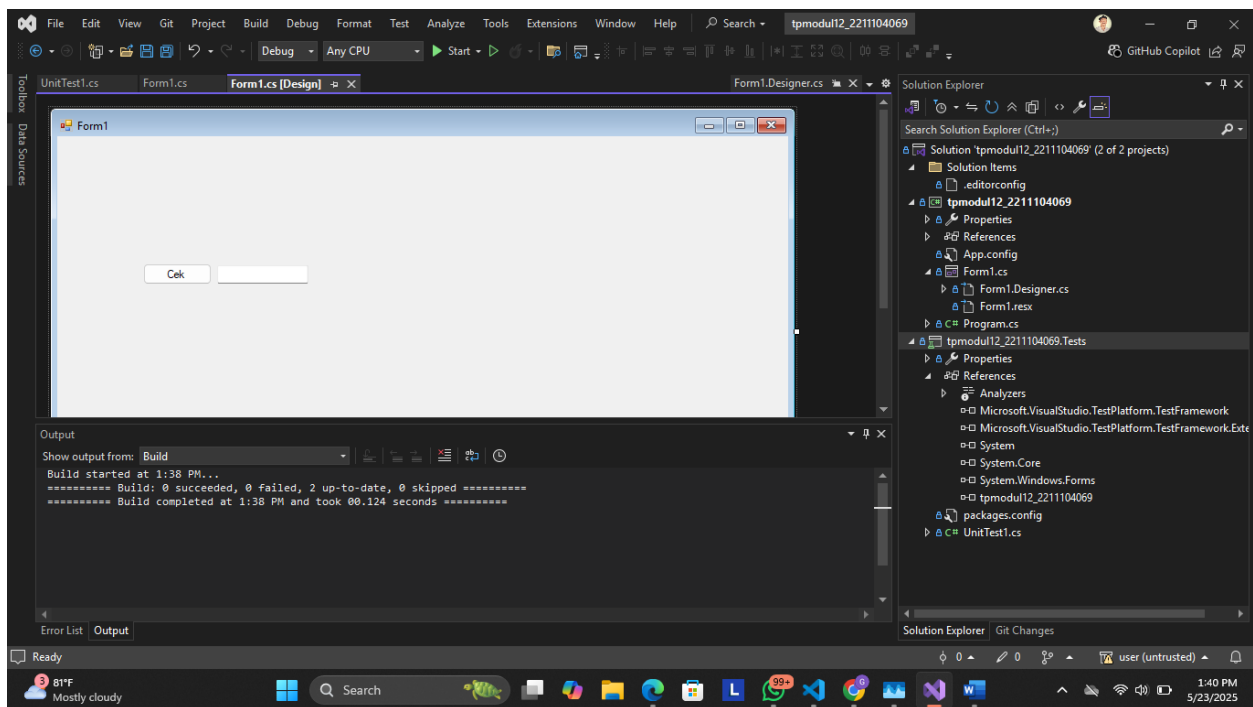


2. MEMBUAT GUI SEDERHANA DAN

Pada project yang telah dibuat sebelumnya:

A. Buatlah suatu Form atau tampilan GUI sederhana dengan satu buah textbox, satu button dan satu label untuk menampilkan output.

Bukti



B. Tambahkan satu method dengan nama “CariTandaBilangan(int a)” yang menerima satu input dan mengembalikan nilai string dengan aturan sebagai berikut:

- Apabila input a kurang dari 0 maka output adalah “Negatif”
- Apabila input a lebih dari 0 maka output adalah “Positif”
- Apabila input sama dengan 0 maka output adalah “Nol”

Form1.cs

```
using System;
```

```
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace tpmodul12_2211104069
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        // Method asli untuk menentukan tanda bilangan
        private string CariTandaBilangan(int a)
        {
            if (a < 0)
                return "Negatif";
            else if (a > 0)
                return "Positif";
            else
                return "Nol";
        }

        // Method publik untuk keperluan unit test
        public string InvokeCariTandaBilangan(int a)
        {
            return CariTandaBilangan(a);
        }

        private void buttonCek_Click_1(object sender, EventArgs e)
        {
            if (int.TryParse(textBoxInput.Text, out int input))
            {
                string hasil = CariTandaBilangan(input);
                labelOutput.Text = hasil;
            }
            else
            {

```

```
        labelOutput.Text = "Input tidak valid!";
    }
}

private void textBoxInput_TextChanged(object sender, EventArgs e)
{
    }
}
}
```

Penjelasan singkat source code:

Kode di atas merupakan implementasi aplikasi Windows Form sederhana bernama Form1 dalam proyek tpmodul12_2211104069. Aplikasi ini memiliki satu textbox untuk input angka, satu tombol untuk mengecek, dan satu label untuk menampilkan hasil. Fungsi utama CariTandaBilangan(int a) akan menentukan apakah angka yang dimasukkan adalah "Positif", "Negatif", atau "Nol", berdasarkan nilainya. Fungsi tersebut bersifat private, namun disediakan method publik InvokeCariTandaBilangan(int a) agar dapat diuji melalui unit test. Saat tombol ditekan, event buttonCek_Click_1 akan mengambil input dari textbox, memanggil fungsi CariTandaBilangan, dan menampilkan hasilnya ke label. Kode ini mendukung pengujian otomatis melalui unit test dengan cakupan yang baik.

C. Pada tampilan GUI, pada saat tombol ditekan, maka label output akan menampilkan hasil pangkat dari pemanggilan fungsi “CariTandaBilangan” dari input textbox yang diberikan.

Positif

Form1

Positif

Cek: 9

Negatif

Form1

Negatif

Cek: -5

Nol

Form1

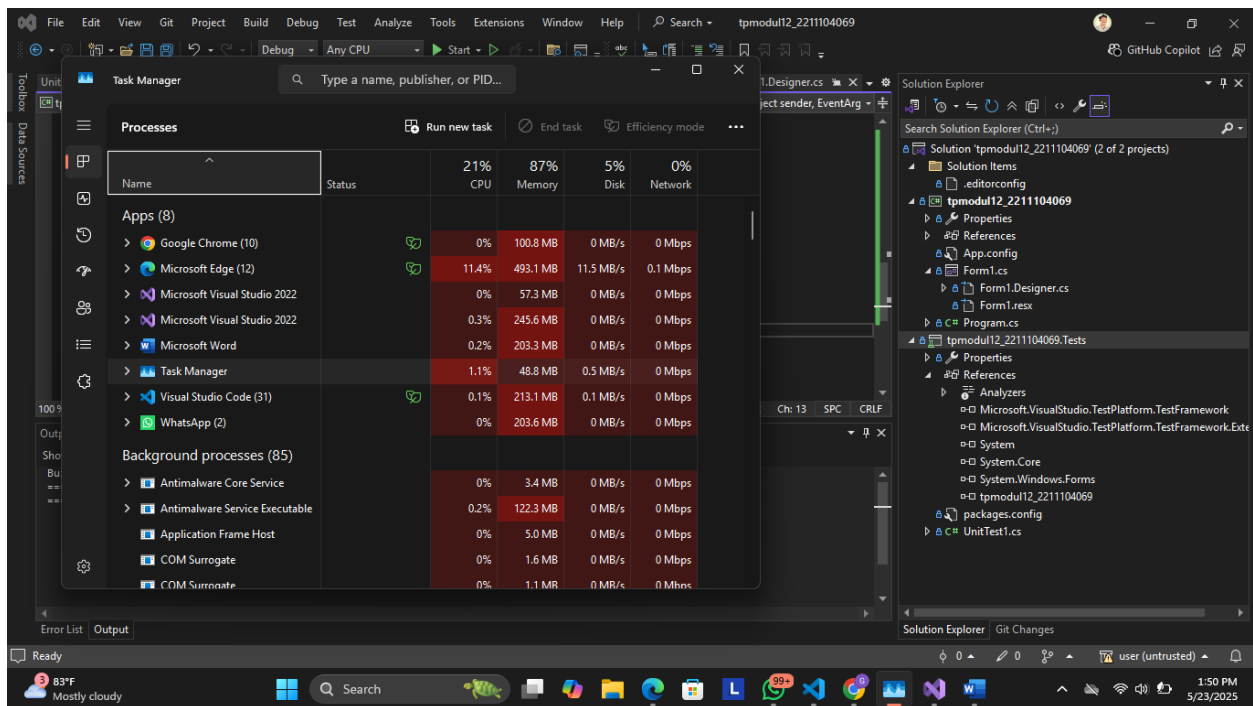
Nol

Cek: 0

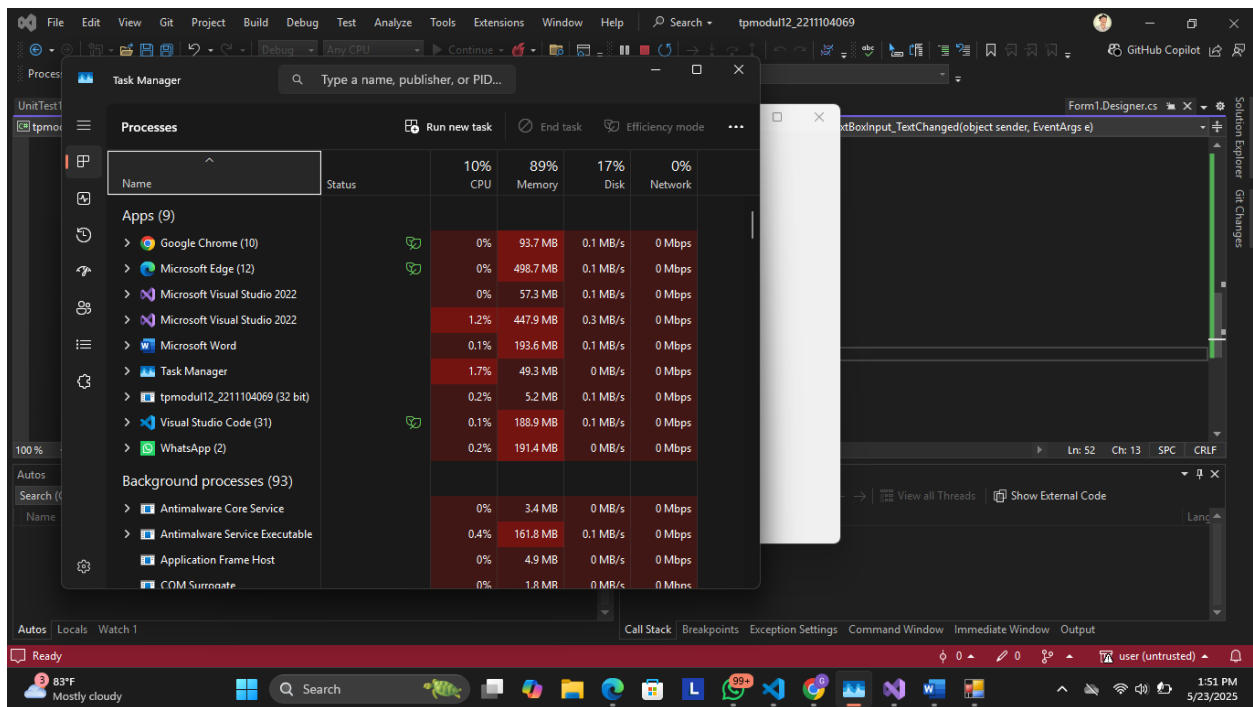
3. MELAKUKAN SOFTWARE PROFILING

Jalankan project yang dibuat sebelumnya dan jalankan profiling tools (misal dari visual studio, task manager atau sejenisnya):

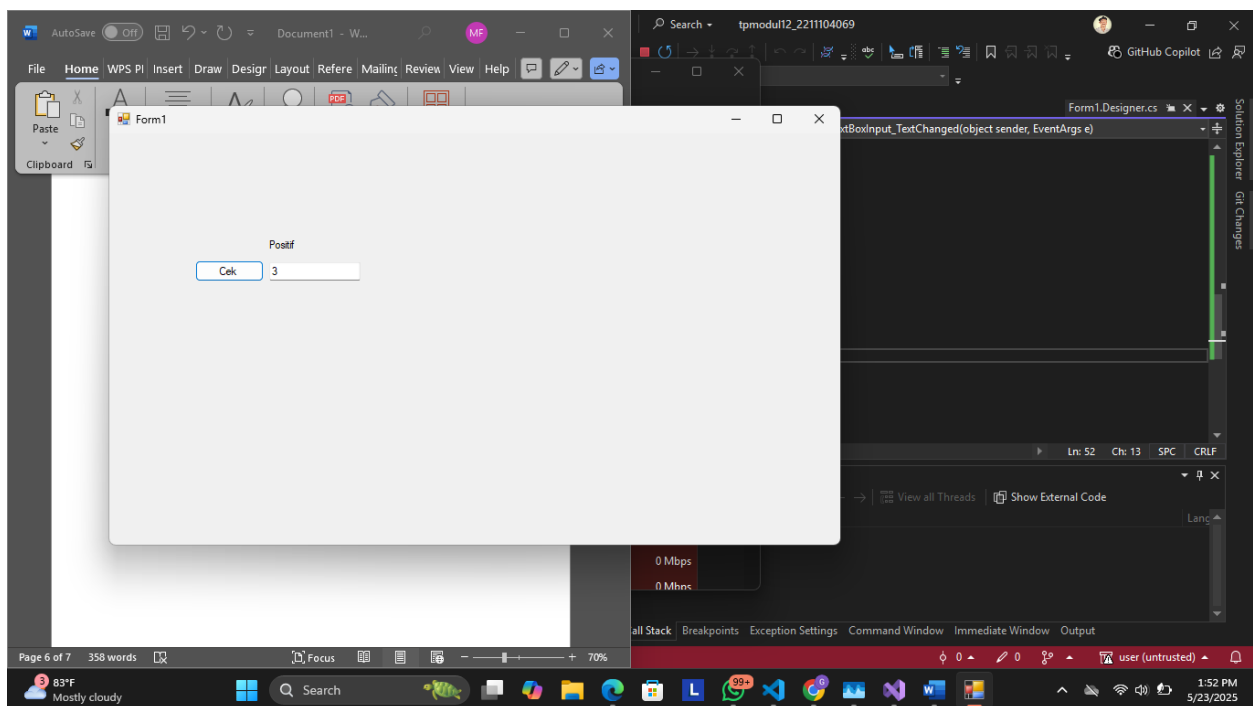
A. Pada saat program berjalan, catat dan amati CPU usage dari aplikasi yang sedang berjalan tanpa melakukan input apapun.



B. Pada saat program berjalan, catat dan amati memory usage dari aplikasi yang sedang berjalan tanpa melakukan input apapun.



C. Coba masukkan beberapa angka pada textbox dan tekan tombol button.



D. Laporkan apakah terdapat perubahan pada CPU usage dan memory (apabila tidak ada perubahan juga perlu dilaporkan di file docx).

The screenshot shows the Windows Task Manager window with the 'Processes' tab selected. It displays a list of running applications and background processes, along with their CPU, Memory, Disk, and Network usage. The processes are categorized into 'Apps (7)' and 'Background processes (89)'.

| Name | Status | CPU | Memory | Disk | Network |
|----------------------------------|--------|------|----------|-----------|----------|
| Apps (7) | | | | | |
| Google Chrome (6) | | 0.2% | 155.6 MB | 0 MB/s | 0 Mbps |
| Microsoft Edge (12) | | 9.2% | 513.3 MB | 11.6 MB/s | 0.1 Mbps |
| Microsoft Visual Studio 2022 | | 1.5% | 478.9 MB | 0 MB/s | 0 Mbps |
| Task Manager | | 2.7% | 76.4 MB | 0 MB/s | 0 Mbps |
| tpmodul12_2211104069 (32 bit) | | 0% | 6.2 MB | 0 MB/s | 0 Mbps |
| Visual Studio Code (31) | | 0% | 96.7 MB | 0.1 MB/s | 0 Mbps |
| WhatsApp (2) | | 0% | 223.5 MB | 0.1 MB/s | 0 Mbps |
| Background processes (89) | | | | | |
| Antimalware Core Service | | 0% | 2.0 MB | 0 MB/s | 0 Mbps |
| Antimalware Service Executable | | 1.1% | 187.4 MB | 0 MB/s | 0 Mbps |
| Application Frame Host | | 0% | 8.0 MB | 0.1 MB/s | 0 Mbps |
| COM Surrogate | | 0% | 1.1 MB | 0 MB/s | 0 Mbps |
| CTF Loader | | 0% | 5.6 MB | 0 MB/s | 0 Mbps |
| ... (other background processes) | | 0% | 1.1 MB | 0 MB/s | 0 Mbps |

4. MENAMBAHKAN UNIT TESTING

Di dalam project yang sama:

- Buatlah kode unit test untuk menguji method “CariTandaBilangan” yang dibuat sebelumnya.
- Pastikan kode unit test tersebut memiliki branch coverage yang baik untuk method “CariTandaBilangan”.

UnitTest1.cs

```
using Microsoft.VisualStudio.TestTools.UnitTesting;
using tpmodul12_2211104069; // Pastikan sesuai dengan namespace project utama

namespace tpmodul12_2211104069.Tests
{
    [TestClass]
    public class Form1Tests
    {
        [TestMethod]
        public void Test_Negatif()
        {
            var form = new Form1();
            Assert.AreEqual("Negatif", form.InvokeCariTandaBilangan(-5));
        }
    }
}
```



```

[TestMethod]
public void Test_Positif()
{
    var form = new Form1();
    Assert.AreEqual("Positif", form.InvokeCariTandaBilangan(7));
}

[TestMethod]
public void Test_Nol()
{
    var form = new Form1();
    Assert.AreEqual("Nol", form.InvokeCariTandaBilangan(0));
}
}
}

```

Penjelasan Kode :

Kode di atas adalah unit test untuk aplikasi Form1 dari proyek tpmodul12_2211104069, menggunakan framework **MSTest**. Class Form1Tests berisi tiga method test: Test_Negatif, Test_Positif, dan Test_Nol, masing-masing untuk menguji apakah fungsi InvokeCariTandaBilangan(int a) dalam Form1 mengembalikan string yang benar sesuai input bilangan negatif, positif, dan nol. Setiap method membuat instance dari Form1, memanggil fungsi yang diuji, lalu mencocokkan hasilnya dengan nilai yang diharapkan menggunakan Assert.AreEqual. Unit test ini memastikan logika program bekerja dengan benar dalam berbagai kasus input.

C. Jalankan kode unit test yang dibuat dan lampirkan hasil unit testing yang dilakukan.

Test Explorer

3

3

0

Search (Ctrl+I)

Ready

4 Warnings

0 Errors

| Test | Duration | Traits | Error Message |
|---------------------------------|----------|--------|---------------|
| tpmodule12_2211104069.Tests (3) | 441 ms | | |
| tpmodule12_2211104069.Tests (3) | 441 ms | | |
| Form1Tests (3) | 441 ms | | |
| Test_Negatif | 441 ms | | |
| Test_Nol | < 1 ms | | |
| Test_Positif | < 1 ms | | |

Run

Debug

Group Summary

tpmodule12_2211104069

Tests in group: 3

Total Duration: 441 ms

Outcomes

3 Passed