

**TUGAS PENDAHULUAN  
KONSTRUKSI PERANGKAT LUNAK**

**MODUL XII  
PERFORMANCE ANALYSIS, UNIT TESTING, DAN  
DEBUGGING**



**Disusun Oleh:**

**Aorinka Anendya Chazanah / 2211104013**

**S1 SE-06-01**

**Dosen Pengampu:**

**Yudha Islami Sulistya, S.Kom., M.Cs**

**PROGRAM STUDI S1 SOFTWARE ENGINEERING  
FAKULTAS INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO**

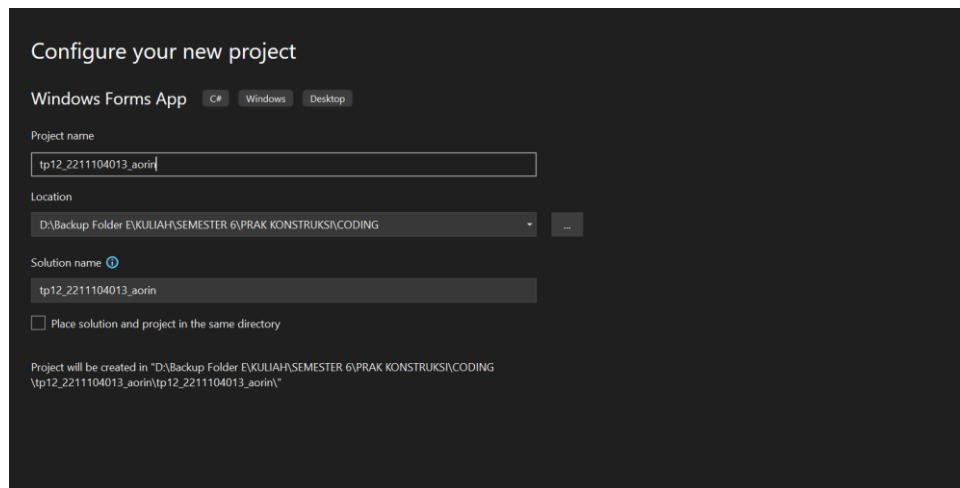
**2025**

## TUGAS PENDAHULUAN

### 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).



### 2. MEMBUAT GUI SEDERHANA

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

## Jawaban

### File Form1.cs

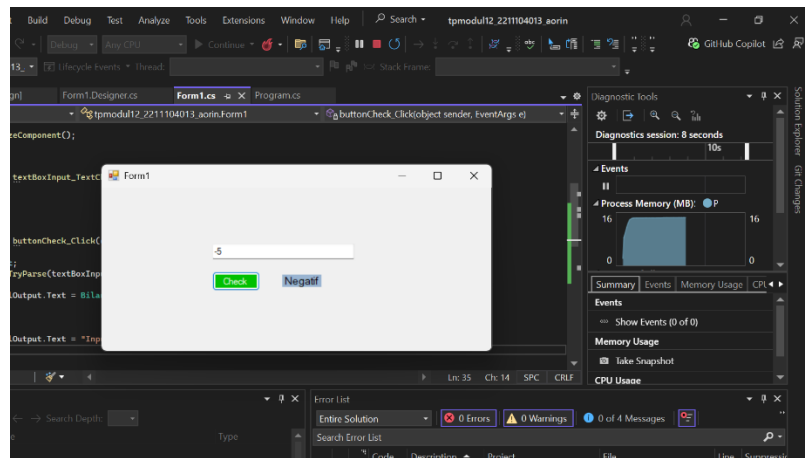
```
1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Drawing;
6  using System.Linq;
7  using System.Text;
8  using System.Threading.Tasks;
9  using System.Windows.Forms;
10
11  namespace tpmodul12_2211104013_aorin
12  {
13      3 references
14      public partial class Form1 : Form
15      {
16          1 reference
17          public Form1()
18          {
19              InitializeComponent();
20          }
21
22          0 references
23          private void textBoxInput_TextChanged(object sender, EventArgs e)
24          {
25          }
26
27          1 reference
28          private void buttonCheck_Click(object sender, EventArgs e)
29          {
30              int input;
31              if (int.TryParse(textBoxInput.Text, out input))
32              {
33                  labelOutput.Text = BilanganHelper.CariTandaBilangan(input);
34              }
35              else
36              {
37                  labelOutput.Text = "Input tidak valid";
38              }
39          }
40
41          0 references
42          private void Form1_Load(object sender, EventArgs e)
43          {
44          }
45      }
46  }
```

### File BilanganHelper.cs

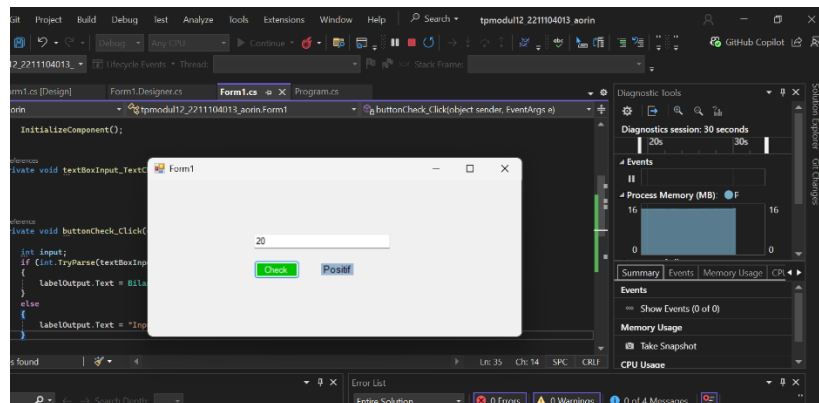
```
1  namespace tpmodul12_2211104013_aorin
2  {
3      4 references
4      public class BilanganHelper // <-- tambahkan public
5      {
6          4 references | 3/3 passing
7          public static string CariTandaBilangan(int a)
8          {
9              if (a < 0) return "Negatif";
10             if (a > 0) return "Positif";
11             return "Nol";
12         }
13     }
```

## Output

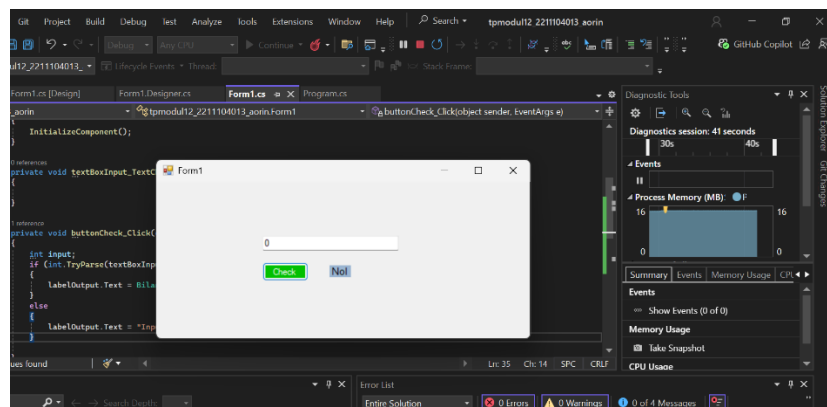
- Apabila input a kurang dari 0 maka output adalah “Negatif”



- Apabila input a lebih dari 0 maka output adalah “Positif”



- Apabila input sama dengan 0 maka output adalah “No!”



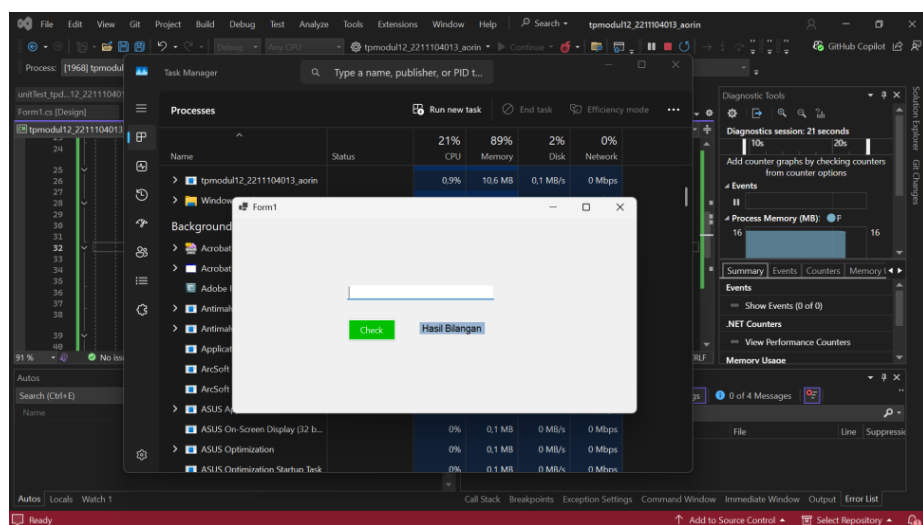
## Penjelasan program

Kode di atas merupakan implementasi dari form aplikasi Windows Forms yang digunakan untuk mengecek apakah suatu bilangan termasuk positif, negatif, atau nol. Saat pengguna memasukkan angka ke dalam `textBoxInput` dan menekan tombol `buttonCheck`, program akan mencoba mengubah teks tersebut menjadi bilangan bulat. Jika berhasil, maka hasil analisis bilangan ditampilkan melalui `labelOutput` dengan bantuan fungsi `BilanganHelper.CariTandaBilangan`. Jika input bukan angka valid, maka akan muncul pesan "Input tidak valid".

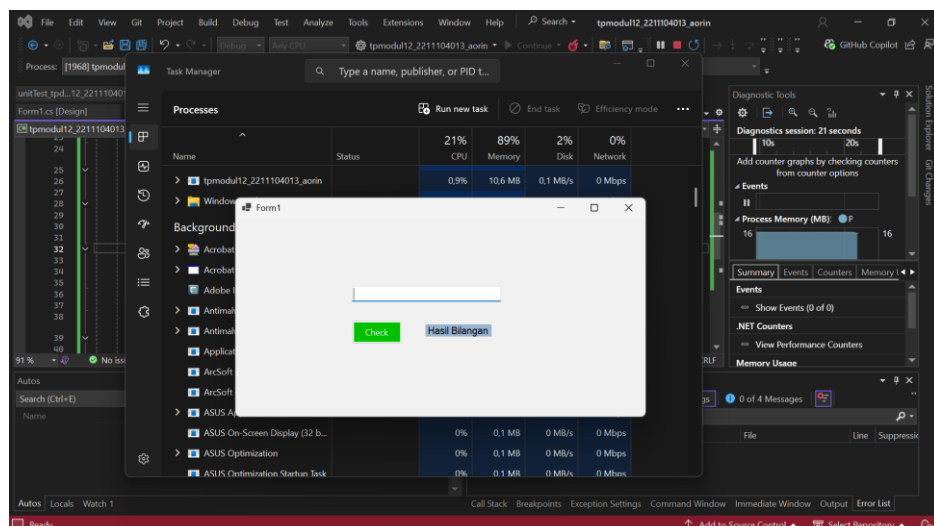
## 3. MELAKUKAN SOFTWARE PROFILING

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

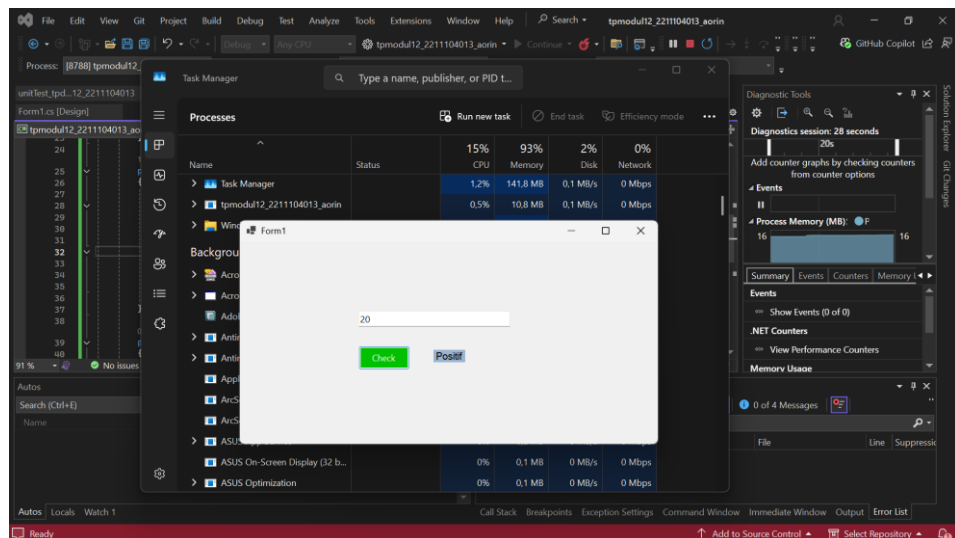
- A. Pada saat program berjalan, catat dan amati CPU usage dari aplikasi yang sedang berjalan tanpa melakukan input apapun.



- 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).

Pada saat melakukan profiling pada program di atas, ketika aplikasi berjalan tanpa melaukan input CPU Usage menunjukkan angka 0,9% sedangkan memory usage 10,6 MB.

Sementara itu, ketika dilakukan input dan meng-klik button Check CPU Usage menunjukkan angka yang stabil mulai dari 0,2 – 0,7% sedangkan Memory Usage naik menjadi 10,8 MB.

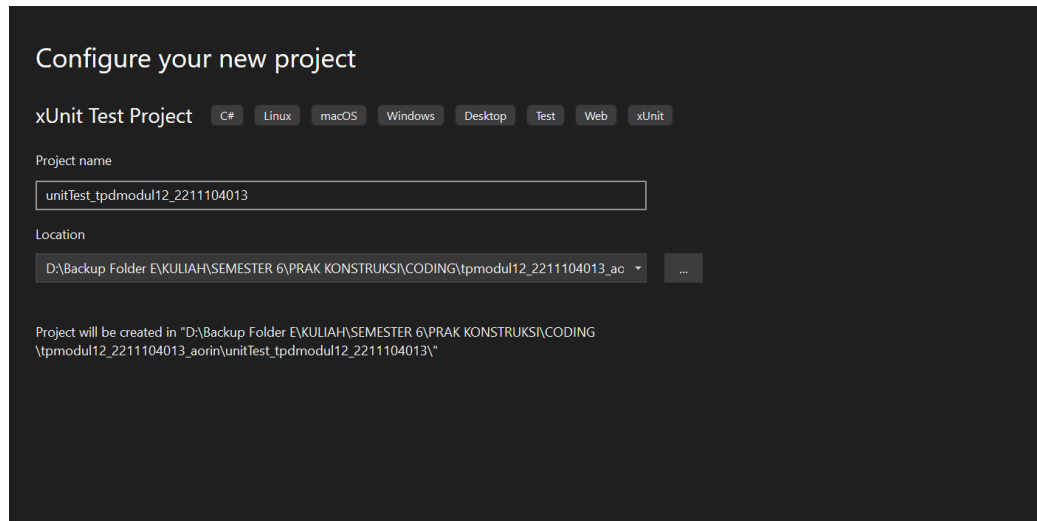
#### 4. MENAMBAHKAN UNIT TESTING

Di dalam project yang sama:

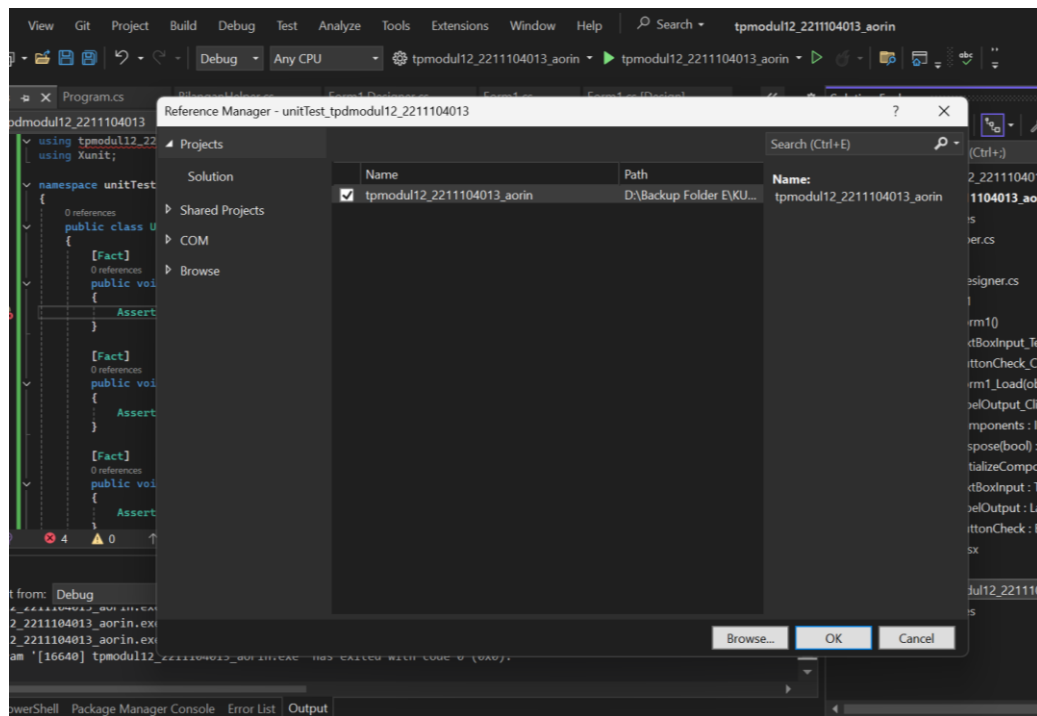
- Buatlah kode unit test untuk menguji method “CariTandaBilangan” yang dibuat sebelumnya.
- Pastikan kode unit test tersebut memiliki branch coverage yang baik untuk method “CariTandaBilangan”.
- Jalankan kode unit test yang dibuat dan lampirkan hasil unit testing yang dilakukan.

## Jawaban

### Membuat project baru untuk testing



### Membuat referensi ke project



## File UnitTest1.cs

```
1  using tpm modul12_2211104013_aorin;
2  using Xunit;
3
4  namespace unitTest_tpdmodul12_2211104013
5  {
6      0 references
7      public class UnitTest1
8      {
9          [Fact]
10         0 references
11         public void TestNegatif()
12         {
13             Assert.Equal("Negatif", BilanganHelper.CariTandaBilangan(-5));
14         }
15
16         [Fact]
17         0 references
18         public void TestPositif()
19         {
20             Assert.Equal("Positif", BilanganHelper.CariTandaBilangan(10));
21         }
22
23         [Fact]
24         0 references
25         public void TestNol()
26         {
27             Assert.Equal("Nol", BilanganHelper.CariTandaBilangan(0));
28         }
29     }
30 }
```

## Hasil testing

Test Explorer				Run   Debug	
Test run finished: 3 Tests (3 Passed, 0 Failed, 0 Skipped) run in 399 ms				0 Warnings 0 Errors	
Test	Duration	Traits	Error Message		
unitTest_tpdmodul12_2211104013 (3)	282 ms			<strong>Group Summary</strong> unitTest_tpdmodul12_2211104013 Tests in group: 3 Total Duration: 282 ms <strong>Outcomes</strong> 3 Passed	
unitTest_tpdmodul12_2211104013 (...)	282 ms				
UnitTest1 (3)	282 ms				
TestNegatif	< 1 ms				
TestNol	282 ms				
TestPositif	< 1 ms				

Berdasarkan hasil testing di atas, semua fungsi pada BilanganHelper.cs berhasil diuji dengan benar tidak ditemukan error pada skenario pengujian terhadap bilangan nol, bilangan positif, maupun negatif.