

**TUGAS JURNAL
KONSTRUKSI PERANGKAT LUNAK**

**MODUL XII
PERFORMANCE ANALYSIS UNIT TESTING & DEBUGGING**



Disusun Oleh :

Rizky Hanifa Afania

2211104017

SE-06-01

Dosen Pengampu :

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

PROGRAM STUDI S1 SOFTWARE ENGINEERING

TELKOM UNIVERSITY PURWOKERTO

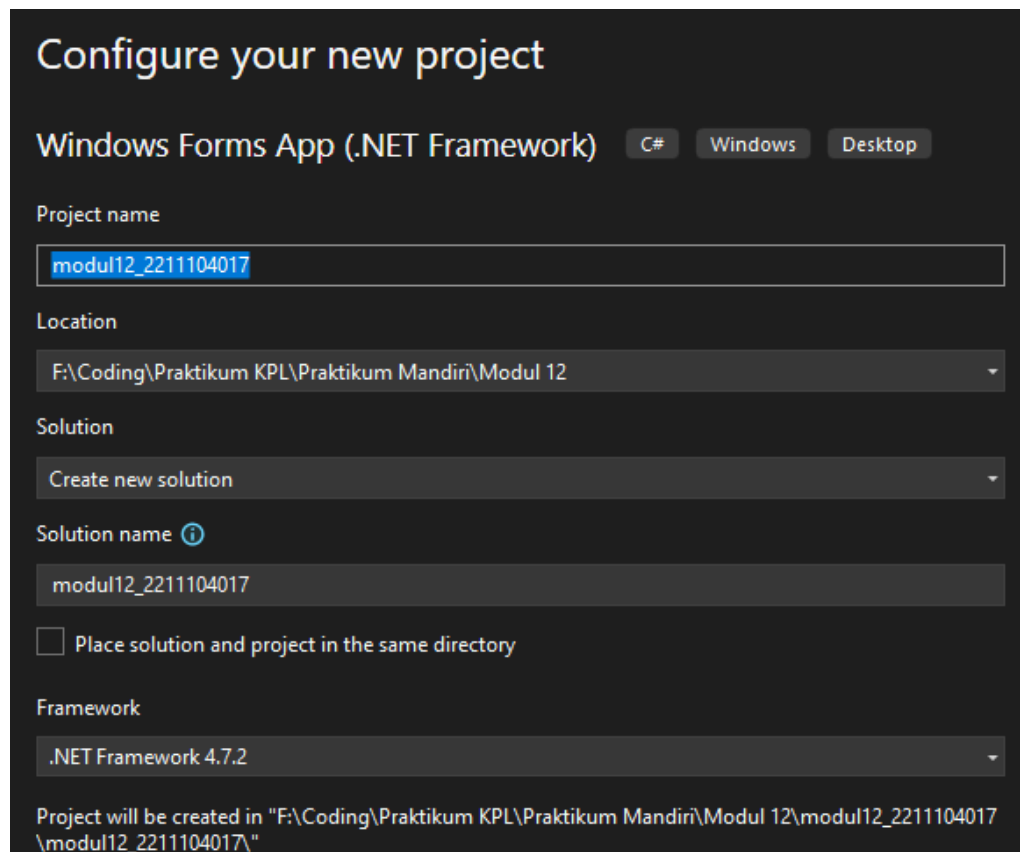
2025

TUGAS JURNAL 12

1. MEMBUAT PROJECT GUI BARU

Buka IDE misalnya dengan Visual Studio

- A. Misalnya menggunakan Visual Studio, buatlah project baru dengan nama modul12_NIM
- B. Pastikan project yang dibuat dapat menggunakan GUI (misalnya tipe Windows Form pada Visual Studio).



2. MEMBUAT GUI SEDERHANA DAN

Pada project yang telah dibuat sebelumnya:

- A. Buatlah suatu Form atau tampilan GUI sederhana dengan dua buah textbox, satu button dan satu label untuk menampilkan output.
- B. Tambahkan satu method dengan nama “CariNilaiPangkat(int a, int b)” yang menerima dua input dan mengembalikan nilai berupa hasil pangkat ab dengan melakukan iterasi (tanpa menggunakan library atau fungsi bawaan).
- C. Pada method tersebut terdapat aturan sebagai berikut (berbeda dengan aturan pangkat normal):

- Apabila input b adalah 0 maka nilai return selalu 1 (walaupun nilai a adalah 0)
- Apabila input b adalah bilangan negatif, maka nilai return adalah -1
- Apabila input b lebih dari 10 atau input a lebih dari 100 maka nilai return adalah -2
- Apabila hasil pangkat melebihi batas maksimal bilangan positif integer (misal dengan checked pada C#) maka nilai return adalah -3

D. Pada tampilan GUI, pada saat tombol ditekan, maka label output akan menampilkan hasil pangkat dari pemanggilan fungsi “CariNilaiPangkat” dari dua input textbox.

Jawab:

Source Code

a. File Form1.cs

```

1  using System;
2  using System.Windows.Forms;
3
4  namespace modul12_2211104817
5  {
6      3 references
7      public partial class Form1 : Form
8      {
9          private readonly PangkatCalculator _calculator;
10
11          1 reference
12          public Form1()
13          {
14              InitializeComponent();
15              _calculator = new PangkatCalculator(); // Menggunakan PangkatCalculator
16
17          1 reference
18          private void buttonHitung_Click(object sender, EventArgs e)
19          {
20              try
21              {
22                  int a = int.Parse(textBox1.Text);
23                  int b = int.Parse(textBox2.Text);
24
25                  int result = _calculator.CariNilaiPangkat(a, b);
26
27                  switch (result)
28                  {
29                      case -1:
30                          labelHasil.Text = "Hasil: " + result;
31                          break;
32                      case -2:
33                          labelHasil.Text = "Hasil: " + result;
34                          break;
35                      case -3:
36                          labelHasil.Text = "Hasil: " + result;
37                          break;
38                      default:
39                          labelHasil.Text = "Hasil: " + result;
40                          break;
41                  }
42              }
43              catch (FormatException)
44              {
45                  MessageBox.Show("Input tidak valid. Masukkan angka saja.");
46              }
47          }
48      }
49  }

```

b. File PangkatCalculator.cs

```
1 using System;
2
3 namespace modul12_2211104017
4 {
5     4 references
6     public class PangkatCalculator
7     {
8         7 references | 4/5 passing
9         public int CariNilaiPangkat(int a, int b)
10        {
11            // Aturan 1: Jika b == 0, hasil selalu 1 (walaupun a adalah 0)
12            if (b == 0) return 1;
13
14            // Aturan 2: Jika b < 0, hasil adalah -1
15            if (b < 0) return -1;
16
17            // Aturan 3: Jika a > 100 atau b > 10, hasil adalah -2
18            if (a > 100 || b > 10) return -2;
19
20            // Aturan 4: Hitung pangkat dengan aman (tanpa Math.Pow) dan deteksi overflow
21            try
22            {
23                checked
24                {
25                    int result = 1;
26                    for (int i = 0; i < b; i++)
27                    {
28                        result = checked(result * a);
29                    }
30                    return result;
31                }
32            }
33            catch (OverflowException)
34            {
35                return -3; // Overflow terdeteksi
36            }
37
38            return -3; // Error lain (jika ada)
39        }
40    }
41 }
```

Hasil:

- Apabila input b adalah 0 maka nilai return selalu 1 (walaupun nilai a adalah 0)

Form1

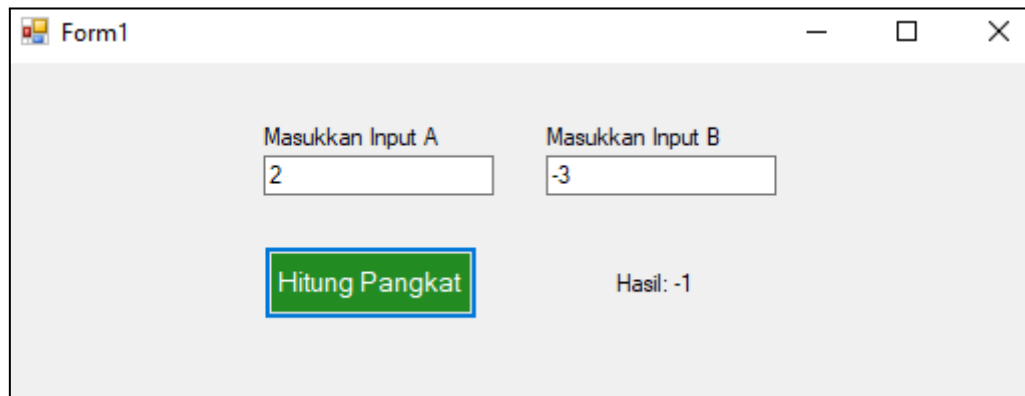
Masukkan Input A: 0

Masukkan Input B: 0

Hitung Pangkat

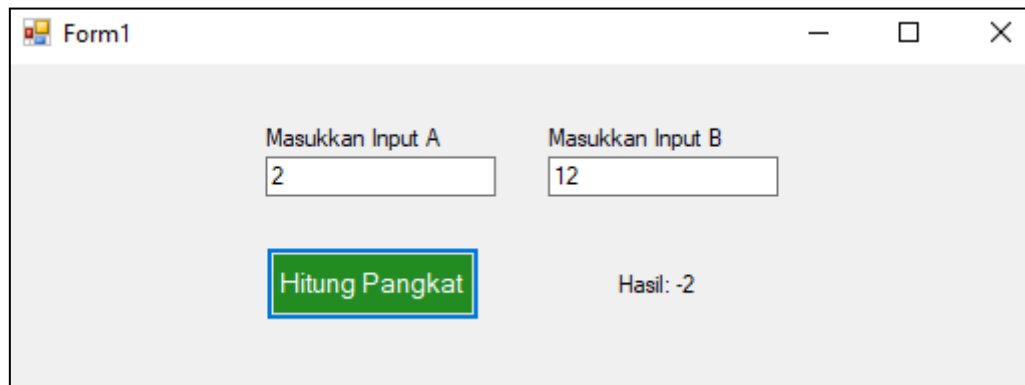
Hasil: 1

- Apabila input b adalah bilangan negatif, maka nilai return adalah -1



The screenshot shows a Windows Form titled "Form1". It contains two text boxes: "Masukkan Input A" with the value "2" and "Masukkan Input B" with the value "-3". Below the first text box is a green button labeled "Hitung Pangkat" with a blue border. To the right of the button, the text "Hasil: -1" is displayed.

- Apabila input b lebih dari 10 atau input a lebih dari 100 maka nilai return adalah -2



The screenshot shows a Windows Form titled "Form1". It contains two text boxes: "Masukkan Input A" with the value "2" and "Masukkan Input B" with the value "12". Below the first text box is a green button labeled "Hitung Pangkat" with a blue border. To the right of the button, the text "Hasil: -2" is displayed.

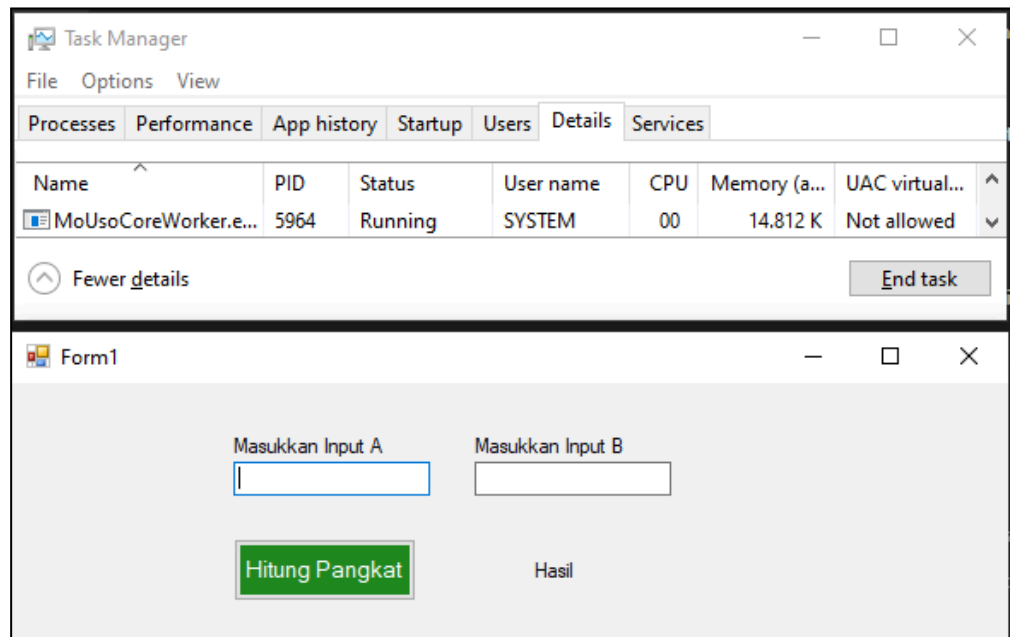
Penjelasan:

File Form1.cs adalah bagian dari aplikasi Windows Forms yang menyediakan antarmuka grafis bagi pengguna untuk menghitung pangkat suatu bilangan. Pengguna memasukkan dua angka (a dan b) melalui TextBox, lalu menekan tombol "Hitung". Aplikasi menggunakan objek PangkatCalculator untuk menghitung pangkat dan menampilkan hasil di label. Jika b adalah 0, hasilnya selalu 1. Jika b negatif, hasilnya -1. Jika $a > 100$ atau $b > 10$, hasilnya -2. Jika perhitungan menyebabkan overflow (hasil terlalu besar), hasilnya -3. Semua kondisi ini dijelaskan dalam file PangkatCalculator.cs, yang menggunakan metode iterasi (loop) dengan pengecekan overflow (checked). Jika input bukan angka, aplikasi akan menampilkan pesan kesalahan.

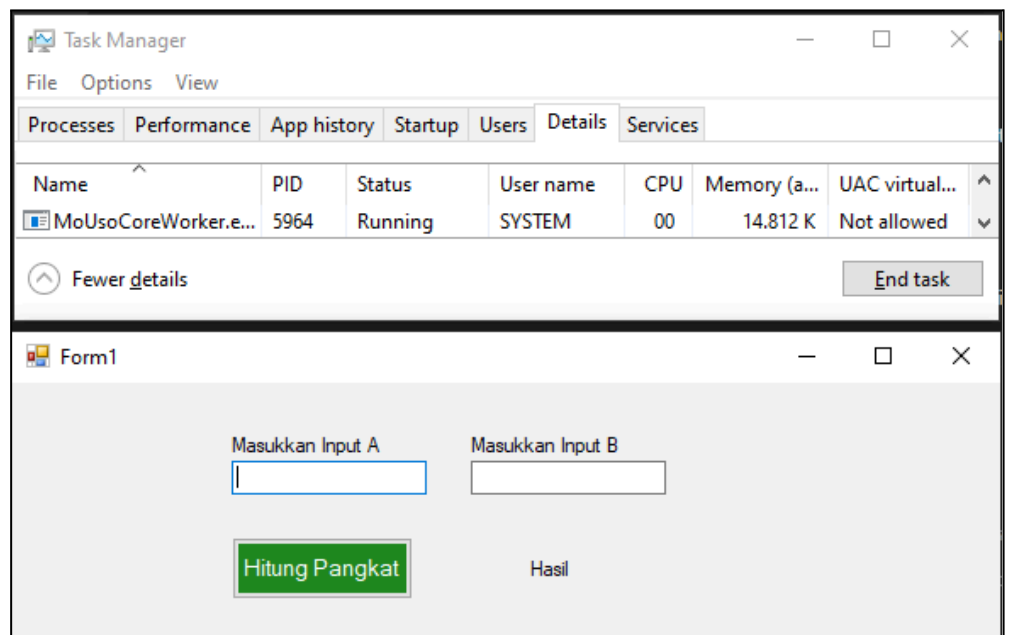
3. MELAKUKAN SOFTWARE PROFILING

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

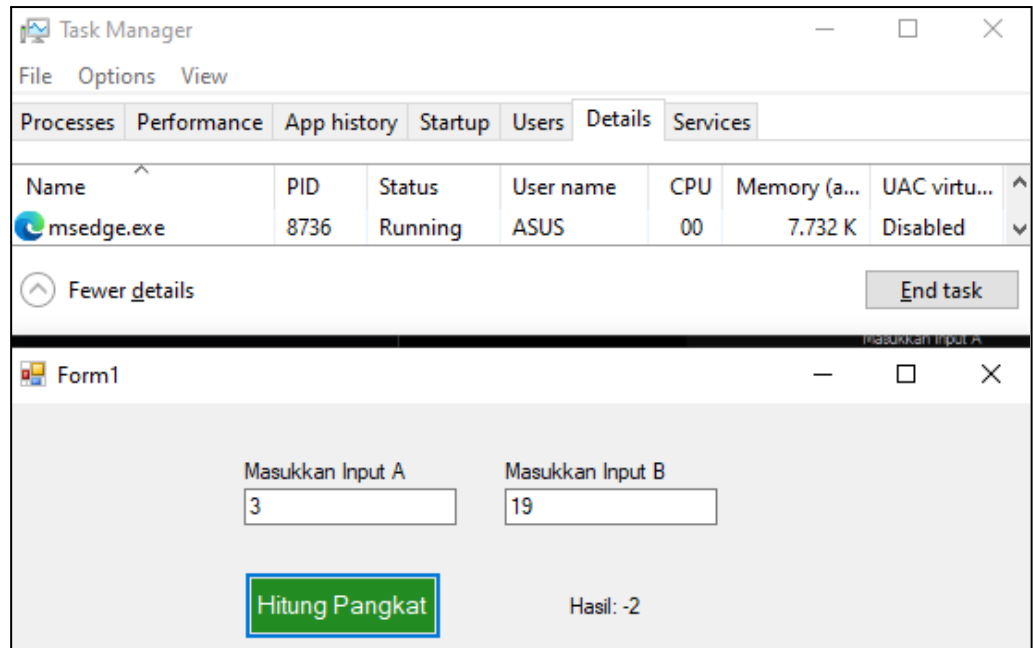
- A. Pada saat program berjalan, catat dan amati CPU usage dari aplikasi yang sedang berjalan tanpa melakukan input apapun.



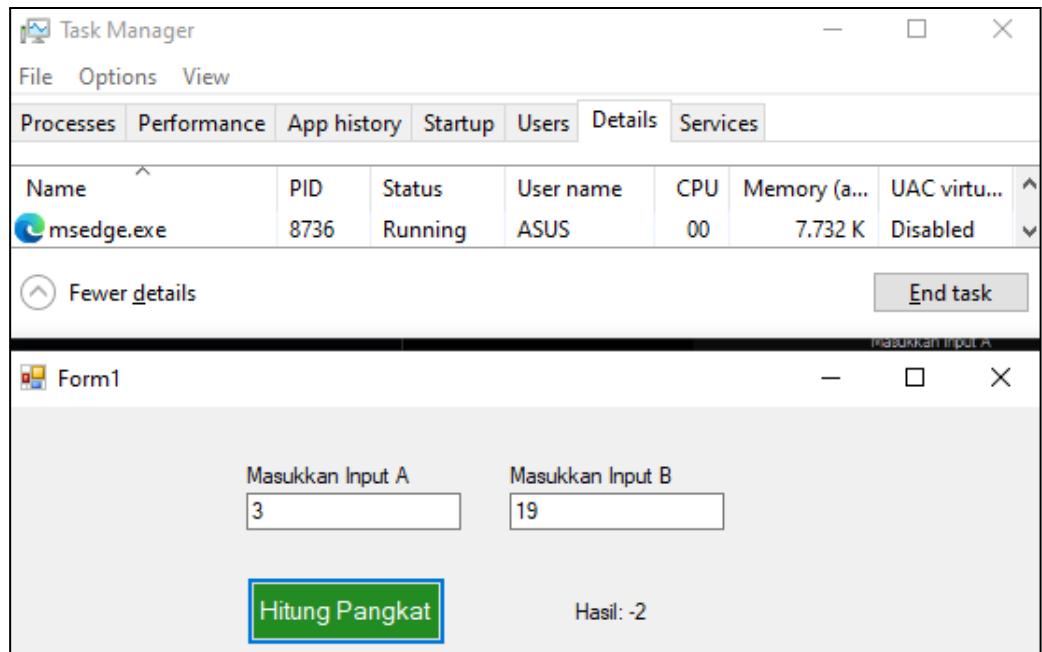
- B. Pada saat program berjalan, catat dan amati memory usage dari aplikasi yang sedang berjalan tanpa melakukan input apapun.



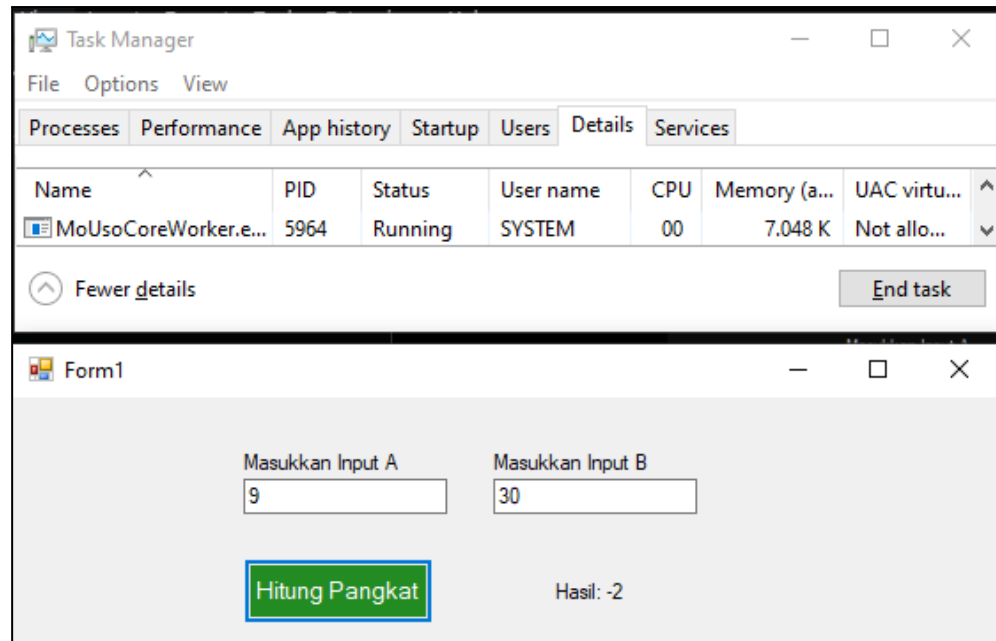
- C. Tambahkan input “3” pada textbox pertama dan “19” pada textbox ketiga, dan tekan tombol button dan catat dan amati memory usage dari aplikasi.



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



- E. Lakukan lagi eksperimen dengan input pertama yaitu “9” dan angka kedua yaitu “30”, laporkan apakah terdapat perubahan di CPU usage dan memory.



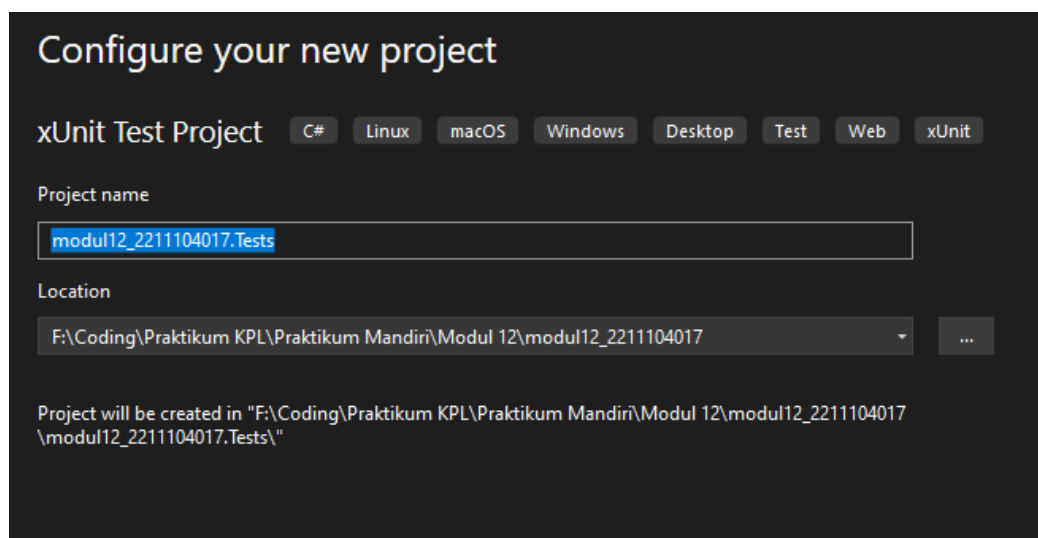
4. MENAMBAHKAN UNIT TESTING

Di dalam project yang sama:

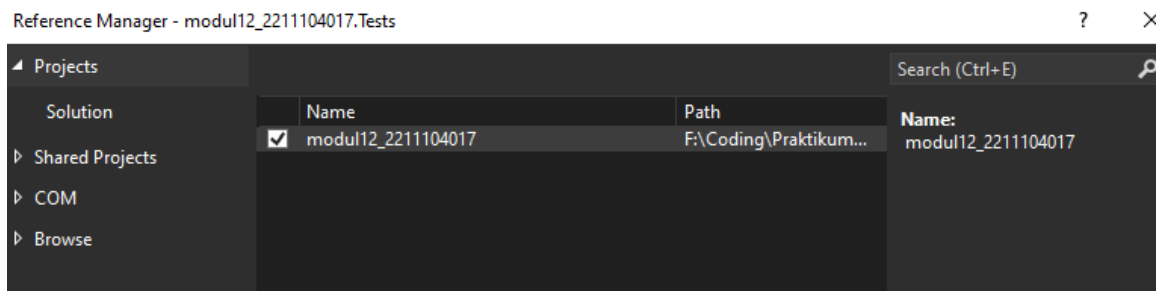
- Buatlah kode unit test untuk menguji method “CariNilaiPangkat” yang dibuat sebelumnya.
- Pastikan kode unit test tersebut memiliki branch coverage yang baik untuk method “CariNilaiPangkat”.
- Jalankan kode unit test yang dibuat dan lampirkan hasil unit testing yang dilakukan.

Jawab:

- Membuat Project untuk Testing



- Membuat referensi



- Source code file UnitTest1.cs

```

1  using modul12_2211104017;
2  using Xunit;
3
4  namespace modul12_2211104017.Tests
5  {
6      1 reference
7      public class CariNilaiPangkatTests
8      {
9          private readonly PangkatCalculator _calculator;
10
11          0 references
12          public CariNilaiPangkatTests()
13          {
14              _calculator = new PangkatCalculator();
15          }
16
17          [Fact]
18          0 references
19          public void Test_B_EqualToZero_ReturnsOne()
20          {
21              int result = _calculator.CariNilaiPangkat(0, 0);
22              Assert.Equal(1, result);
23          }
24
25          [Fact]
26          0 references
27          public void Test_B_Negative_ReturnsMinusOne()
28          {
29              int result = _calculator.CariNilaiPangkat(2, -3);
30              Assert.Equal(-1, result);
31          }
32
33          [Fact]
34          0 references
35          public void Test_A_GreaterThan100_Or_B_GreaterThan10_ReturnsMinusTwo()
36          {
37              int result1 = _calculator.CariNilaiPangkat(101, 5);
38              int result2 = _calculator.CariNilaiPangkat(5, 11);
39
40              Assert.Equal(-2, result1);
41              Assert.Equal(-2, result2);
42          }
43      }
44  }

```

```

40  public void Test_Overflow_ReturnsMinusThree()
41  {
42      int result = _calculator.CariNilaiPangkat(200, 5); // 200^5 melebihi batas int
43      Assert.Equal(-3, result);
44  }
45
46  [Fact]
47  0 references
48  public void Test_NormalPangkatCalculation()
49  {
50      int result = _calculator.CariNilaiPangkat(2, 3); // 2^3 = 8
51      Assert.Equal(8, result);
52  }
53  }

```

- Hasil testing:

| Test | Durati... | Traits | Error Message | |
|---------------------------------|-----------|--------|---------------|--------------------------|
| modul12_2211104017.Tests (4) | 475 ms | | | Run Debug |
| modul12_2211104017.Tests (4) | 475 ms | | | Group Summary |
| CariNilaiPangkatTests (4) | 475 ms | | | modul12_2211104017.Tests |
| Test_A_GreaterThan100_Or_B_G... | < 1 ms | | | Tests in group: 4 |
| Test_B_EqualToZero_ReturnsOne | < 1 ms | | | ⌚ Total Duration: 475 ms |
| Test_NormalPangkatCalculation | < 1 ms | | | Outcomes |
| Test_B_Negative_ReturnsMinus... | 475 ms | | | ✓ 4 Passed |