

LAPORAN PRAKTIKUM
JURNAL MODUL 12



Nama :

Aulia Jasifa Br Ginting 2311104060

S1SE-07-02

Dosen :

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

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK

FAKULTAS INFORMATIKA

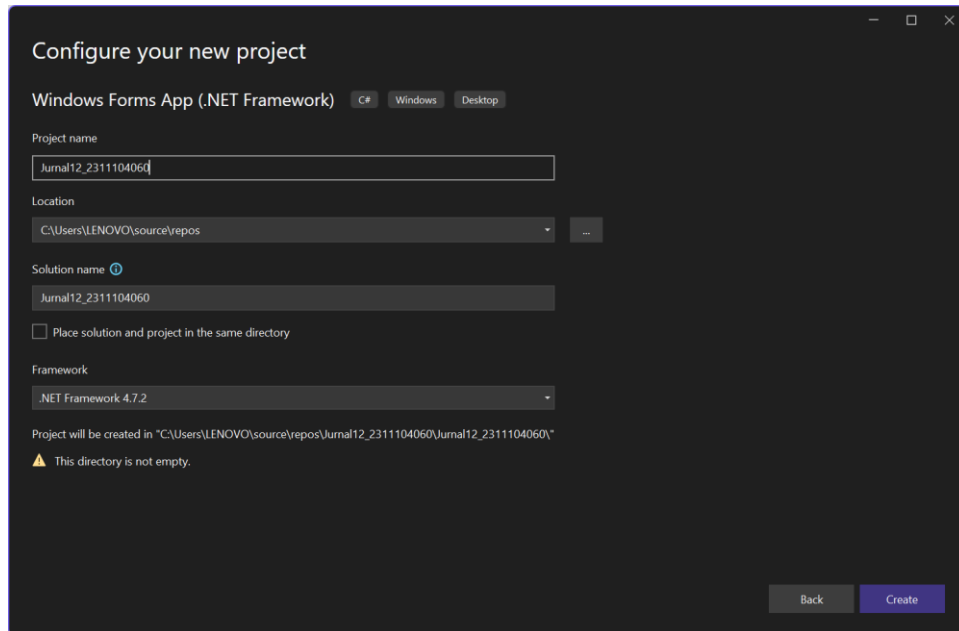
TELKOM UNIVERSITY PURWOKERTO

2025

I. LINK GITHUB

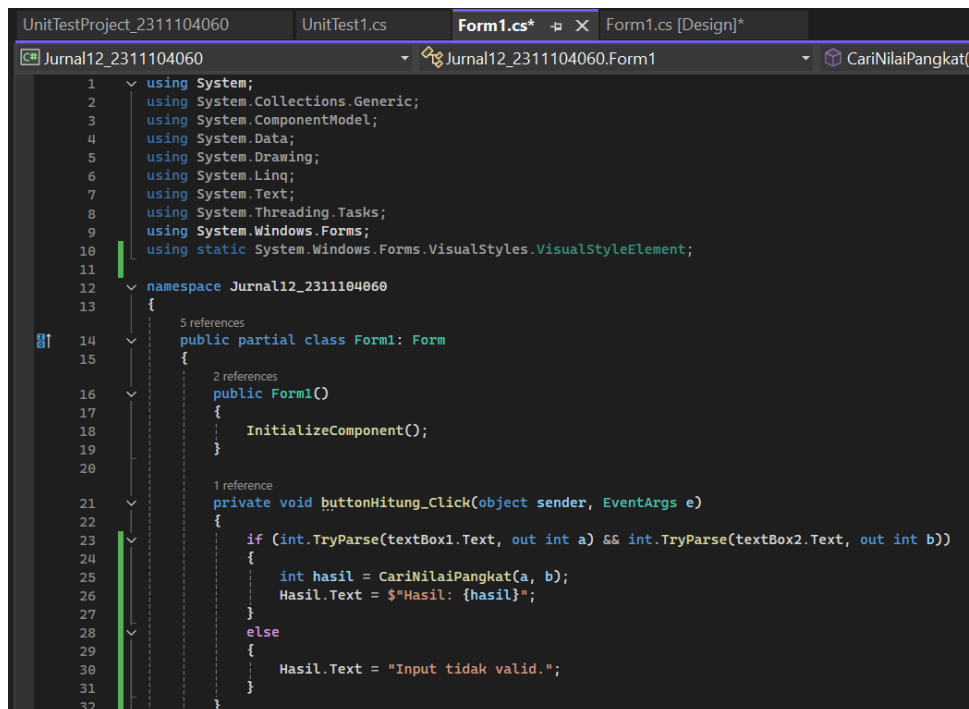
https://github.com/auliajsf06/KPL_Aulia-Jasifa-Br-Ginting_2311104060_SE-07-02

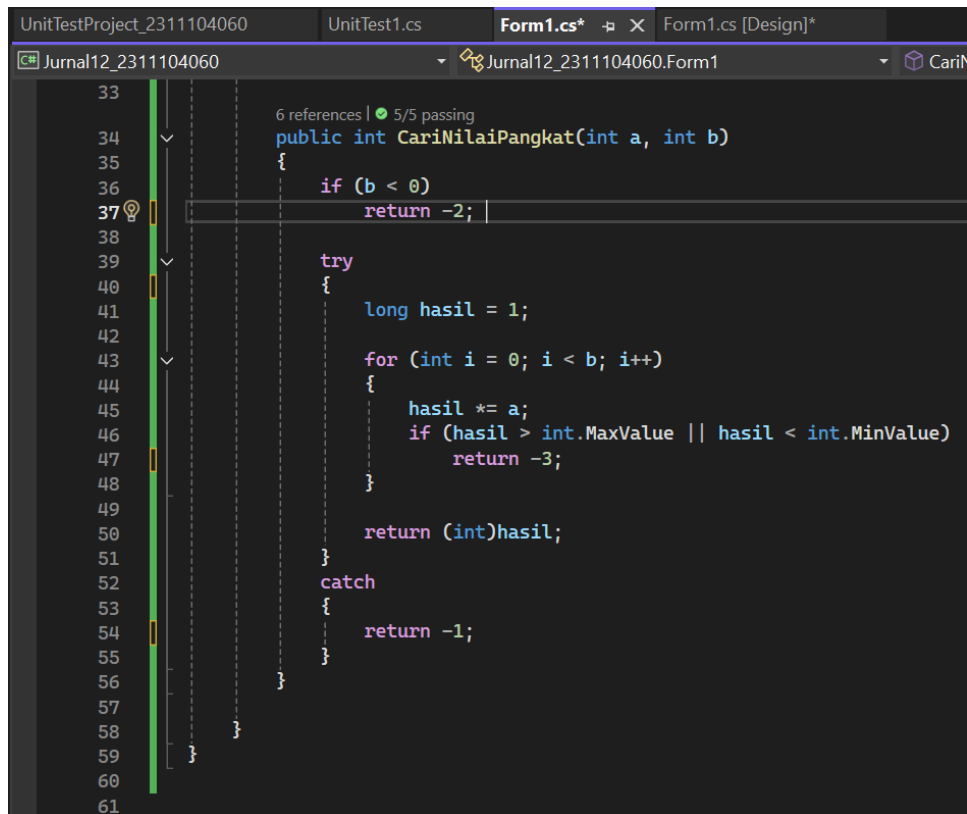
II. MEMBUAT PROJECT GUI BARU



III. MEMBUAT GUI SEDERHANA DAN

Syntax "Form1.cs"





```
UnitTestProject_2311104060  UnitTest1.cs  Form1.cs*  Form1.cs [Design]*
Jurnal12_2311104060  Jurnal12_2311104060.Form1  CariN

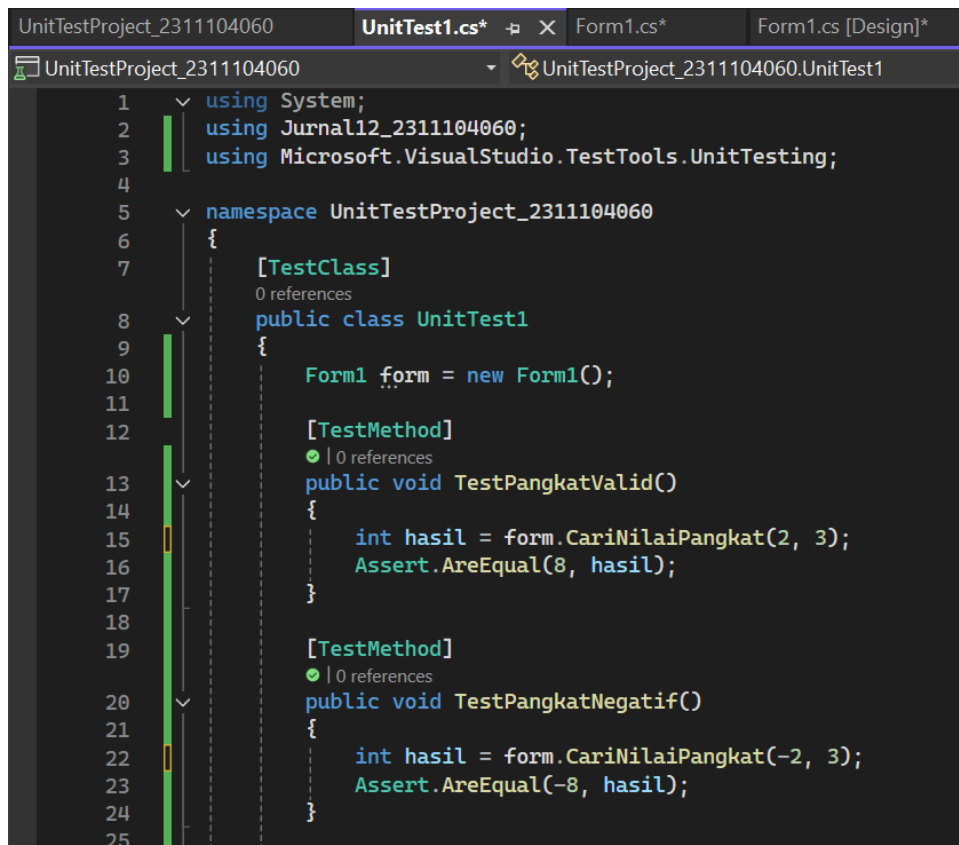
33
34
35
36
37 6 references | 5/5 passing
38  public int CariNilaiPangkat(int a, int b)
39  {
40      if (b < 0)
41      {
42          return -2;
43      }
44
45      try
46      {
47          long hasil = 1;
48
49          for (int i = 0; i < b; i++)
50          {
51              hasil *= a;
52              if (hasil > int.MaxValue || hasil < int.MinValue)
53              {
54                  return -3;
55              }
56          }
57
58          return (int)hasil;
59      }
60      catch
61      {
62          return -1;
63      }
64  }
```

Penjelasan Syntaxnya:

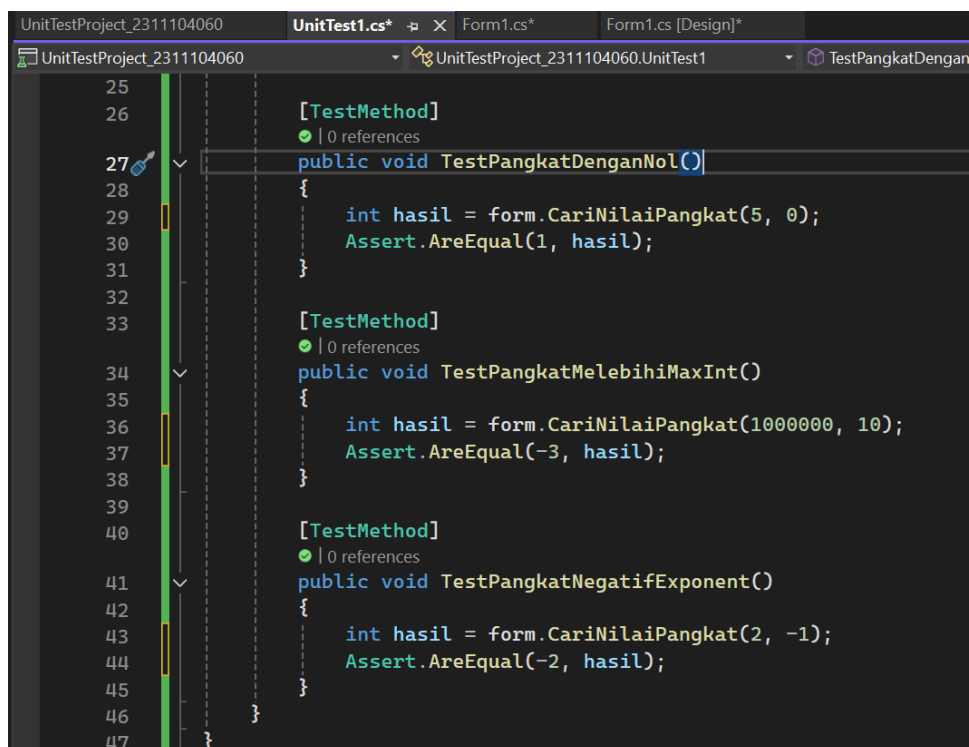
Kode di atas merupakan implementasi dari aplikasi Windows Form sederhana menggunakan bahasa C# dalam namespace Jurnal12_2311104060. Program ini memiliki satu form bernama Form1 yang menyediakan antarmuka grafis untuk menghitung hasil perpangkatan dari dua bilangan bulat yang dimasukkan pengguna. Di dalam form terdapat dua TextBox untuk input bilangan basis (a) dan eksponen (b), serta satu tombol (buttonHitung) yang saat diklik akan menjalankan fungsi CariNilaiPangkat(a, b). Fungsi ini akan menghitung hasil dari a pangkat b dengan mempertimbangkan beberapa kondisi khusus. Jika eksponen b bernilai negatif, fungsi akan mengembalikan nilai -2 sebagai indikator error. Proses perpangkatan dilakukan menggunakan tipe data long agar bisa mendeteksi kemungkinan overflow saat hasilnya melebihi batas tipe int; jika overflow terjadi, maka akan mengembalikan -3. Bila terjadi exception tak terduga, fungsi akan mengembalikan -1. Hasil akhir perhitungan ditampilkan di label Hasil. Secara keseluruhan, kode ini sudah dirancang dengan penanganan error sederhana untuk kasus input tidak valid, eksponen negatif, atau hasil yang terlalu besar.

IV. MENAMBAHKAN UNIT TESTING

Syntax “UnitTest1.cs”



```
1  using System;
2  using Jurnal12_2311104060;
3  using Microsoft.VisualStudio.TestTools.UnitTesting;
4
5  namespace UnitTestProject_2311104060
6  {
7      [TestClass]
8      0 references
9      public class UnitTest1
10     {
11         Form1 form = new Form1();
12
13         [TestMethod]
14         0 references
15         public void TestPangkatValid()
16         {
17             int hasil = form.CariNilaiPangkat(2, 3);
18             Assert.AreEqual(8, hasil);
19         }
20
21         [TestMethod]
22         0 references
23         public void TestPangkatNegatif()
24         {
25             int hasil = form.CariNilaiPangkat(-2, 3);
26             Assert.AreEqual(-8, hasil);
27         }
28     }
29 }
```



```
25
26
27     [TestMethod]
28     0 references
29     public void TestPangkatDenganNol()
30     {
31         int hasil = form.CariNilaiPangkat(5, 0);
32         Assert.AreEqual(1, hasil);
33     }
34
35     [TestMethod]
36     0 references
37     public void TestPangkatMelebihiMaxInt()
38     {
39         int hasil = form.CariNilaiPangkat(1000000, 10);
40         Assert.AreEqual(-3, hasil);
41     }
42
43     [TestMethod]
44     0 references
45     public void TestPangkatNegatifExponent()
46     {
47         int hasil = form.CariNilaiPangkat(2, -1);
48         Assert.AreEqual(-2, hasil);
49     }
50 }
```

Penjelasan Syntaxnya:

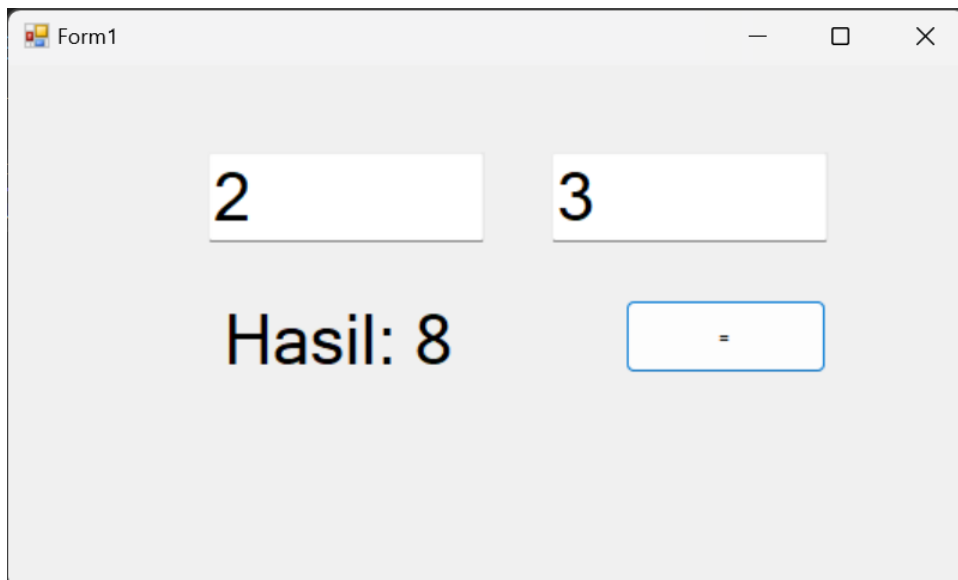
Pengujian unit (unit testing) menggunakan framework Microsoft.VisualStudio.TestTools.UnitTesting dalam C#. Tujuan dari kode ini adalah untuk menguji fungsionalitas dari metode CariNilaiPangkat yang ada dalam kelas

Form1. Kode dimulai dengan mendeklarasikan sebuah kelas unit test bernama UnitTest1, di mana setiap metode dalam kelas ini diberi atribut [TestMethod] untuk menandakan bahwa metode tersebut merupakan bagian dari pengujian.

Pada kelas ini, terdapat beberapa metode pengujian yang masing-masing menguji berbagai kondisi pemanggilan metode CariNilaiPangkat. Metode pertama, TestPangkatValid, menguji apakah hasil pemangkatan dengan basis positif, seperti CariNilaiPangkat(2, 3), menghasilkan 8, yang merupakan hasil yang diharapkan. TestPangkatNegatif menguji kasus dengan basis negatif, yaitu CariNilaiPangkat(-2, 3), yang seharusnya menghasilkan -8. Sementara itu, TestPangkatDenganNol menguji kasus di mana eksponen bernilai 0, yang sesuai dengan aturan matematika seharusnya menghasilkan 1. TestPangkatMelebihiMaxInt menguji bagaimana program menangani kasus di mana hasil pemangkatan melebihi batas maksimum tipe data integer, dengan asumsi bahwa hasilnya adalah nilai yang tidak valid atau overflow, dan dalam hal ini diharapkan menghasilkan -3. Terakhir, TestPangkatNegatifExponent menguji apakah eksponen negatif seperti pada CariNilaiPangkat(2, -1) menghasilkan hasil yang sesuai, yaitu -2, yang mungkin merupakan implementasi khusus untuk eksponen negatif dalam kode.

Setiap pengujian menggunakan Assert.AreEqual(expected, actual) untuk memeriksa apakah hasil yang diperoleh sesuai dengan hasil yang diharapkan. Jika hasilnya tidak sesuai, tes akan gagal. Secara keseluruhan, kode ini memastikan bahwa metode CariNilaiPangkat diuji secara menyeluruh dalam berbagai kondisi yang mungkin terjadi, untuk memastikan metode tersebut berfungsi dengan benar.

V. OUTPUT YANG DI HASILKAN



The screenshot shows a simple graphical user interface for a calculator application. It features two input boxes at the top for entering numbers, with the first box containing '2' and the second box containing '3'. Below the input boxes, the result 'Hasil: 8' is displayed. To the right of the result, there is a button with an equals sign '='.

Form1

7 4

Hasil: 2401 =

Form1

5 3

Hasil: 125 =

RUN ALL TEST

Test Explorer

Test run finished: 5 Tests (5 Passed, 0 Failed, 0 Skipped) run in 763 ms

Test	Duration	Traits	Error Message
✔ UnitTestProject_2311104060 (5)	327 ms		
✔ UnitTestProject_2311104060 (5)	327 ms		
✔ UnitTest1 (5)	327 ms		
✔ TestPangkatDenganNol	327 ms		
✔ TestPangkatMelebihiMaxInt	< 1 ms		
✔ TestPangkatNegatif	< 1 ms		
✔ TestPangkatNegatifExponent	< 1 ms		
✔ TestPangkatValid	< 1 ms		

1 Warning 0 Errors

Test Detail Summary

✔ TestPangkatValid

Source: UnitTest1

Duration: < 1 ms

VI. KESIMPULAN

Pada praktikum kali ini mempelajari aplikasi Windows Forms ini memungkinkan pengguna untuk menghitung nilai pangkat dari dua angka yang dimasukkan melalui antarmuka grafis. Kode ini menangani beberapa skenario, seperti validasi input untuk memastikan angka yang dimasukkan benar, pengecekan overflow untuk mencegah hasil yang melebihi batas tipe data integer, dan penanganan eksponen negatif yang mengembalikan nilai khusus jika eksponen lebih kecil dari nol. Jika perhitungan berhasil, hasil pangkat ditampilkan; jika tidak, aplikasi memberikan pesan kesalahan yang sesuai. Dengan demikian, aplikasi ini dirancang untuk menghitung pangkat dengan aman dan memberi respons yang jelas terhadap input yang salah atau kondisi yang tidak terduga.