

Nama : Naufal Aflakh Wijayanto

Kelas : SE063

NIM : 2211104073

Program.cs

```
using System;
using System.Windows.Forms;

namespace Modul12App
{
    public class MainForm : Form
    {
        private TextBox textBoxA;
        private TextBox textBoxB;
        private Button button;
        private Label label;

        public MainForm()
        {
            // Initialize components
            textBoxA = new TextBox() { Location = new System.Drawing.Point(20, 20) };
            textBoxB = new TextBox() { Location = new System.Drawing.Point(20, 60) };
            button = new Button() { Text = "Hitung", Location = new System.Drawing.Point(20, 100) };
        };

        label = new Label() { Location = new System.Drawing.Point(20, 140), Width = 200 };

        button.Click += new EventHandler(Button_Click);

        Controls.Add(textBoxA);
        Controls.Add(textBoxB);
        Controls.Add(button);
        Controls.Add(label);
    }

    private void Button_Click(object sender, EventArgs e)
    {
        int a = Convert.ToInt32(textBoxA.Text);
        int b = Convert.ToInt32(textBoxB.Text);
        int result = CariNilaiPangkat(a, b);
        label.Text = "Hasil Pangkat: " + result.ToString();
    }
}
```

```

public int CariNilaiPangkat(int a, int b)
{
    // Aturan jika b == 0
    if (b == 0) return 1;

    // Aturan jika b negatif
    if (b < 0) return -1;

    // Aturan jika a > 100 atau b > 10
    // Dalam hal ini, overflow mungkin terjadi
    if (a > 100 || b > 10) return -2;

    int result = 1;
    for (int i = 0; i < b; i++)
    {
        // Mengecek apakah perkalian berikutnya menyebabkan overflow
        if (result > int.MaxValue / a)
        {
            return -3; // Overflow
        }

        result *= a;
    }

    return result;
}

[STAThread]
public static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new MainForm());
}
}

```

Program di atas adalah aplikasi Windows Forms sederhana dalam bahasa C# yang menghitung nilai pemangkatan (pangkat) dari dua bilangan bulat yang dimasukkan oleh pengguna. Pengguna memasukkan nilai basis (a) dan eksponen (b) ke dalam dua TextBox, lalu menekan tombol "Hitung" untuk mendapatkan hasilnya. Proses perhitungan dilakukan oleh fungsi CariNilaiPangkat(int a, int b), yang dilengkapi dengan berbagai pengecekan kondisi,

seperti: jika eksponen nol akan mengembalikan 1, jika eksponen negatif akan mengembalikan -1, jika nilai terlalu besar ($a > 100$ atau $b > 10$) akan mengembalikan -2, dan jika terjadi overflow saat perkalian akan mengembalikan -3. Hasil dari perhitungan ditampilkan di sebuah Label. Program juga menggunakan metode `int.MaxValue` untuk mencegah overflow saat melakukan perkalian berulang.

Test1.cs

```
using Microsoft.VisualStudio.TestTools.UnitTesting;
using Modul12App; // Mengimpor namespace Modul12App untuk mengakses MainForm

namespace Modul12App.Tests
{
    [TestClass]
    public class MainFormTests
    {
        // Test untuk kasus  $a^b$  dengan bilangan positif
        [TestMethod]
        [DataRow(2, 3, 8)] //  $2^3 = 8$ 
        [DataRow(3, 2, 9)] //  $3^2 = 9$ 
        public void TestCariNilaiPangkat_Positive(int a, int b, int expected)
        {
            var form = new MainForm();
            var result = form.CariNilaiPangkat(a, b);
            Assert.AreEqual(expected, result);
        }

        // Test untuk kasus  $b = 0$ 
        [TestMethod]
        [DataRow(5, 0, 1)] //  $5^0 = 1$ 
        public void TestCariNilaiPangkat_BerapaBawah(int a, int b, int expected)
        {
            var form = new MainForm();
            var result = form.CariNilaiPangkat(a, b);
            Assert.AreEqual(expected, result);
        }

        // Test untuk kasus b negatif
        [TestMethod]
        [DataRow(4, -2, -1)] // b negatif menghasilkan -1
        public void TestCariNilaiPangkat_BNegatif(int a, int b, int expected)
        {
            var form = new MainForm();
            var result = form.CariNilaiPangkat(a, b);
        }
    }
}
```

```

        Assert.AreEqual(expected, result);
    }

    // Test untuk kasus a > 100 atau b > 10
    [TestMethod]
    [DataRow(150, 5, -2)] // a > 100 menghasilkan -2
    [DataRow(3, 12, -2)] // b > 10 menghasilkan -2
    public void TestCariNilaiPangkat_A_Atau_BLebihBesar(int a, int b, int expected)
    {
        var form = new MainForm();
        var result = form.CariNilaiPangkat(a, b);
        Assert.AreEqual(expected, result);
    }

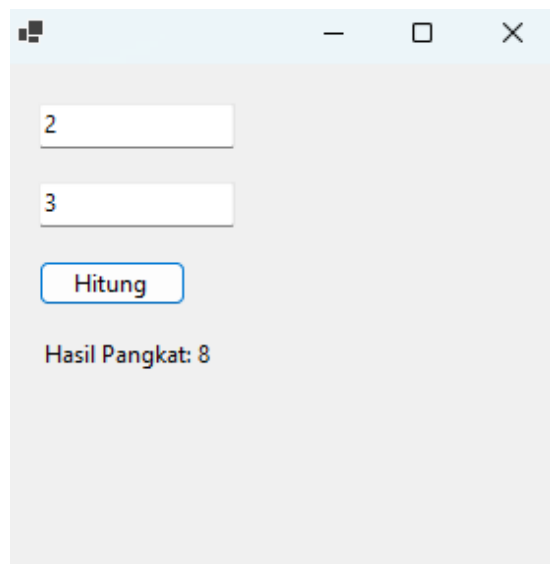
    // Test untuk overflow
    [TestMethod]
    [DataRow(1000, 100, -3)] // Terjadi overflow
    public void TestCariNilaiPangkat_Overflow(int a, int b, int expected)
    {
        var form = new MainForm();
        var result = form.CariNilaiPangkat(a, b);
        Assert.AreEqual(expected, result);
    }
}

```

Kode di atas adalah unit test menggunakan *MSTest* untuk menguji fungsi `CariNilaiPangkat` dari kelas `MainForm` dalam aplikasi `Modul12App`. Terdapat lima metode pengujian (`TestMethod`) yang mencakup berbagai skenario penting: pemangkatan bilangan positif (`TestCariNilaiPangkat_Positive`), kasus eksponen nol (`TestCariNilaiPangkat_BerapaBawah`), eksponen negatif (`TestCariNilaiPangkat_BNegatif`), nilai `a` atau `b` melebihi batas yang ditentukan (`TestCariNilaiPangkat_A_Atau_BLebihBesar`), dan kasus overflow (`TestCariNilaiPangkat_Overflow`). Masing-masing metode menggunakan atribut `[DataRow]` untuk memberikan data uji secara langsung dan memastikan hasilnya sesuai dengan yang diharapkan menggunakan `Assert.AreEqual`. Dengan pengujian ini, fungsionalitas dan validasi batas pada metode `CariNilaiPangkat` dapat dipastikan bekerja dengan benar dan aman.

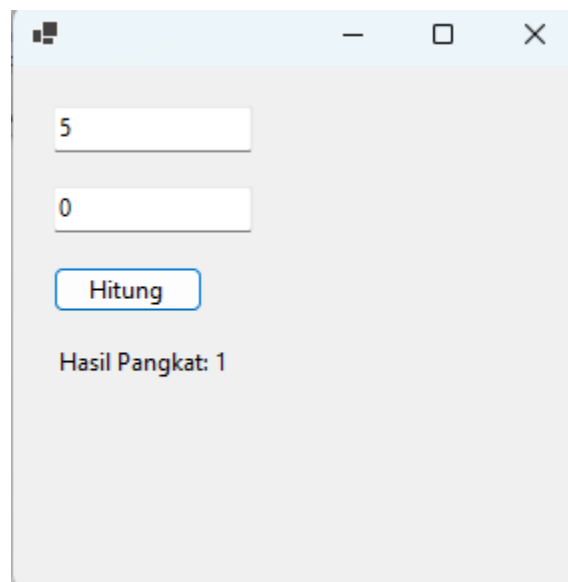
Output

Input dengan Bilangan Positif



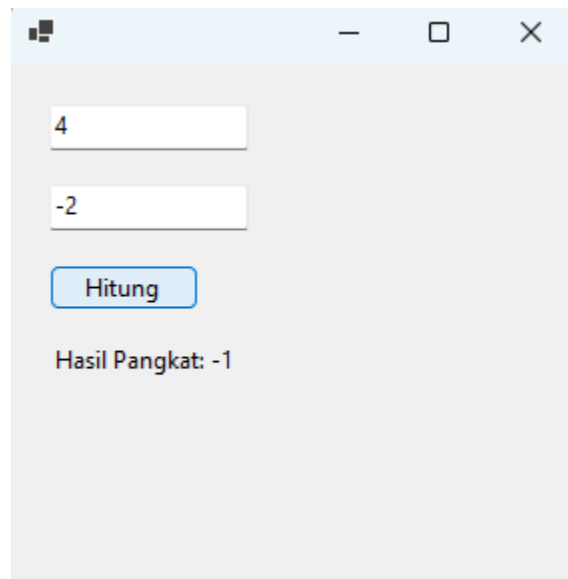
A screenshot of a Java Swing window with a light blue title bar and standard window controls (minimize, maximize, close). The window contains two text input fields. The first field contains the number '2' and the second field contains the number '3'. Below the input fields is a button labeled 'Hitung' with a blue border. Below the button, the text 'Hasil Pangkat: 8' is displayed.

Input dengan $b = 0$



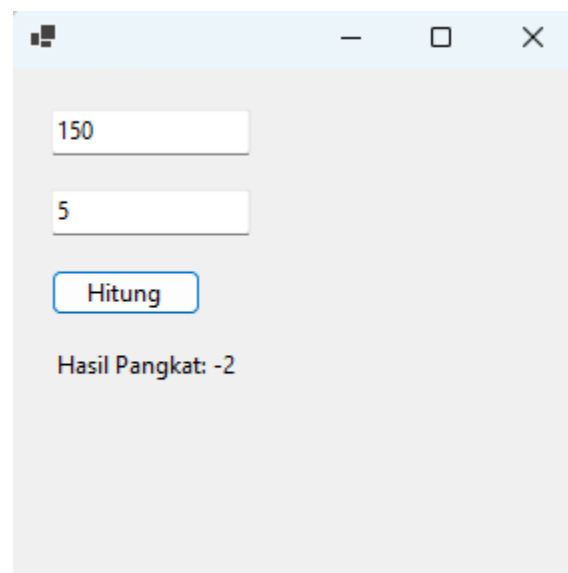
A screenshot of a Java Swing window with a light blue title bar and standard window controls (minimize, maximize, close). The window contains two text input fields. The first field contains the number '5' and the second field contains the number '0'. Below the input fields is a button labeled 'Hitung' with a blue border. Below the button, the text 'Hasil Pangkat: 1' is displayed.

Input dengan b Negatif



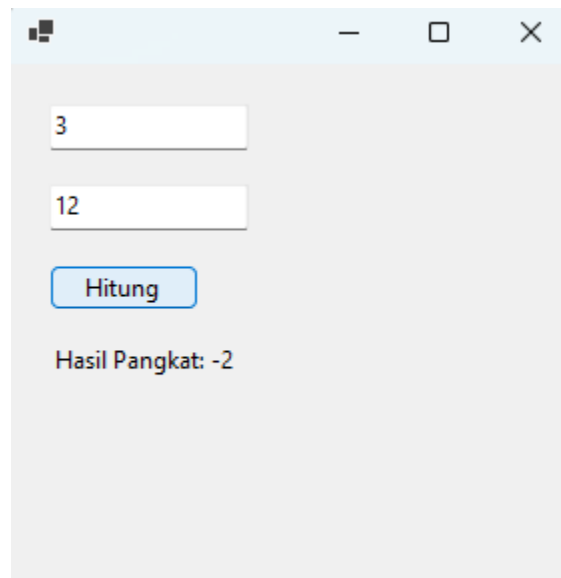
A screenshot of a Java Swing window titled "Hitung Pangkat". The window has a light blue title bar with standard Windows window controls (minimize, maximize, close). The main content area is light gray. It contains two text input fields: the first contains the number "4" and the second contains "-2". Below the input fields is a blue button with the text "Hitung". At the bottom of the window, the text "Hasil Pangkat: -1" is displayed.

Input dengan a > 100



A screenshot of a Java Swing window titled "Hitung Pangkat". The window has a light blue title bar with standard Windows window controls (minimize, maximize, close). The main content area is light gray. It contains two text input fields: the first contains the number "150" and the second contains "5". Below the input fields is a blue button with the text "Hitung". At the bottom of the window, the text "Hasil Pangkat: -2" is displayed.

Input dengan $b > 10$



A screenshot of a Java Swing window titled "Hitung Pangkat". The window has a light blue title bar with standard minimize, maximize, and close buttons. Inside the window, there are two text input fields. The first field contains the number "3" and the second field contains the number "12". Below these fields is a blue button with the text "Hitung". Underneath the button, the text "Hasil Pangkat: -2" is displayed. The window's background is a light gray.

Menambahkan Unit Test

