

Muhammad Ikhsan Al Hakim

2311104064

## Laporan Tugas Pendahuluan Modul 12

### 1. From1.cs

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

namespace modul12_2311104064
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        public int CariNilaiPangkat(int a, int b)
        {
            if (b == 0)
            {
                return 1;
            }

            if (b < 0)
            {
                return -1;
            }

            if (b > 10 || a > 100)
            {
                return -2;
            }

            long hasilPangkat = 1;
            try
            {
                checked
                {
                    for (int i = 0; i < b; i++)
                    {
                        hasilPangkat = hasilPangkat * a;

                        if (hasilPangkat > int.MaxValue)
                        {
                            return -3;
                        }
                    }
                }
            }
            catch (OverflowException)
            {
                return -3;
            }

            if (hasilPangkat > int.MaxValue || hasilPangkat <
int.MinValue)
            {
                return -3;
            }
        }
    }
}
```

```

    }

    return (int)hasilPangkat;
}

private void buttonHitung_Click(object sender, EventArgs
e)
{
    try
    {
        int nilaiA = Convert.ToInt32(textBoxA.Text);
        int nilaiB = Convert.ToInt32(textBoxB.Text);
        int hasil = CariNilaiPangkat(nilaiA, nilaiB);
        labelOutput.Text = "Hasil: " + hasil.ToString();
    }
    catch (FormatException)
    {
        labelOutput.Text = "Error: Input a dan b harus
berupa angka yang valid!";
    }
    catch (OverflowException)
    {
        labelOutput.Text = "Error: Angka input terlalu
besar atau kecil!";
    }
    catch (Exception ex)
    {
        labelOutput.Text = "Error tidak diketahui: " +
ex.Message;
    }
}
}
}

```

Penjelasan:

Kode C# ini adalah untuk aplikasi Windows Forms yang menghitung nilai pangkat ab dengan aturan khusus.

### Penjelasan Kode

Kode ini mendefinisikan sebuah form (Form1) dalam *namespace* modul12\_2311104064. Form ini memiliki logika untuk mengambil dua input angka dari pengguna, menghitung pangkatnya berdasarkan aturan tertentu, dan menampilkan hasilnya atau pesan error.

### Konstruktor Form1()

- `public Form1()`: Ini adalah metode yang pertama kali dijalankan saat form dibuat.
- `InitializeComponent();`: Baris ini sangat penting. Ia memanggil metode lain (yang dibuat otomatis oleh Visual Studio di file `Form1.Designer.cs`) untuk membuat dan mengatur semua komponen visual yang sudah Anda desain (seperti `TextBox`, `Button`, dan `Label`).

### Metode CariNilaiPangkat(int a, int b)

Fungsi ini adalah inti dari logika perhitungan pangkat dengan aturan-aturan berikut:

1. **Jika b (pangkat) adalah 0:** Fungsi langsung mengembalikan 1 (aturan i).
2. **Jika b adalah bilangan negatif:** Fungsi mengembalikan -1 (aturan ii).
3. **Jika b lebih dari 10 ATAU a (basis) lebih dari 100:** Fungsi mengembalikan -2 (aturan iii).
4. **Perhitungan Pangkat dengan Iterasi:**
  - Variabel hasilPangkat diinisialisasi sebagai 1 dan bertipe long untuk bisa menampung angka yang lebih besar sementara, guna mendeteksi *overflow* (melebihi batas int).
  - Dilakukan perulangan (for) sebanyak b kali. Dalam setiap iterasi, hasilPangkat dikalikan dengan a.
  - **Deteksi Overflow (Aturan iv):**
    - Blok checked { ... } digunakan agar operasi perkalian `hasilPangkat = hasilPangkat * a;` melempar `OverflowException` jika hasilnya melebihi batas tipe data long (meskipun di sini kita lebih fokus pada batas int).
    - Setelah perkalian, ada pengecekan if (`hasilPangkat > int.MaxValue`). Jika hasil sementara sudah melebihi batas maksimal int, fungsi mengembalikan -3.
    - Jika `OverflowException` tertangkap selama perkalian (misalnya jika `hasilPangkat * a` sudah melebihi batas long), fungsi juga mengembalikan -3.
5. **Pengecekan Overflow Akhir:** Setelah loop, ada pengecekan tambahan apakah hasilPangkat (yang bertipe long) berada di luar jangkauan int. Jika ya, kembalikan -3.
6. **Hasil Akhir:** Jika semua aturan dan pengecekan lolos, hasilPangkat (yang sudah dipastikan muat di int) dikonversi ke int dan dikembalikan.

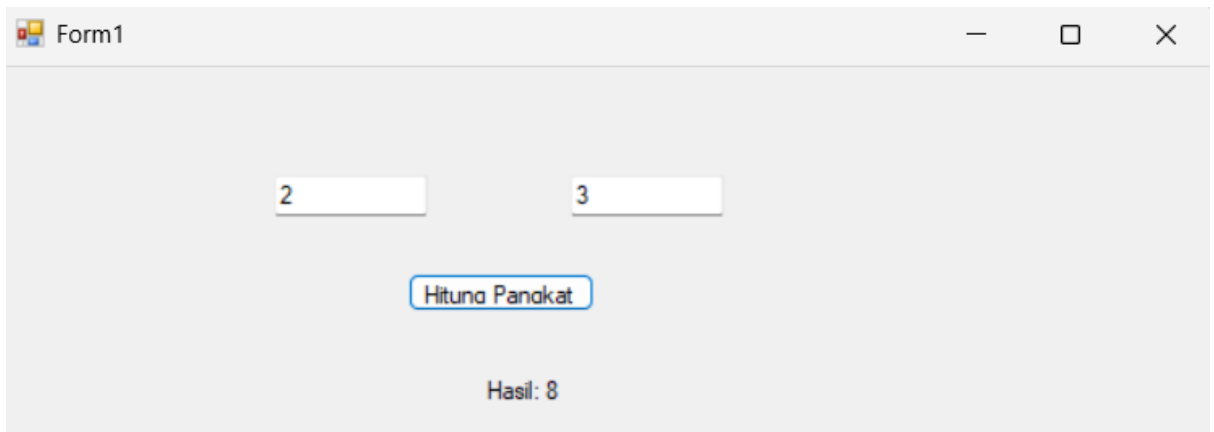
### Metode buttonHitung\_Click(object sender, EventArgs e)

Metode ini dijalankan ketika tombol yang terhubung (kemungkinan bernama buttonHitung) diklik.

1. **Blok try:** Mencoba menjalankan kode yang berpotensi menimbulkan error.
  - Mengambil teks dari `textBoxA` dan `textBoxB` (diasumsikan ini adalah nama `TextBox` untuk input a dan b).
  - Mengonversi teks tersebut menjadi bilangan bulat (int) menggunakan `Convert.ToInt32()`.
  - Memanggil `CariNilaiPangkat()` dengan kedua angka tersebut untuk mendapatkan hasil.
  - Menampilkan hasil perhitungan ke `labelOutput` (diasumsikan ini nama `Label` untuk output) dengan format "Hasil: [hasil]".
2. **Blok catch:** Menangani error yang mungkin terjadi di dalam blok try.

- `catch (FormatException)`: Jika input dari `TextBox` tidak bisa diubah menjadi angka (misalnya pengguna memasukkan huruf), `labelOutput` akan menampilkan pesan error format.
- `catch (OverflowException)`: Jika angka yang dimasukkan pengguna saat konversi terlalu besar atau kecil untuk tipe `int`, `labelOutput` akan menampilkan pesan error overflow.
- `catch (Exception ex)`: Menangkap semua jenis error lain yang mungkin tidak terduga dan menampilkan pesan errornya.

## 2. Hasil Run



Form1

2 3

Hituna Panakat

Hasil: 8



Form1

4 6

Hituna Panakat

Hasil: 4096