

Nama : Dimastian Aji Wibowo

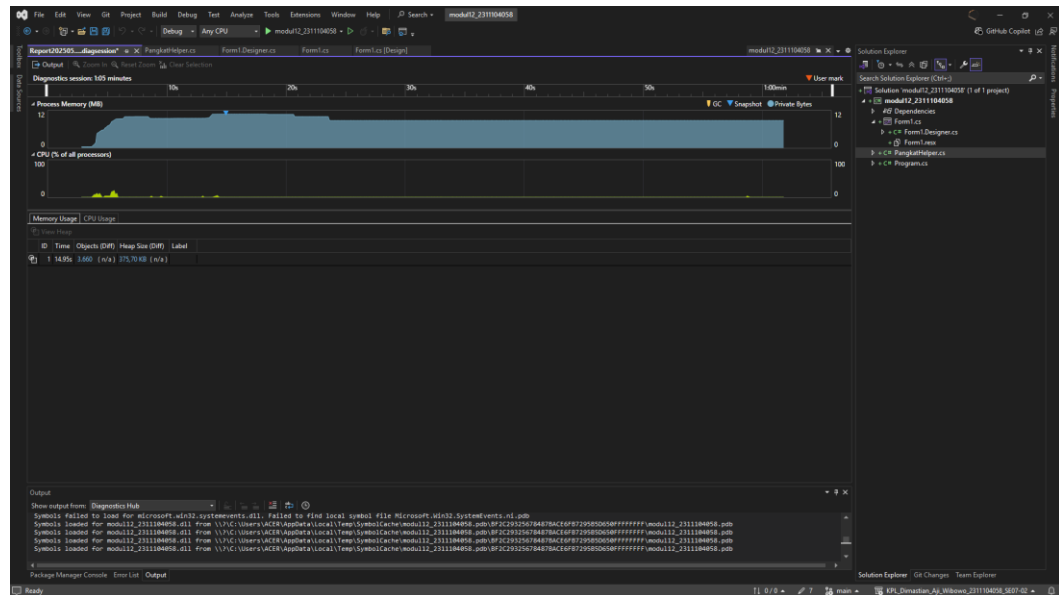
NIM : 2311104058

JURNAL MODUL 12

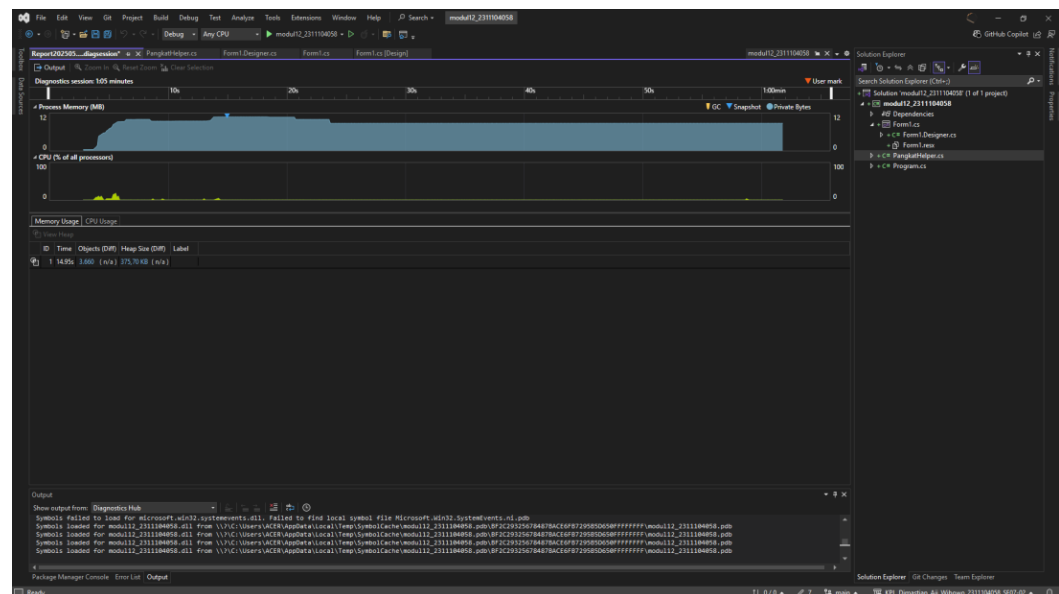
Profiling

A. Idle

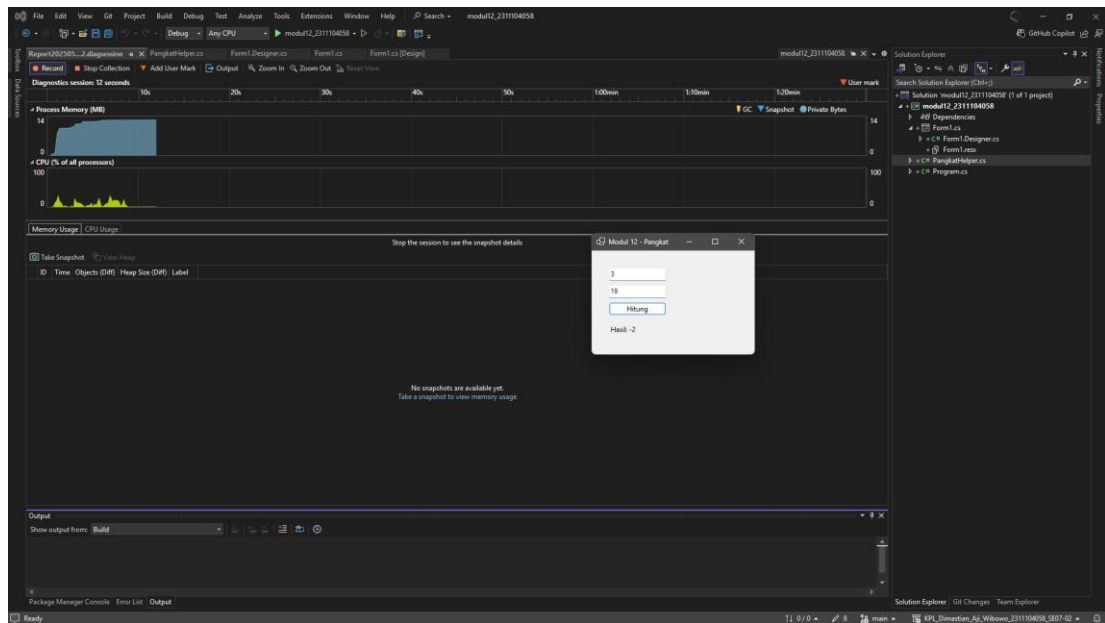
1. Memory



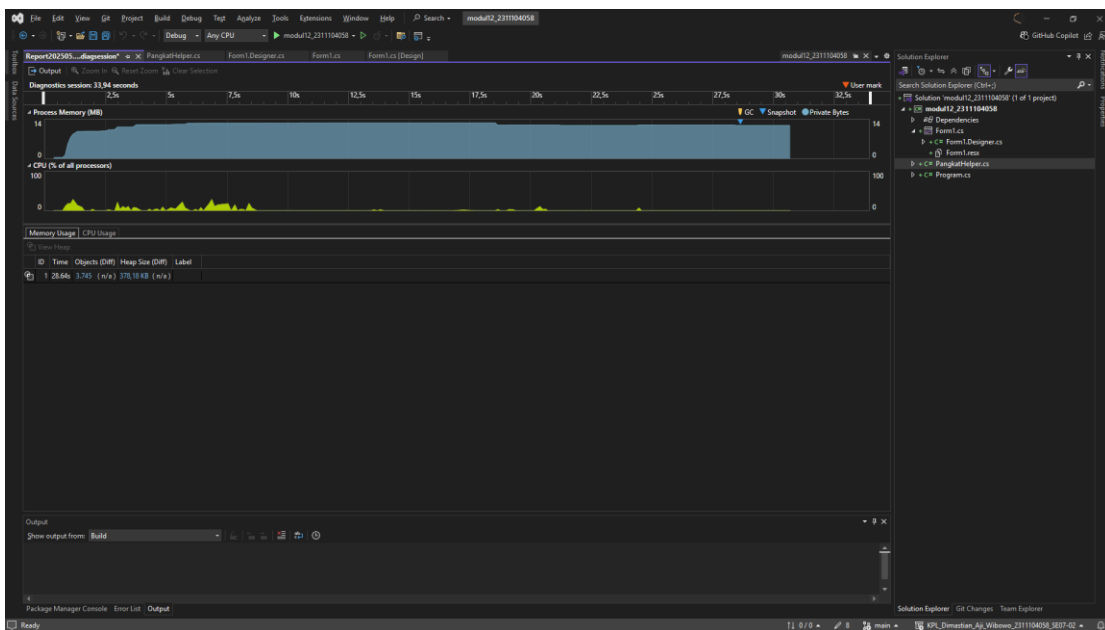
2. CPU



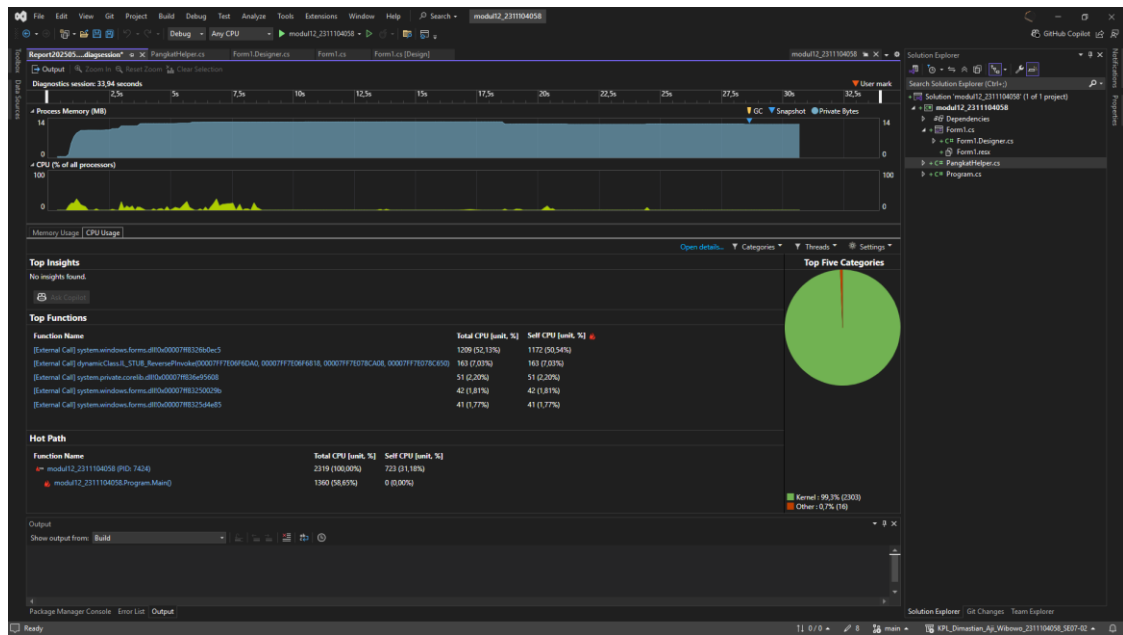
B. Input “3” dan “19”



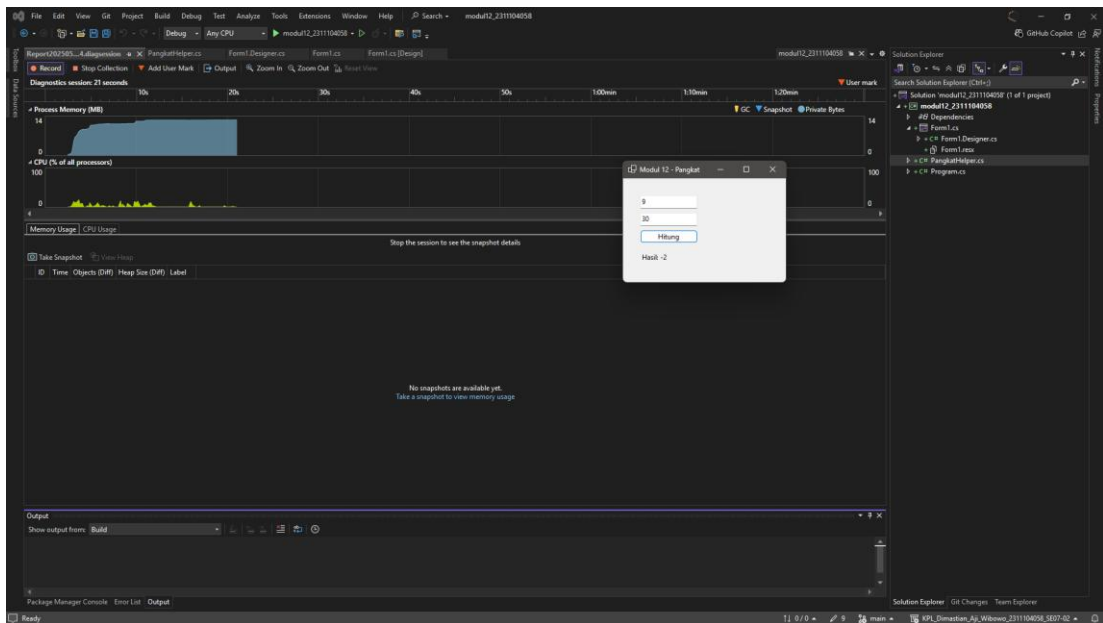
1. Memory



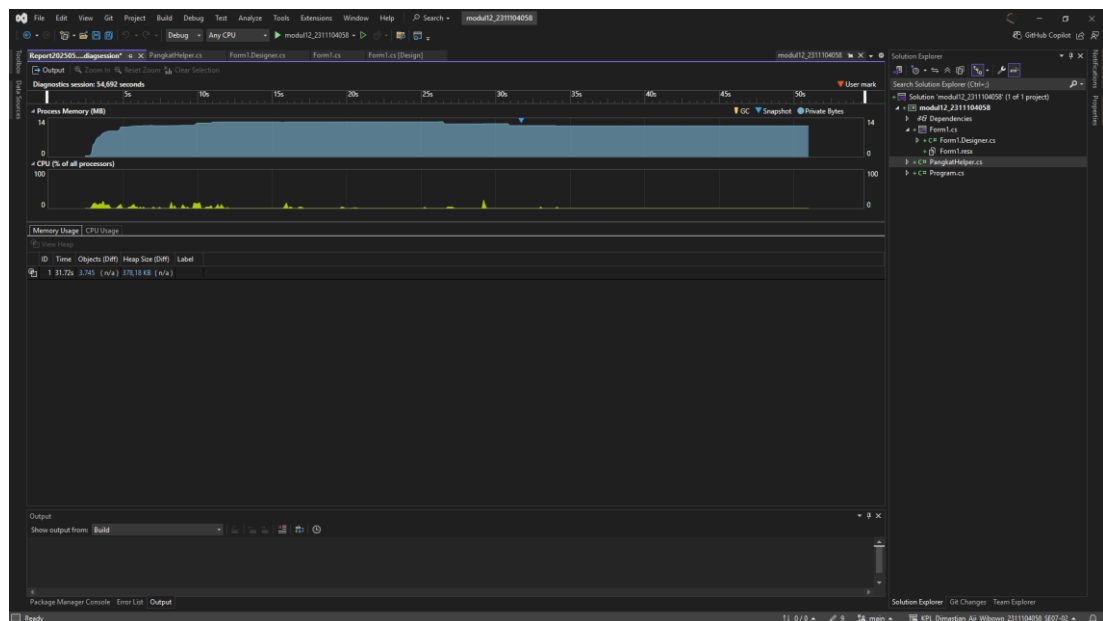
2. CPU



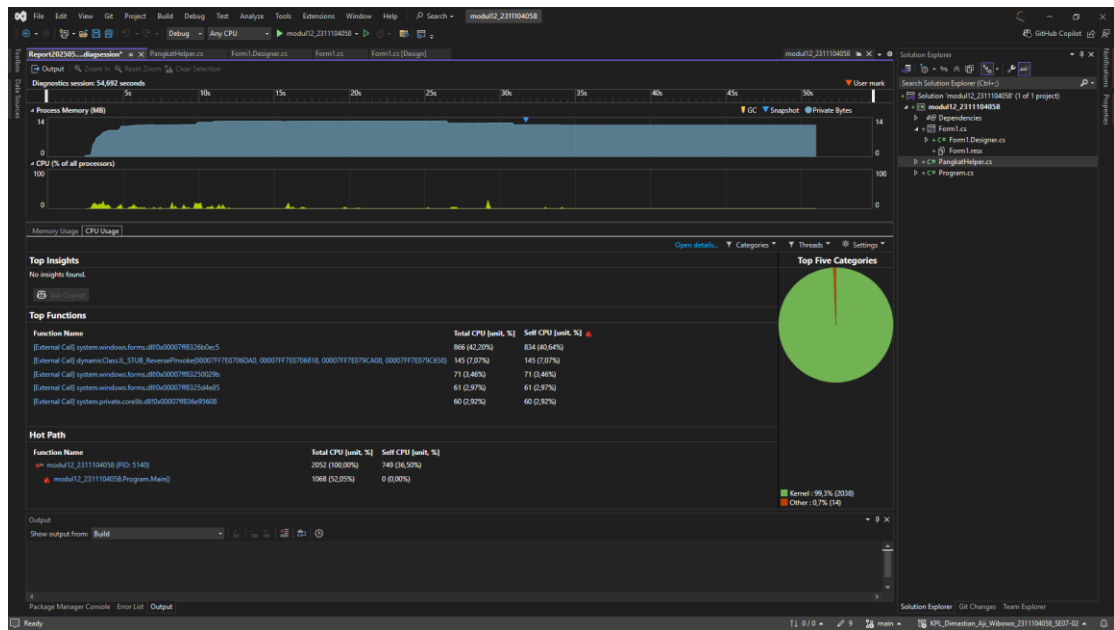
C. Input “9” dan “30”



1. Memory



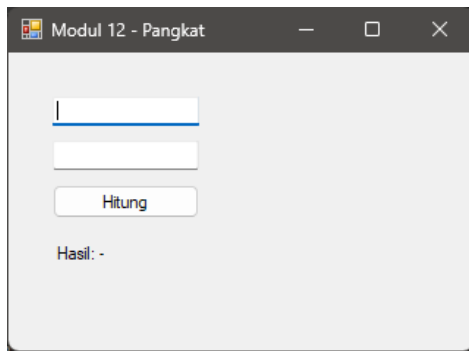
2. CPU



Hasil Unit Test

Test Explorer				Search (Ctrl+I)	
Ready				1 Warning	0 Errors
Test	Duration	Traits	Error Message	Run Debug	
testmodul12 (6)	260 ms				
modul12_123456.Tests (6)	260 ms				
PangkatTests (6)	260 ms				
Test_A_More_Than_100_Should...	260 ms				
Test_B_Equals_0_ShouldReturn_1	< 1 ms				
Test_B_More_Than_10_ShouldR...	< 1 ms				
Test_B_Negative_ShouldReturn_...	< 1 ms				
Test_Normal_Case_ShouldRetur...	< 1 ms				
Test_Overflow_ShouldReturn_M...	< 1 ms				

Windows Form



Source Code

Form1.cs

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

namespace modul12_2311104058
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void btnHitung_Click(object sender, EventArgs e)
        {
            try
            {
                int a = int.Parse(txtA.Text);
                int b = int.Parse(txtB.Text);
                int hasil = PangkatHelper.CariNilaiPangkat(a, b);
                lblHasil.Text = $"Hasil: {hasil}";
            }
            catch (FormatException)
            {
                lblHasil.Text = "Input tidak valid!";
            }
        }
    }
}
```

Kode di atas merupakan bagian dari aplikasi Windows Forms. Aplikasi ini dirancang untuk menghitung nilai pangkat dari dua bilangan yang dimasukkan oleh pengguna. Ketika tombol `btnHitung` diklik, program akan mencoba membaca nilai integer dari dua kotak teks (`txtA` dan `txtB`), lalu menghitung hasil pemangkatan dengan memanggil metode `CariNilaiPangkat` dari kelas `PangkatHelper`. Hasil perhitungannya ditampilkan di label `lblHasil`. Jika input tidak bisa dikonversi ke angka (misalnya karena huruf atau simbol lain), maka akan ditangani dengan `try-catch`, dan program akan menampilkan pesan "Input tidak valid!" sebagai umpan balik.

Form1.Designer.cs

```
namespace modul12_2311104058
{
    partial class Form1
    {
        private System.ComponentModel.IContainer components = null;
        private System.Windows.Forms.TextBox txtA;
        private System.Windows.Forms.TextBox txtB;
        private System.Windows.Forms.Button btnHitung;
        private System.Windows.Forms.Label lblHasil;

        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null)) components.Dispose();
            base.Dispose(disposing);
        }

        private void InitializeComponent()
        {
            this.txtA = new System.Windows.Forms.TextBox();
            this.txtB = new System.Windows.Forms.TextBox();
            this.btnHitung = new System.Windows.Forms.Button();
            this.lblHasil = new System.Windows.Forms.Label();
            this.SuspendLayout();
            //
            // txtA
            //
            this.txtA.Location = new System.Drawing.Point(30, 30);
            this.txtA.Name = "txtA";
            this.txtA.Size = new System.Drawing.Size(100, 20);
            //
            // txtB
            //
            this.txtB.Location = new System.Drawing.Point(30, 60);
            this.txtB.Name = "txtB";
            this.txtB.Size = new System.Drawing.Size(100, 20);
            //
            // btnHitung
            //
            this.btnHitung.Location = new System.Drawing.Point(30, 90);
            this.btnHitung.Name = "btnHitung";
            this.btnHitung.Size = new System.Drawing.Size(100, 23);
            this.btnHitung.Text = "Hitung";
            this.btnHitung.UseVisualStyleBackColor = true;
            this.btnHitung.Click += new System.EventHandler(this.btnHitung_Click);
            //
            // lblHasil
            //
            this.lblHasil.AutoSize = true;
            this.lblHasil.Location = new System.Drawing.Point(30, 130);
            this.lblHasil.Name = "lblHasil";
            this.lblHasil.Size = new System.Drawing.Size(39, 13);
            this.lblHasil.Text = "Hasil: -";
            //
            // Form1
            //
            this.ClientSize = new System.Drawing.Size(284, 181);
            this.Controls.Add(this.txtA);
            this.Controls.Add(this.txtB);
            this.Controls.Add(this.btnHitung);
            this.Controls.Add(this.lblHasil);
            this.Name = "Form1";
            this.Text = "Modul 12 - Pangkat";
            this.ResumeLayout(false);
            this.PerformLayout();
        }
    }
}
```

Kode di atas merupakan bagian dari desain antarmuka pengguna (UI) dalam aplikasi Windows Forms dengan namespace modul12_2311104058. Kelas Form1 bertipe partial dan berisi deklarasi serta inisialisasi komponen UI berupa dua TextBox (txtA dan txtB) untuk input angka,

satu Button (btnHitung) untuk memicu proses perhitungan, serta satu Label (lblHasil) untuk menampilkan hasil. Dalam metode InitializeComponent(), setiap komponen ditempatkan pada posisi tertentu dalam form, dan event Click pada tombol dikaitkan ke metode btnHitung_Click. Form ini akan ditampilkan dengan ukuran 284x181 piksel dan diberi judul "Modul 12 - Pangkat".

PangkatHelper.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace modul12_2311104058
{
    public static class PangkatHelper
    {
        public static int CariNilaiPangkat(int a, int b)
        {
            if (b == 0)
                return 1;
            if (b < 0)
                return -1;
            if (b > 10 || a > 100)
                return -2;

            try
            {
                int hasil = 1;
                checked
                {
                    for (int i = 0; i < b; i++)
                    {
                        hasil *= a;
                    }
                }
                return hasil;
            }
            catch (OverflowException)
            {
                return -3;
            }
        }
    }
}
```

Kode PangkatHelper di atas adalah sebuah kelas statis dalam namespace modul12_2311104058 yang menyediakan fungsi untuk menghitung nilai pangkat dari dua bilangan bulat. Metode CariNilaiPangkat(int a, int b) menerima dua parameter: a sebagai bilangan pokok (basis) dan b sebagai pangkat (eksponen), lalu mengembalikan hasil dari operasi a^b .

Metode ini juga menyertakan beberapa validasi:

1. Jika $b == 0$, hasilnya langsung dikembalikan sebagai 1 (karena bilangan apapun pangkat 0 adalah 1).

2. Jika $b < 0$, fungsi mengembalikan -1 sebagai indikator kesalahan (tidak mendukung pangkat negatif).
3. Jika nilai b lebih dari 10 atau a lebih dari 100, fungsi mengembalikan -2 sebagai batasan agar tidak menghitung nilai terlalu besar.
4. Proses perhitungan dilakukan dalam blok checked untuk mendeteksi overflow. Jika terjadi overflow (hasil terlalu besar untuk tipe int), maka fungsi mengembalikan -3.

Dengan demikian, kelas ini tidak hanya menghitung pangkat, tetapi juga melakukan validasi input dan penanganan kesalahan agar program tetap stabil.

Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace modul12_2311104058
{
    internal static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

Kode di atas merupakan titik awal (entry point) dari aplikasi Windows Forms dalam C#. Kelas Program memiliki metode Main() yang didekorasi dengan atribut [STAThread], yang menunjukkan bahwa aplikasi menggunakan model threading *Single-Threaded Apartment*—dibutuhkan untuk menjalankan elemen UI seperti Clipboard dan OpenFileDialog.

Di dalam Main(), terdapat tiga baris utama:

1. Application.EnableVisualStyles(); — Mengaktifkan tampilan visual modern sesuai tema Windows.
2. Application.SetCompatibleTextRenderingDefault(false); — Menentukan metode perenderan teks agar menggunakan GDI+ (lebih baru).
3. Application.Run(new Form1()); — Menjalankan form utama aplikasi, yaitu Form1, sebagai jendela utama.

Kode ini memastikan aplikasi dijalankan dengan konfigurasi yang tepat untuk antarmuka pengguna grafis.

UnitTest1.cs

```
using Microsoft.VisualStudio.TestTools.UnitTesting;
using modul12_123456;
using modul12_2311104058;

namespace modul12_123456.Tests
{
    [TestClass]
    public class PangkatTests
    {
        [TestMethod]
        public void Test_B_Equals_0_ShouldReturn_1()
        {
            int result = PangkatHelper.CariNilaiPangkat(5, 0);
            Assert.AreEqual(1, result);
        }

        [TestMethod]
        public void Test_B_Negative_ShouldReturn_Minus1()
        {
            int result = PangkatHelper.CariNilaiPangkat(4, -2);
            Assert.AreEqual(-1, result);
        }

        [TestMethod]
        public void Test_B_More_Than_10_ShouldReturn_Minus2()
        {
            int result = PangkatHelper.CariNilaiPangkat(4, 11);
            Assert.AreEqual(-2, result);
        }

        [TestMethod]
        public void Test_A_More_Than_100_ShouldReturn_Minus2()
        {
            int result = PangkatHelper.CariNilaiPangkat(101, 2);
            Assert.AreEqual(-2, result);
        }

        [TestMethod]
        public void Test_Normal_Case_ShouldReturn_CorrectPower()
        {
            int result = PangkatHelper.CariNilaiPangkat(2, 3);
            Assert.AreEqual(8, result);
        }

        [TestMethod]
        public void Test_Overflow_ShouldReturn_Minus3()
        {
            int result = PangkatHelper.CariNilaiPangkat(100, 10);
            Assert.AreEqual(-3, result);
        }
    }
}
```

Kode di atas adalah unit test menggunakan MSTest untuk menguji metode CariNilaiPangkat dari kelas PangkatHelper. Berbagai skenario diuji, mulai dari kasus pangkat nol yang harus menghasilkan 1, pangkat negatif yang harus mengembalikan -1, hingga pembatasan nilai input seperti basis lebih dari 100 atau eksponen lebih dari 10 yang harus mengembalikan -2. Juga diuji kasus normal seperti 2 pangkat 3 yang harus menghasilkan 8, serta deteksi overflow saat menghitung nilai yang terlalu besar seperti 100 pangkat 10 yang harus mengembalikan -3.

Pengujian ini memastikan fungsi berjalan sesuai spesifikasi dan mampu menangani berbagai kondisi input dengan baik.