

Laporan Tugas Jurnal Modul 10

1. OperasiMatematika.cs

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

namespace MatematikaLibraries
{
    public static class OperasiMatematika
    {
        public static int FPB(int input1, int input2)
        {
            input1 = Math.Abs(input1);
            input2 = Math.Abs(input2);

            while (input2 != 0)
            {
                int temp = input2;
                input2 = input1 % input2;
                input1 = temp;
            }
            return input1;
        }

        public static int KPK(int input1, int input2)
        {
            if (input1 == 0 || input2 == 0)
            {
                return 0;
            }
            input1 = Math.Abs(input1);
            input2 = Math.Abs(input2);
            return (input1 * input2) / FPB(input1, input2);
        }

        public static string Turunan(int[] persamaan)
        {
            if (persamaan == null || persamaan.Length == 0)
            {
                return "0";
            }

            StringBuilder sb = new StringBuilder();
            int n = persamaan.Length;

            for (int i = 0; i < n - 1; i++)
            {
                int pangkatAwal = n - 1 - i;
                if (pangkatAwal == 0) continue;

                int koefisienAwal = persamaan[i];
                int koefisienTurunan = koefisienAwal * pangkatAwal;
                int pangkatTurunan = pangkatAwal - 1;
```

```

        if (koefisienTurunan == 0) continue;

        if (sb.Length > 0 && koefisienTurunan > 0)
        {
            sb.Append("+");
        }

        if (koefisienTurunan == -1 && pangkatTurunan != 0)
        {
            sb.Append("-");
        }
        else if (koefisienTurunan != 1 || pangkatTurunan == 0)
        {
            if (koefisienTurunan == -1 && pangkatTurunan == 0)
            {
                sb.Append(koefisienTurunan);
            }
            else if (koefisienTurunan != 1 || pangkatTurunan ==
0)
            {
                if (koefisienTurunan == 1 && pangkatTurunan ==
0)
                {
                    sb.Append(koefisienTurunan);
                }
                else if (koefisienTurunan != 1)
                {
                    sb.Append(koefisienTurunan);
                }
            }
        }

        if (pangkatTurunan > 0)
        {
            sb.Append("x");
            if (pangkatTurunan > 1)
            {
                sb.Append("^").Append(pangkatTurunan);
            }
        }

        if (sb.Length == 0) return "0";

        if (sb.Length > 0 && sb[0] == '+')
        {
            return sb.ToString(1, sb.Length - 1);
        }

        return sb.ToString();
    }

    public static string Integral(int[] persamaan)
    {
        if (persamaan == null || persamaan.Length == 0)
        {
            return "C";
        }
    }

```

```

        StringBuilder sb = new StringBuilder();
        int n = persamaan.Length;

        for (int i = 0; i < n; i++)
        {
            int pangkatAwal = n - 1 - i;
            int koefisienAwal = persamaan[i];

            if (koefisienAwal == 0 && pangkatAwal != 0) continue;

            int pangkatIntegral = pangkatAwal + 1;
            int koefisienIntegral = koefisienAwal / pangkatIntegral;

            if (koefisienAwal == 0 && pangkatAwal == 0 && i == n - 1
&& n == 1 && persamaan[0] == 0)
            {
            }
            else if (koefisienIntegral == 0 && koefisienAwal != 0)
            {
            }

            if (koefisienIntegral == 0 && koefisienAwal == 0 && i ==
n - 1)
            {
            }
            else if (koefisienIntegral != 0 || (koefisienIntegral ==
0 && koefisienAwal == 0))
            {
                if (sb.Length > 0 && koefisienIntegral > 0)
                {
                    sb.Append("+");
                }
                else if (sb.Length > 0 && koefisienIntegral == 0 &&
koefisienAwal == 0 && pangkatAwal == 0 && persamaan[i] == 0)
                {
                }

                if (koefisienIntegral == -1 && pangkatIntegral != 0)
                {
                    sb.Append("-");
                }

                else if (koefisienIntegral != 1 && koefisienIntegral
!= 0)
                {
                    sb.Append(koefisienIntegral);
                }
                else if (koefisienIntegral == 1 && pangkatIntegral
== 0)
                {
                }
                else if (koefisienIntegral == 0 && koefisienAwal ==
0 && pangkatAwal == 0)
                {
            }
        }
    }

```

```

    }
    else if (koefisienIntegral == 1 && pangkatIntegral
!= 0)
    {
    }
    else if (koefisienIntegral != 0)
    {
        sb.Append(koefisienIntegral);
    }

    if (pangkatIntegral > 0)
    {
        sb.Append("x");
        if (pangkatIntegral > 1)
        {
            sb.Append("^").Append(pangkatIntegral);
        }
    }
    }
}

if (sb.Length == 0)
{
    return "C";
}
else
{
    string resultString = sb.ToString();
    if (resultString.StartsWith("+"))
    {
        resultString = resultString.Substring(1);
    }

    if (string.IsNullOrEmpty(resultString) &&
persamaan.All(c => c == 0))
    {
        return "C";
    }
    else if (string.IsNullOrEmpty(resultString) &&
!persamaan.All(c => c == 0))
    {
        return "C";
    }

    bool onlyZeros = true;
    foreach (int kfs in persamaan)
    {
        if (kfs != 0)
        {
            onlyZeros = false;
            break;
        }
    }
    if (onlyZeros && resultString == "C")
    {
        return "C";
    }

    if (!string.IsNullOrEmpty(resultString))
    {
        sb.Append("+C");
    }
}

```

```

        else
        {
            return "C";
        }
    }

    string finalResult = sb.ToString();
    if (finalResult.StartsWith("+") &&
        !finalResult.Contains("C+"))
    {
        if (finalResult.Length > 1 && finalResult[1] == 'C')
        {
        }
        else
        {
            finalResult = finalResult.Substring(1);
        }
    }

    if (finalResult == "+C") return "C";

    return finalResult;
}
}
}

```

Penjelasan:

Kode C# ini mendefinisikan sebuah class library statis bernama OperasiMatematika di dalam namespace MatematikaLibraries. Library ini menyediakan empat fungsi matematika:

Fungsi Matematika

1. FPB(int input1, int input2)

Menghitung Faktor Persekutuan Terbesar (FPB) dari dua bilangan bulat. Fungsi ini menggunakan algoritma Euclidean setelah memastikan kedua input adalah positif.

2. KPK(int input1, int input2)

Menghitung Kelipatan Persekutuan Terkecil (KPK) dari dua bilangan bulat. Jika salah satu input adalah 0, hasilnya adalah 0. Selain itu, KPK dihitung menggunakan rumus yang melibatkan FPB: $(\text{input1} * \text{input2}) / \text{FPB}(\text{input1}, \text{input2})$.

3. Turunan(int[] persamaan)

Menghitung turunan dari sebuah polinomial yang koefisiennya diberikan dalam bentuk array integer (dari pangkat tertinggi ke terendah). Fungsi ini membangun representasi string dari polinomial turunan, dengan memperhatikan format penulisan (misalnya, tanda positif/negatif, penulisan koefisien 1, dan pangkat variabel). Jika input kosong atau null, atau jika

turunannya adalah nol (misalnya, turunan dari konstanta), fungsi akan mengembalikan "0".

4. Integral(int[] persamaan)

Menghitung integral tak tentu dari sebuah polinomial yang koefisiennya diberikan dalam bentuk array integer. Mirip dengan fungsi Turunan, fungsi ini membangun representasi string dari polinomial hasil integral. Fungsi ini juga menangani penambahan konstanta integrasi "+C". Ada beberapa logika untuk memformat output string dengan benar, termasuk penanganan kasus di mana integralnya hanya konstanta "C" (misalnya, integral dari 0). Fungsi ini mengasumsikan bahwa pembagian koefisien dengan pangkat baru akan menghasilkan bilangan bulat, sesuai dengan contoh yang biasanya diberikan dalam konteks seperti ini.

2. Program.cs

```
using System;
using System.Collections.Generic;
using MatematikaLibraries;

class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("=== Mendemokan Library Matematika ===");
        Console.WriteLine("-----");

        int bil1_fpb = 60;
        int bil2_fpb = 45;
        Console.WriteLine($"\\nFPB dari {bil1_fpb} dan {bil2_fpb}:");
        try
        {
            int hasilFPB = OperasiMatematika.FPB(bil1_fpb, bil2_fpb);
            Console.WriteLine($"Hasil: {hasilFPB}");
        }
        catch (Exception ex)
        {
            Console.WriteLine($"Error saat menghitung FPB:
{ex.Message}");
        }

        int bil1_kpk = 12;
        int bil2_kpk = 8;
        Console.WriteLine($"\\nKPK dari {bil1_kpk} dan {bil2_kpk}:");
        try
        {
            int hasilKPK = OperasiMatematika.KPK(bil1_kpk, bil2_kpk);
            Console.WriteLine($"Hasil: {hasilKPK}");
        }
        catch (Exception ex)
        {
            Console.WriteLine($"Error saat menghitung KPK:
{ex.Message}");
        }

        int[] persamaanTurunan = { 1, 4, -12, 9 };
```

```

        Console.WriteLine($"Turunan dari persamaan dengan koefisien
{{{string.Join(", ", persamaanTurunan)}}} (x^3 + 4x^2 - 12x + 9):");
        try
        {
            string hasilTurunan =
OperasiMatematika.Turunan(persamaanTurunan);
            Console.WriteLine($"Hasil: \"{hasilTurunan}\"");
        }
        catch (Exception ex)
        {
            Console.WriteLine($"Error saat menghitung Turunan:
{ex.Message}");
        }

        int[] persamaanIntegral = { 4, 6, -12, 9 };
        Console.WriteLine($"Integral dari persamaan dengan koefisien
{{{string.Join(", ", persamaanIntegral)}}} (4x^3 + 6x^2 - 12x + 9):");
        try
        {
            string hasilIntegral =
OperasiMatematika.Integral(persamaanIntegral);
            Console.WriteLine($"Hasil: \"{hasilIntegral}\"");
        }
        catch (Exception ex)
        {
            Console.WriteLine($"Error saat menghitung Integral:
{ex.Message}");
        }

        Console.WriteLine("\n-----");
        Console.WriteLine("Pengujian selesai. Tekan tombol apa saja
untuk keluar...");
        Console.ReadKey();
    }
}

```

Penjelasan:

Kode C# ini adalah program konsol yang berfungsi untuk menguji coba sebuah pustaka kelas (library) bernama MatematikaLibraries.

Cara Kerja Program

Program ini melakukan hal-hal berikut:

1. **Menggunakan Pustaka:** Ia memanggil fungsi-fungsi dari kelas OperasiMatematika yang ada di dalam MatematikaLibraries.
2. **Menguji Fungsi Matematika:**
 - Menghitung dan menampilkan **FPB** (Faktor Persekutuan Terbesar) dari 60 dan 45.
 - Menghitung dan menampilkan **KPK** (Kelipatan Persekutuan Terkecil) dari 12 dan 8.
 - Menghitung dan menampilkan **turunan** dari polinomial $x^3+4x^2-12x+9$.
 - Menghitung dan menampilkan **integral** dari polinomial $4x^3+6x^2-12x+9$.
3. **Menampilkan Hasil:** Semua hasil perhitungan dicetak ke layar konsol.

4. **Penanganan Error:** Setiap pemanggilan fungsi dibungkus dalam blok try-catch untuk menangani dan menampilkan pesan jika terjadi kesalahan (exception) saat fungsi dari library dijalankan.
5. **Menunggu Input Pengguna:** Program akan berhenti sejenak di akhir dan menunggu pengguna menekan tombol apa saja sebelum jendela konsol tertutup (Console.ReadKey()).

3. Hasil Run

```
=== Mendemokan Library Matematika ===
-----

FPB dari 60 dan 45:
Hasil: 15

KPK dari 12 dan 8:
Hasil: 24

Turunan dari persamaan dengan koefisien {1, 4, -12, 9} ( $x^3 + 4x^2 - 12x + 9$ ):
Hasil: "3x^2+8x-12"

Integral dari persamaan dengan koefisien {4, 6, -12, 9} ( $4x^3 + 6x^2 - 12x + 9$ ):
Hasil: "x^4+2x^3-6x^2+9x+C"
-----
```