

**TUGAS PENDAHULUAN
KONSTRUKSI PERANGKAT LUNAK**

PERTEMUAN 12



**Disusun Oleh :
Muhammad Abdul Aziz
2211104026
SE0601**

**Asisten Praktikum :
Naufal El Kamil Aditya Pratama Rahman
Imelda**

**Dosen Pengampu :
Yudha Islami Sulistya, S.Kom., M.Cs.**

**PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025**

1. Screenshot hasil run

Sebelum Refactoring

```
Microsoft Visual Studio Debug Console
Akar Persamaan Kuadrat:
x1 = 5, x2 = -2

Hasil Kuadrat dari (2x - 3)^2:
4x^2 -12x 9
```

Sesudah Refactoring

```
Microsoft Visual Studio Debug Console
Akar-akar persamaan kuadrat:
2
1

Hasil dari (ax + b)^2:
4x^2 + 12x + 9
```

2. Refactoring

Sebelum Refactoring

Source Code :

Aljabar.cs

```
AljabarLibraries
AljabarLibraries.Aljabar

1 using System;
2
3 namespace AljabarLibraries
4 {
5     2 references
6     public class Aljabar
7     {
8         // Fungsi untuk mencari akar-akar persamaan kuadrat
9         1 reference
10        public static double[] AkarPersamaanKuadrat(double[] persamaan)
11        {
12            double a = persamaan[0];
13            double b = persamaan[1];
14            double c = persamaan[2];
15
16            double diskriminan = b * b - 4 * a * c;
17
18            if (diskriminan < 0)
19            {
20                throw new ArgumentException("Tidak ada akar real");
21            }
22
23            double akar1 = (-b + Math.Sqrt(diskriminan)) / (2 * a);
24            double akar2 = (-b - Math.Sqrt(diskriminan)) / (2 * a);
25
26            return new double[] { akar1, akar2 };
27        }
28
29        // Fungsi untuk menghitung hasil kuadrat dari (ax + b)^2
30        1 reference
31        public static double[] HasilKuadrat(double[] persamaan)
32        {
33            double a = persamaan[0];
34            double b = persamaan[1];
35
36            double koefX2 = a * a;
37            double koefX = 2 * a * b;
38            double konstanta = b * b;
39
40            return new double[] { koefX2, koefX, konstanta };
41        }
42    }
43 }
```

Program.cs

```
modul10_2211104026 tpmodul10_2211104026.Program
1 using System;
2 using AljabarLibraries;
3
4 namespace tpmodul10_2211104026
5 {
6     class Program
7     {
8         static void Main(string[] args)
9         {
10             // Contoh pemanggilan fungsi AkarPersamaanKuadrat
11             double[] hasilAkar = Aljabar.AkarPersamaanKuadrat(new double[] { 1, -3, -10 });
12             Console.WriteLine("Akar Persamaan Kuadrat:");
13             Console.WriteLine($"x1 = {hasilAkar[0]}, x2 = {hasilAkar[1]}");
14
15             // Contoh pemanggilan fungsi HasilKuadrat
16             double[] hasilKuadrat = Aljabar.HasilKuadrat(new double[] { 2, -3 });
17             Console.WriteLine("\nHasil Kuadrat dari (2x - 3)^2:");
18             Console.WriteLine($"{hasilKuadrat[0]}x^2 {hasilKuadrat[1]}x {hasilKuadrat[2]}");
19         }
20     }
21 }
```

Sesudah Refactoring

Source Code :

Aljabar.cs

```
AljabarLibraries AljabarLibraries.Aljabar
1 using System;
2
3 namespace AljabarLibraries
4 {
5     public class Aljabar
6     {
7         // Menghitung akar-akar dari persamaan kuadrat  $ax^2 + bx + c$ 
8         // 1 reference
9         public double[] HitungAkarPersamaanKuadrat(double[] koefisien)
10         {
11             double koefA = koefisien[0];
12             double koefB = koefisien[1];
13             double koefC = koefisien[2];
14
15             double diskriminan = koefB * koefB - 4 * koefA * koefC;
16
17             if (diskriminan < 0)
18             {
19                 return new double[0]; // Tidak memiliki akar real
20             }
21
22             double akar1 = (-koefB + Math.Sqrt(diskriminan)) / (2 * koefA);
23             double akar2 = (-koefB - Math.Sqrt(diskriminan)) / (2 * koefA);
24
25             return new double[] { akar1, akar2 };
26         }
27
28         // Menghitung hasil kuadrat dari bentuk  $(ax + b)^2$ 
29         // 1 reference
30         public double[] HitungHasilKuadrat(double[] koefisien)
31         {
32             double koefA = koefisien[0];
33             double koefB = koefisien[1];
34
35             double kuadratA = koefA * koefA;
36             double duaAB = 2 * koefA * koefB;
37             double kuadratB = koefB * koefB;
38
39             return new double[] { kuadratA, duaAB, kuadratB };
40         }
41     }
42 }
```

Program.cs

```
modul10_2211104026 tpmodul14_2211104026.Program
1 using System;
2 using AljabarLibraries;
3
4 namespace tpmodul14_2211104026
5 {
6     0 references
7     internal class Program
8     {
9         0 references
10        static void Main(string[] args)
11        {
12            Aljabar aljabar = new Aljabar();
13
14            // Input koefisien persamaan kuadrat ax^2 + bx + c
15            double[] koefKuadrat = { 1, -3, 2 };
16            double[] akar = aljabar.HitungAkarPersamaanKuadrat(koefKuadrat);
17
18            Console.WriteLine("Akar-akar persamaan kuadrat:");
19            foreach (double hasil in akar)
20            {
21                Console.WriteLine(hasil);
22            }
23
24            // Input koefisien dari bentuk (ax + b)^2
25            double[] koefLinier = { 2, 3 };
26            double[] hasilKuadrat = aljabar.HitungHasilKuadrat(koefLinier);
27
28            Console.WriteLine("\nHasil dari (ax + b)^2:");
29            Console.WriteLine($"{hasilKuadrat[0]}x^2 + {hasilKuadrat[1]}x + {hasilKuadrat[2]}");
30        }
31    }
32 }
```

Penjelasan Perubahan (refactoring)

Aspek Refactoring	Penjelasan
Naming Convention	Method seperti AkarPersamaanKuadrat() diubah menjadi HitungAkarPersamaanKuadrat() untuk lebih deskriptif dan sesuai PascalCase. Variabel a, b, c diubah menjadi koefA, koefB, koefC agar lebih jelas.
Whitespace & Indentasi	Semua indentasi distandarkan menggunakan 4 spasi, tidak ada baris kosong berlebihan, dan antar method dipisahkan baris.
Deklarasi Variabel	Variabel hasil perhitungan diubah menjadi nama yang lebih deskriptif (akar1, duaAB, dll).
Komentar	Ditambahkan komentar untuk menjelaskan fungsi dan proses dalam method.
Readability	Struktur program dibuat lebih modular, ringkas, dan jelas.