

**TUGAS PENDAHULUAN  
KONSTRUKSI PERANGKAT LUNAK**

**MODUL X  
LIBRARY CONSTRUCTION**



**Disusun Oleh :**

Nadia Putri Rahmaniar / 2211104012

S1 SE-06-01

**Dosen Pengampu :**

Yudha Islami Sulistya, S.Kom., M.Cs.

**PROGRAM STUDI S1 SOFTWARE ENGINEERING FAKULTAS  
INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2025**

## TUGAS PENDAHULUAN

### 1. MEMBUAT PROJECT DAN CLASS LIBRARY

Buka IDE misalnya dengan Visual Studio

A. Misalnya menggunakan Visual Studio, buatlah solution baru dengan nama tpmodul10\_NIM yang berisi blank project. dapat

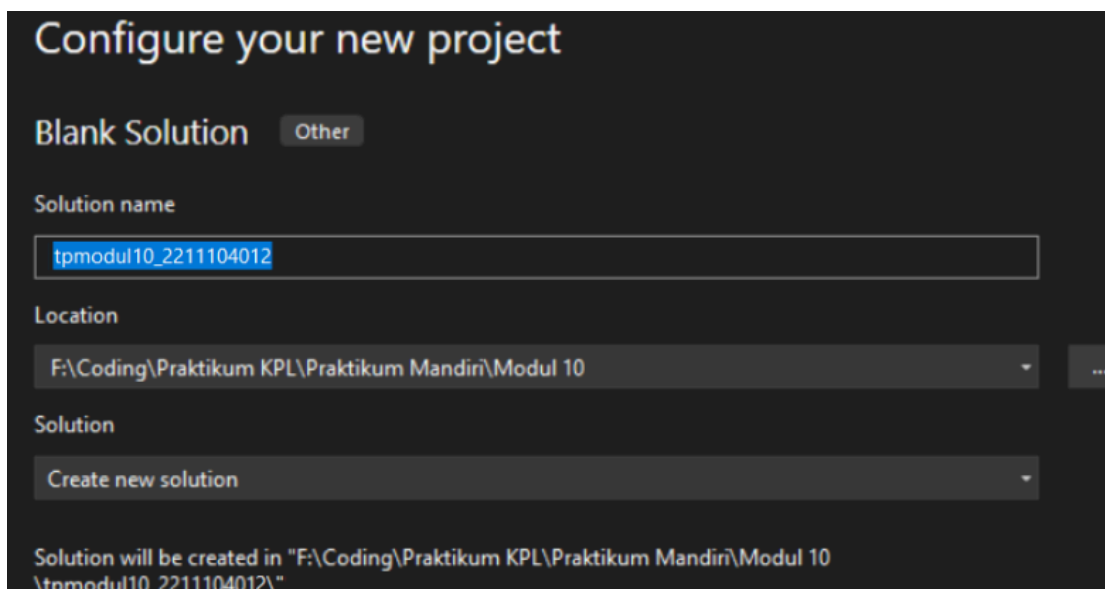
B. Setelah itu, buatlah class library project di dalam solution tersebut.

Langkah-langkah dengan Visual studio dicek di link berikut:

<https://docs.microsoft.com/enus/dotnet/core/tutorials/library-with-visual-studio?pivot=dotnet-6-0>

**Jawab:**

- Blank Project



### 2. MEMBUAT LIBRARY ALJABAR

Buatlah suatu library bernama (namespace) AljabarLibraries yang melakukan proses kalkulasi untuk persamaan kuadrat:

$$ax^2 + bx + c = 0.$$

Library ini mempunyai dua fungsi sebagai berikut.

A. Mencari akar-akar dari persamaan kuadrat:

double[] AkarPersamaanKuadrat(double[] persamaan)

Coba ingat-ingat rumus untuk mencari akar-akarnya sebagai berikut

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Contohnya untuk persamaan:  $x^2 - 3x - 10 = 0$  mempunyai akar 5 dan -2. Proses pemanggilan dari fungsi ini adalah sebagai berikut:

AkarPersamaanKuadrat({1, -3, -10})

Output: {5, -2}

B. Mendapatkan hasil kuadrat dari persamaan berpangkat

1: double[] HasilKuadrat(double[] persamaan)

Rumus singkat untuk mendapatkan hasil kuadrat persamaan sebagai berikut:

$$(a + b)^2 = a^2 + 2ab + b^2$$

Contohnya untuk persamaan:  $2x - 3$  hasil kuadratnya adalah  $4x^2 - 12x + 9$

Proses pemanggilan dari fungsi ini adalah sebagai berikut:

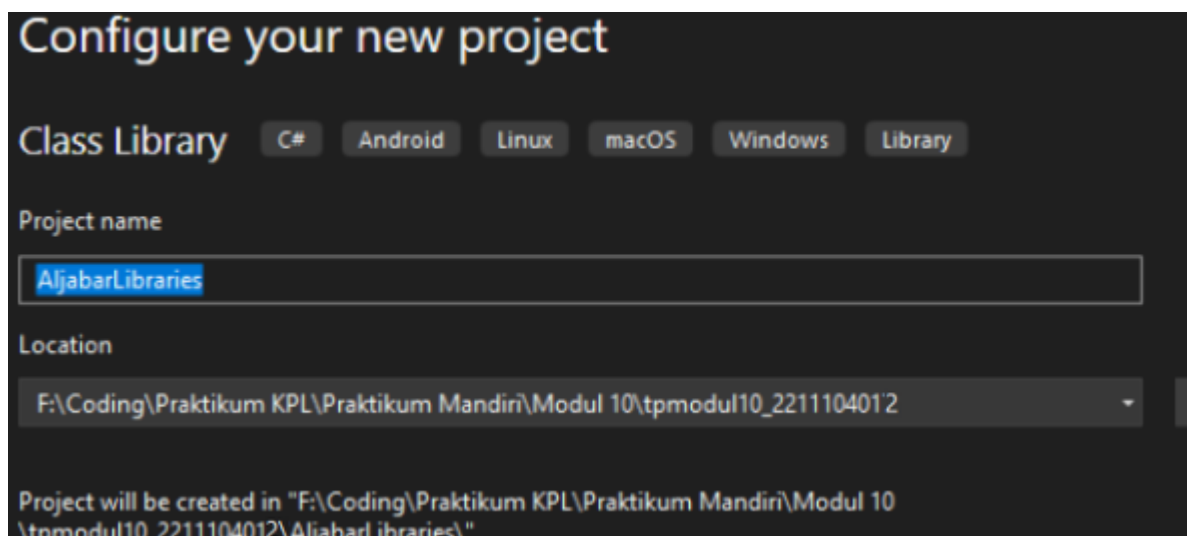
Hasil Kuadrat({2, -3})

Output: {4, -12, 9}

**Jawab:**

- **Source code**

Membuat library AljabarLibraries



File Aljabar.cs

```
using System;

namespace AljabarLibraries
{
    2 references
    public class Aljabar
    {
        1 reference
        public static double[] AkarPersamaanKuadrat(double[] persamaan)
        {
            double a = persamaan[0];
            double b = persamaan[1];
            double c = persamaan[2];
            double diskriminan = b * b - 4 * a * c;

            if (diskriminan < 0)
            {
                return new double[0];
            }

            double x1 = (-b + Math.Sqrt(diskriminan)) / (2 * a);
            double x2 = (-b - Math.Sqrt(diskriminan)) / (2 * a);

            return new double[] { x1, x2 };
        }

        1 reference
        public static double[] HasilKuadrat(double[] persamaan)
        {
            double a = persamaan[0];
            double b = persamaan[1];

            double a_kuadrat = a * a;
            double ab_dua = 2 * a * b;
            double b_kuadrat = b * b;

            return new double[] { a_kuadrat, ab_dua, b_kuadrat };
        }
    }
}
```

### 3. MEMANGGIL LIBRARY DI FUNGSI UTAMA

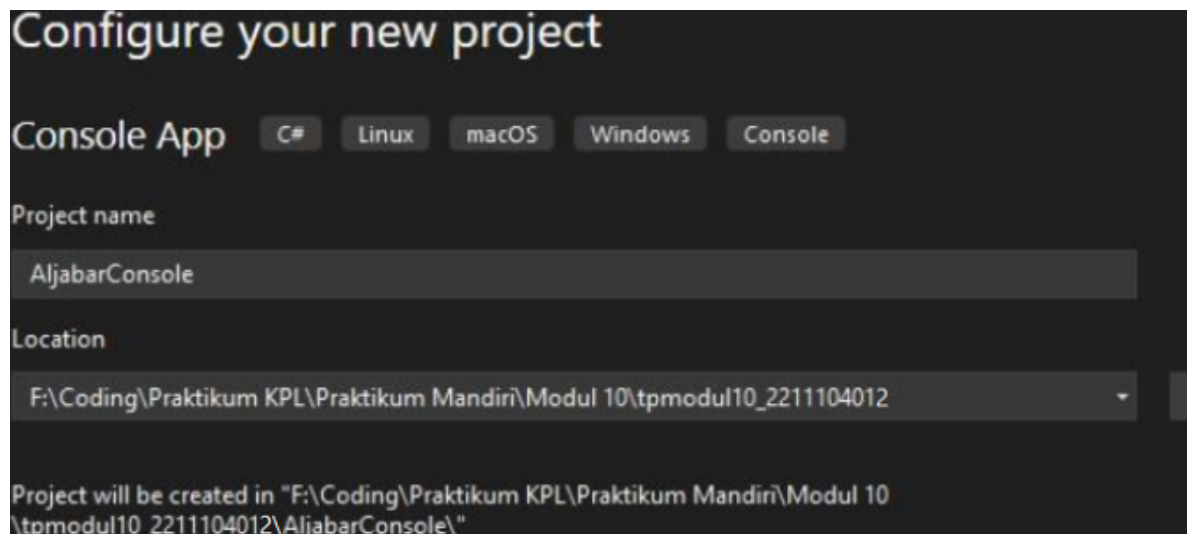
Setelah library selesai dibuat:

- A. Buatlah sebuah **console application project baru** yang memanggil semua fungsi di library AljabarLibraries sebelumnya, misalnya: Aljabar.HasilKuadrat({2, -3})
- B. Catatan: pada pengerjaan modul ini diminta untuk memanfaatkan **Class Library** dan menambahkan library AljabarLibraries. Misalnya dengan Visual Studio dapat dilakukan dengan cara menambahkannya di Project Reference (dependencies) untuk console project yang baru dibuat.

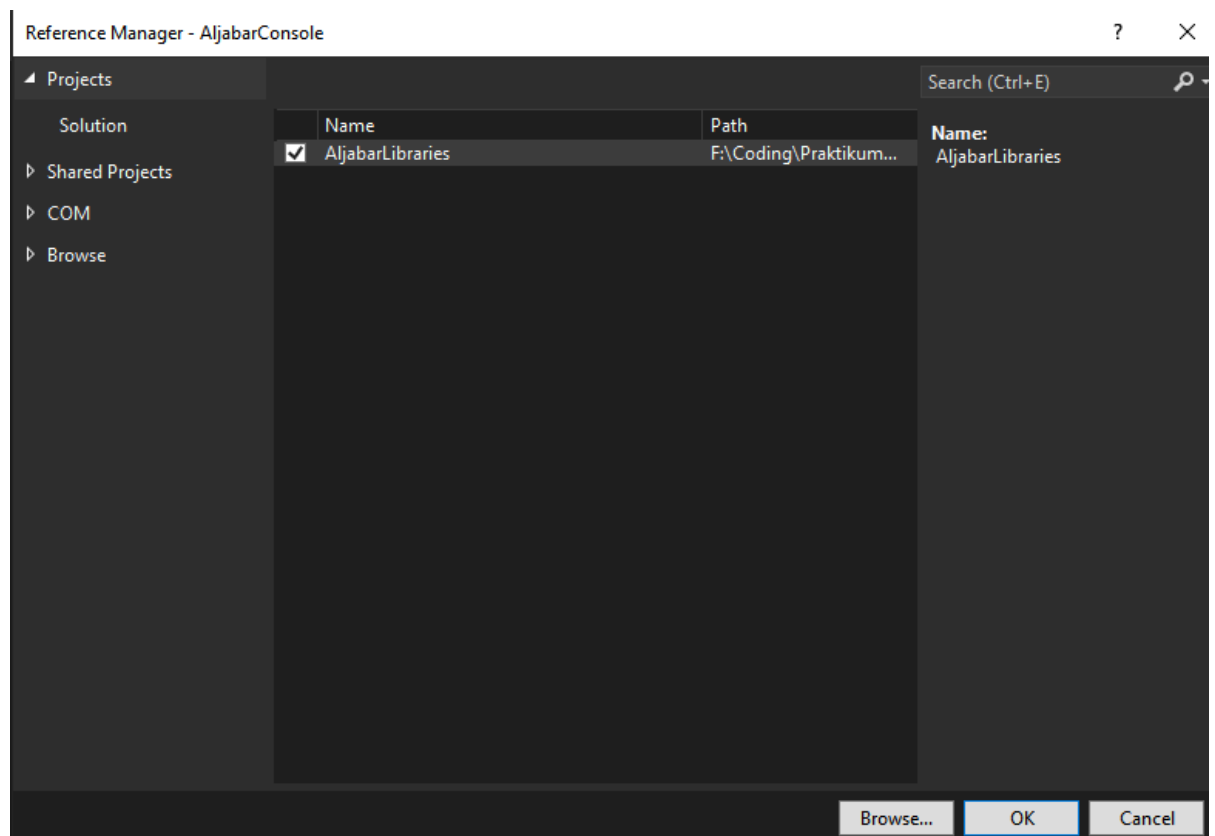
**Jawab:**

- **Source code**

Membuat console application project baru bernama “AljabarConsole”



Menambahkan Referensi ke Library “AljabarLibraries”



File Program.cs

```
using System;
using AljabarLibraries;

namespace AljabarConsole
{
    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {
            Console.WriteLine("--- Nama: Nadia Putri Rahmantiar ---");
            Console.WriteLine("--- NIM: 2211104012 ---");
            Console.WriteLine("--- Kelas: SE0601 ---");

            // AkarPersamaanKuadrat
            double[] persamaan1 = { 1, -3, -10 };
            double[] akar = Aljabar.AkarPersamaanKuadrat(persamaan1);

            Console.WriteLine("\nHasil Akar Persamaan:");
            foreach (double x in akar)
            {
                Console.WriteLine(x);
            }

            // HasilKuadrat
            double[] persamaan2 = { 2, -3 };
            double[] kuadrat = Aljabar.HasilKuadrat(persamaan2);

            Console.WriteLine("\nHasil Kuadrat:");
            foreach (double x in kuadrat)
            {
                Console.WriteLine(x);
            }
        }
    }
}
```

- Hasil

```
--- Nama : Nadia Putri Rahmantiar ---
--- NIM : 2211104012 ---
--- Kelas: SE0601 ---

Diketahui Akar Persamaan Kuadrat = {1, -3, -10}
Hasil Akar Persamaan:
5
-2

Diketahui Kuadrat = {2, -3}
Hasil Kuadrat:
4
-12
9
```

- **Penjelasan**

Library Construction diterapkan untuk membangun dan memanfaatkan pustaka kode kustom. Pada file `Aljabar.cs`, terdapat kelas `Aljabar` yang merupakan bagian dari namespace `AljabarLibraries`, berisi berbagai fungsi matematis. Fungsi `AkarPersamaanKuadrat` dirancang untuk menyelesaikan persamaan kuadrat menggunakan rumus ABC, sementara `HasilKuadrat` bertugas menghitung hasil dari bentuk  $(a+b)^2$  dan mengembalikannya dalam format array. Mengingat ini adalah sebuah class library, kode di dalamnya tidak dapat dieksekusi secara langsung, melainkan harus dipanggil oleh proyek lain.

Selanjutnya, file `Program.cs` berfungsi sebagai pemanggil dan eksekutor fungsi-fungsi yang tersedia dalam library `AljabarLibraries`. Program ini memulai dengan menampilkan informasi identitas pengguna, kemudian melanjutkan untuk menghitung dan menyajikan akar-akar persamaan kuadrat serta hasil kuadrat dari dua bilangan. Pendekatan ini menawarkan keuntungan signifikan dalam hal pemisahan logika: logika perhitungan dienkapsulasi dalam library (`Aljabar.cs`), sedangkan logika tampilan dan eksekusi program berada di proyek utama (`Program.cs`). Hasilnya adalah program yang lebih terstruktur, modular, dan dapat dikembangkan dengan lebih mudah.