

**LAPORAN JURNAL  
MODUL 4**



**Disusun Oleh :**

**Zaenarif Putra 'Ainurdin – 2311104049**

**Kelas :**

**SE-07-02**

**Dosen :**

**Yudha Islami Sulistya**

**PROGRAM STUDI SOFTWARE ENGINEERING  
DIREKTORAT KAMPUS PURWOKERTO  
TELKOM UNIVERSITY  
PURWOKERTO  
2025**

## I. Link Github

[https://github.com/zaenarifputra/KPL\\_Zaenarif-Putra-Ainurdin\\_2311104049\\_S1SE-07-02/tree/5f9002e73ba8f850f6f32bf81935a3e32d5999c8/04\\_Automata dan Table-Driven Construction/tjmodul4\\_2311104049/tjmodul4\\_2311104049](https://github.com/zaenarifputra/KPL_Zaenarif-Putra-Ainurdin_2311104049_S1SE-07-02/tree/5f9002e73ba8f850f6f32bf81935a3e32d5999c8/04_Automata%20dan%20Table-Driven%20Construction/tjmodul4_2311104049/tjmodul4_2311104049)

## II. Hasil Running

```
C:\Users\LENOVO\source\rep x + v
=== Selamat datang di program kode buah ===
Masukkan nama buah:
Apel
Kode buah untuk Apel: A00

=== Selamat datang di program karakter game ===
Masukkan NIM Anda: 2311104049
Karakter mulai dari posisi berdiri.
Posisi standby

Tekan W (naik), S (turun), atau X (keluar):
W
Tombol arah atas ditekan
Karakter mulai terbang.

Tekan W (naik), S (turun), atau X (keluar):
W
Tidak bisa naik lebih tinggi.

Tekan W (naik), S (turun), atau X (keluar):
W
Tidak bisa naik lebih tinggi.

Tekan W (naik), S (turun), atau X (keluar):
S
Tombol arah bawah ditekan
Karakter kembali berdiri.

Tekan W (naik), S (turun), atau X (keluar):
X
Keluar dari program.
```

## III. Penjelasan Syntax Secara Singkat

### 1. Implementasi Class KodeBuah

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

namespace tjmodul4_2311104049
{
    class KodeBuah
    {
        private static Dictionary<string, string> KodeBuahDict = new Dictionary<string, string>
        {
            {"Apel", "A00"}, {"Aprikot", "B00"}, {"Alpukat", "C00"},
            {"Pisang", "D00"}, {"Paprika", "E00"}, {"Kurma", "K00"},
            {"Durian", "L00"}, {"Anggur", "M00"}, {"Melon", "N00"},
            {"Semangka", "O00"}
        };
        public static string getKodeBuah(string buah)
        {
            return KodeBuahDict.ContainsKey(buah) ? KodeBuahDict[buah]
                : "Kode buah tidak ditemukan";
        }
    }
}
```

Program C# ini memanfaatkan struktur data Dictionary untuk menyimpan koleksi nama buah beserta kode unik yang terkait. Fungsi `getKodeBuah()` dirancang untuk menerima nama buah sebagai parameter dan akan mengembalikan kode buah yang relevan jika nama tersebut ada dalam koleksi, atau memberikan informasi bahwa kode buah tidak tersedia jika nama buah tersebut tidak terdaftar.

## 2. Implementasi Class PosisiKarakterGame

```
using System;

namespace tjmodul4_2311104049
{
    public interface IState
    {
        void TombolW(Karakter karakter);
        void TombolS(Karakter karakter);
    }

    public class Berdiri : IState
    {
        public void TombolW(Karakter karakter)
        {
            Console.WriteLine("Tombol arah atas ditekan"); // NIM % 3 == 0
            karakter.SetState(new Terbang());
            Console.WriteLine("Karakter mulai terbang.");

            if (karakter.NIM % 3 == 2)
            {
                Console.WriteLine("Posisi take off");
            }
        }

        public void TombolS(Karakter karakter)
        {
            Console.WriteLine("Tombol arah bawah ditekan"); // NIM % 3 == 0
            karakter.SetState(new Jongkok());
            Console.WriteLine("Karakter mulai jongkok.");
        }
    }
}
```

```
    {
        public void TombolW(Karakter karakter)
        {
            Console.WriteLine("Tombol arah atas ditekan"); // NIM % 3 == 0
            karakter.SetState(new Berdiri());
            Console.WriteLine("Karakter kembali berdiri.");

            if (karakter.NIM % 3 == 1)
            {
                Console.WriteLine("Posisi standby");
            }
        }

        public void TombolS(Karakter karakter)
        {
            Console.WriteLine("Tombol arah bawah ditekan"); // NIM % 3 == 0
            karakter.SetState(new Tengkurap());
            Console.WriteLine("Karakter tengkurap.");

            if (karakter.NIM % 3 == 1)
            {
                Console.WriteLine("Posisi istirahat");
            }
        }
    }

    public class Tengkurap : IState
    {
        public void TombolW(Karakter karakter)
        {
            Console.WriteLine("Tombol arah atas ditekan"); // NIM % 3 == 0
            karakter.SetState(new Jongkok());
            Console.WriteLine("Karakter kembali jongkok.");
        }

        public void TombolS(Karakter karakter)
        {
            Console.WriteLine("Tidak bisa turun lagi, sudah posisi terendah.");
        }
    }
}
```

```
public class Terbang : IState
{
    public void TombolW(Karakter karakter)
    {
        Console.WriteLine("Tidak bisa naik lebih tinggi.");
    }

    public void TombolS(Karakter karakter)
    {
        Console.WriteLine("Tombol arah bawah ditekan"); // NIM % 3 == 0
        karakter.SetState(new Berdiri());
        Console.WriteLine("Karakter kembali berdiri.");

        if (karakter.NIM % 3 == 2)
        {
            Console.WriteLine("Posisi landing");
        }
    }
}

public class Karakter
{
    private IState _state;
    public long NIM { get; private set; }

    public Karakter(long nim)
    {
        NIM = nim;
        _state = new Berdiri(); // Default state
        Console.WriteLine("Karakter mulai dari posisi berdiri.");

        if (NIM % 3 == 1)
        {
            Console.WriteLine("Posisi standby");
        }
    }
}
```

```
public void SetState(IState state)
{
    _state = state;
}

public void TekanTombolW()
{
    _state.TombolW(this);
}

public void TekanTombolS()
{
    _state.TombolS(this);
}
```

Program C# ini mengimplementasikan pola State untuk mengatur pergerakan karakter berdasarkan input dari tombol W (naik) dan S (turun). Karakter memiliki beberapa status, yaitu Berdiri, Jongkok, Tengkurap, dan Terbang, dengan transisi yang ditentukan oleh tindakan pengguna. Setiap perubahan status dipengaruhi oleh nilai  $NIM \% 3$ , yang menetapkan kondisi tambahan seperti standby, take off, atau landing. Ketika program dijalankan, karakter mulai dalam posisi berdiri dan beradaptasi sesuai dengan tombol yang ditekan, memberikan fleksibilitas dalam simulasi pergerakan.

### 3. Implementasi Class Program

```
using System;

namespace tjmodul4_2311104049
{
    class Program
    {
        static void Main()
        {
            Console.WriteLine("=== Selamat datang di program kode buah ===");
            Console.WriteLine("Masukkan nama buah: ");
            string inputBuah = Console.ReadLine();
            string kode = KodeBuah.getKodeBuah(inputBuah);
            Console.WriteLine($"Kode buah untuk {inputBuah}: {kode}");
            Console.ReadLine();

            Console.WriteLine("=== Selamat datang di program karakter game ===");
            Console.WriteLine("Masukkan NIM Anda: ");
            string input = Console.ReadLine();

            if (!long.TryParse(input, out long nim))
            {
                Console.WriteLine("NIM tidak valid! Masukkan angka yang benar.");
                return;
            }

            Karakter karakter = new Karakter(nim);

            // Loop untuk menerima input pengguna
            while (true)
            {
                Console.WriteLine("\nTekan W (naik), S (turun), atau X (keluar): ");
                string perintah = Console.ReadLine().ToUpper();

                if (perintah == "X")
                {
                    Console.WriteLine("Keluar dari program.");
                    break;
                }

                // Loop untuk menerima input pengguna
                while (true)
                {
                    Console.WriteLine("\nTekan W (naik), S (turun), atau X (keluar): ");
                    string perintah = Console.ReadLine().ToUpper();

                    if (perintah == "X")
                    {
                        Console.WriteLine("Keluar dari program.");
                        break;
                    }
                    else if (perintah == "W")
                    {
                        karakter.TekanTombolW();
                    }
                    else if (perintah == "S")
                    {
                        karakter.TekanTombolS();
                    }
                    else
                    {
                        Console.WriteLine("Perintah tidak dikenali. Silakan tekan W, S, atau X.");
                    }
                }
                Console.ReadLine();
            }
        }
    }
}
```

Program C# ini mengintegrasikan dua fitur utama: pencarian kode untuk buah dan simulasi pergerakan karakter dalam permainan. Pengguna dapat memasukkan nama buah untuk memperoleh kode uniknya melalui metode `getKodeBuah()`. Selanjutnya, program meminta NIM pengguna untuk menetapkan kondisi awal karakter, yang dikelola menggunakan pola State. Pengguna dapat menekan tombol W (naik), S (turun), atau X (keluar) untuk mengubah status karakter, dengan aturan transisi yang bergantung pada nilai  $NIM \% 3$ . Program beroperasi dalam sebuah loop hingga pengguna memilih untuk keluar, sehingga menciptakan interaksi yang dinamis dan fleksibel.