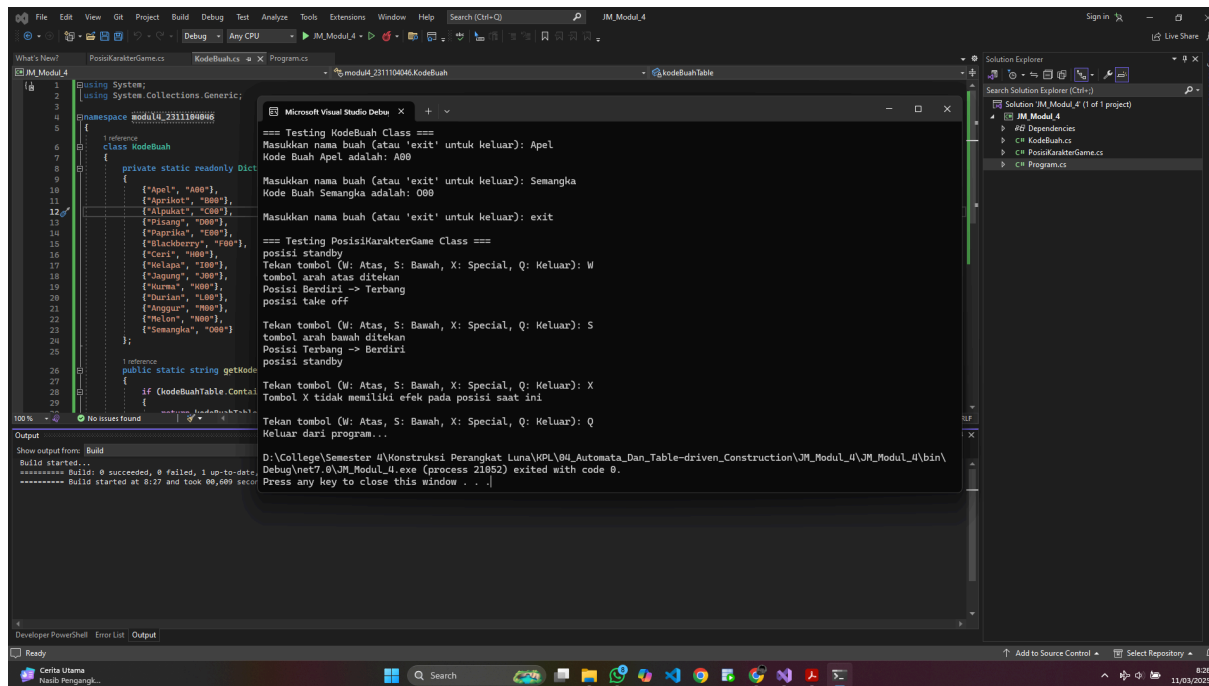


Jurnal Modul 4

Dhiemas Tulus Ikhsan
2311104046



Penjelasan Program:

Kode di atas terdiri dari tiga kelas utama: `KodeBuah`, `PosisiKarakterGame`, dan `Program`. Setiap kelas memiliki fungsi dan logika tersendiri dalam program berbasis konsol ini.

1. Kelas `KodeBuah`

Kelas `KodeBuah` digunakan untuk menyimpan dan mengambil kode buah berdasarkan nama buah tertentu. Pada kelas ini, digunakan struktur data `Dictionary<string, string>` untuk menyimpan pasangan nama buah sebagai key dan kode buah sebagai value. Deklarasinya menggunakan modifier `private static readonly`, yang berarti kamus ini hanya dapat diinisialisasi sekali dan bersifat statis, sehingga dapat diakses tanpa membuat objek dari kelas `KodeBuah`.

Method `getKodeBuah(string namaBuah)` bersifat static, artinya method ini bisa langsung dipanggil dengan `KodeBuah.getKodeBuah()` tanpa membuat instance dari kelas `KodeBuah`. Method ini memeriksa apakah `namaBuah` ada dalam kamus menggunakan `ContainsKey()`. Jika ditemukan, maka kode buah akan dikembalikan. Jika tidak, method akan mengembalikan string "Buah tidak ditemukan".

2. Kelas `PosisiKarakterGame`

Kelas `PosisiKarakterGame` mengimplementasikan logika perubahan posisi karakter dalam game menggunakan pendekatan Finite State Machine (FSM). Untuk

mendefinisikan posisi yang mungkin, digunakan enum bernama `State` dengan empat status: `Berdiri`, `Jongkok`, `Tengkurap`, dan `Terbang`.

Constructor kelas `PosisiKarakterGame` menginisialisasi posisi awal karakter menjadi `Berdiri` dan menampilkan pesan "posisi standby". Hal ini juga berkaitan dengan kondisi khusus di mana `NIM % 3 == 1` yang berarti pada kondisi ini, posisi standby ditampilkan.

Method `TombolS()` digunakan untuk menangani aksi ketika tombol S ditekan. Pada method ini digunakan struktur switch-case untuk menentukan perubahan posisi karakter sesuai dengan kondisi saat ini. Contohnya, ketika karakter dalam posisi `Berdiri` dan tombol S ditekan, maka karakter akan berubah menjadi `Jongkok` dan menampilkan pesan "Posisi Berdiri -> Jongkok".

Method `TombolW()` mirip dengan `TombolS()`, tetapi untuk tombol W (arah atas). Jika karakter dalam posisi `Berdiri`, maka akan berubah menjadi `Terbang` dan menampilkan pesan "posisi take off" (kondisi ini sesuai dengan `NIM % 3 == 2`). Begitu juga dengan kondisi lainnya, method ini memastikan setiap transisi posisi dijalankan sesuai logika yang telah ditentukan.

Method `TombolX()` adalah method khusus untuk aksi spesial. Jika karakter dalam posisi `Terbang`, maka tombol X akan membuat karakter kembali ke posisi `Jongkok` dengan pesan "posisi landing". Jika karakter dalam posisi lain, tombol X tidak akan memberikan efek apapun.

3. Kelas `Program`

Kelas `Program` berfungsi sebagai entry point dari program ini dan mengintegrasikan kelas `KodeBuah` dan `PosisiKarakterGame`. Program dimulai dengan menampilkan antarmuka konsol untuk menguji kelas `KodeBuah`. Pengguna dapat memasukkan nama buah dan mendapatkan kode buah terkait menggunakan method `KodeBuah.getKodeBuah(namaBuah)`. Jika pengguna mengetikkan "exit", maka program akan keluar dari bagian ini.

Setelah bagian kode buah, program melanjutkan ke pengujian kelas `PosisiKarakterGame`. Objek `PosisiKarakterGame` dibuat, dan pengguna dapat menekan tombol W, S, X, atau Q untuk mengontrol posisi karakter dalam game. Tombol Q digunakan untuk keluar dari program, sementara tombol lainnya memanggil method yang sesuai dari kelas `PosisiKarakterGame`.

Program membaca input tombol menggunakan `Console.ReadKey()` dan menyimpannya dalam `ConsoleKeyInfo keyInfo`. Kemudian menggunakan switch-case untuk mengeksekusi method yang sesuai berdasarkan tombol yang

ditekan. Jika tombol yang ditekan tidak dikenali, program akan menampilkan pesan "Tombol tidak dikenali".