Design Patterns

The Design Patterns are blueprints that are utilized to solve the design problems in the code. They are Creational, Structural, and Behavioral patterns which are used to improve the code performance in terms of efficacy, flexibility, and assignment of responsibilities between various objects of the program and better organization. For the flappy bird game, we need scrolling pipes generated in every game loop and the top and bottom pipes orientation are decided based on the coordinates at the time of creation. Since the orientation of the object is decided at the time of creation, it is a wise plan to use the Builder pattern for pipe generation. And this is adopted for the bird object as well.

When looking at the game instances, we shall have a single bird object of class bird whose surface, positions are used in various game functionalities like moving through the pipes, checking for the collide, and tracking the game state. So the design pattern that we adopted for the bird object is Singleton.

For the collide functionality, we used the already existing pipe list to check for any collisions between the bird object and the pipe object, so it’s the Builder pattern again that is used to achieve it. Finally for tracking the game state and starting the bird animations like flapping, jump and move, we used the pattern Observer. This is because we had a variable called START which is set true for the game to start and as the game continues the various operations are decided by the START value and the game ends when it is updated to false subsequently after the first collision.

The design patterns for the Project 4 framework will be discussed in the Project 4 documentation.