Design Patterns: State Pattern

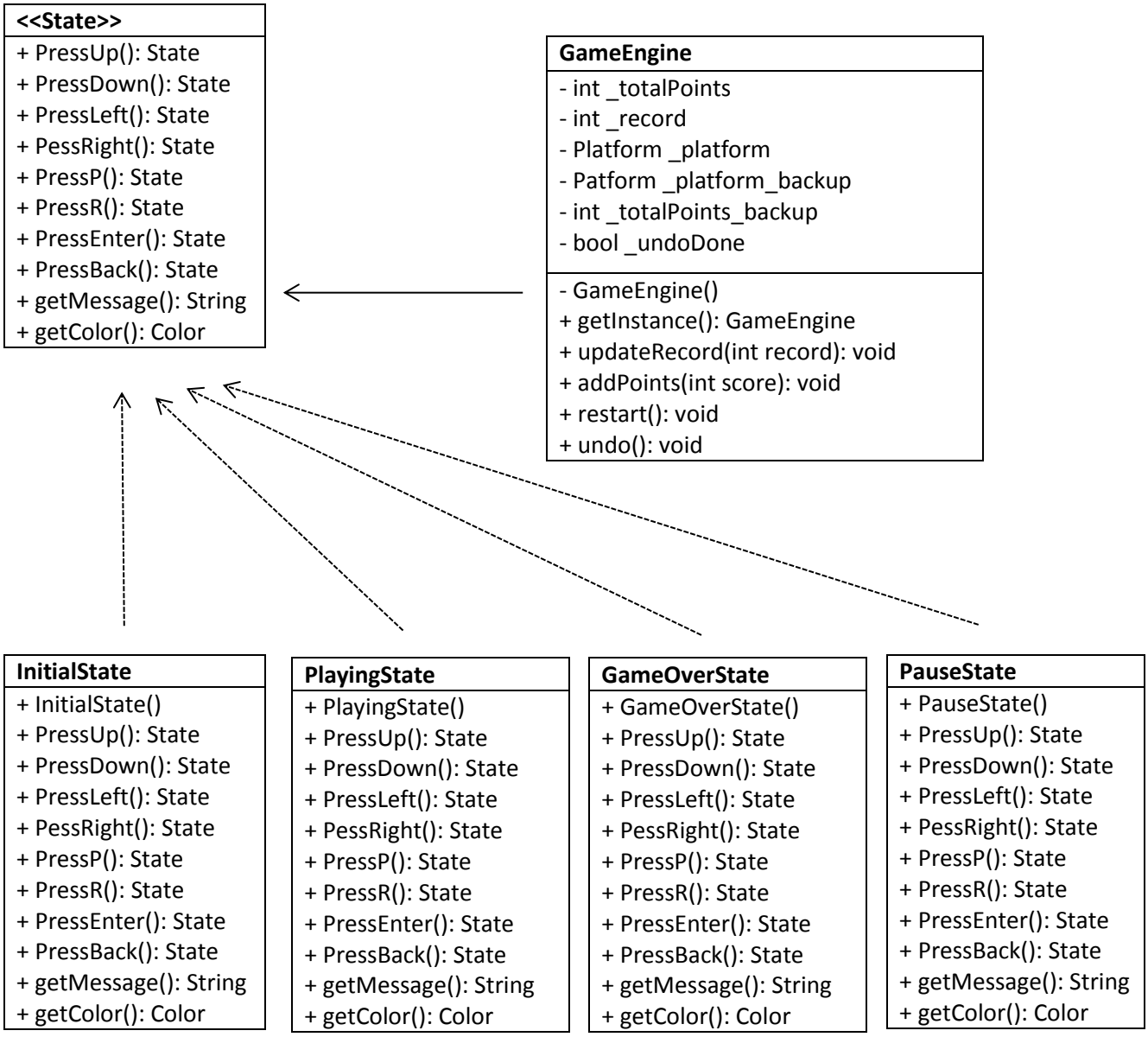
1. Write a natural language description of why and how the pattern is implemented in your code.

We have decided to implement the State Pattern to allow the game engine to change its behaviour when the internal state changes. We created an interface called State and four states implementing it: Initial, Playing, Pause and GameOver. Each state changes the game engine in a different way, according to its function.

Before using the state pattern, in Mainframe (class responsible for display user interface) we had a lot of edge cases, that were forcing to put a lot of if statements and making badly looked code.

After adding this pattern, the interface is state depended, so it’s only depend on the current state and doesn´t have to check a lot of flags.

2. Make a class diagram of how the pattern is structured statically in your code



3. Make a sequence diagram of how the pattern works dynamically in your code
4.