Memory P3 ADSOF

Pablo Sancho Saiz & Daniel Varela Sánchez

1.a What would happen if we let the state name change, once the object has been created?
It would happen that you could have two states with the same name, and that's not allowed.
How can we avoid it?
We could avoid this problem by checking if the name doesn't already exist in any other state.

1.b What did you do to avoid creating objects of type State outside of the stateMachine package?
We made the state class only accessible from that package by making the following constructor of the state class:
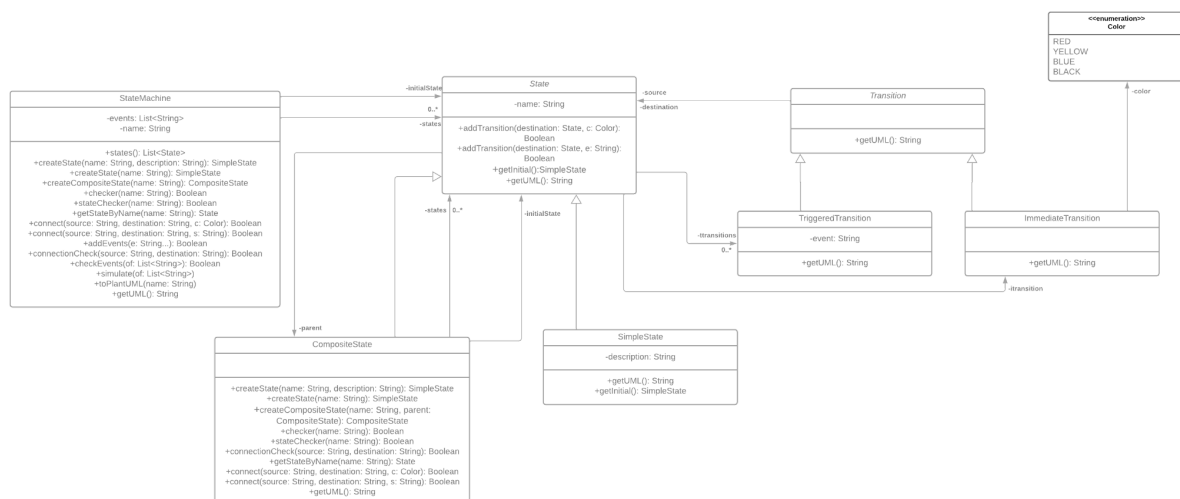
```
State (String name){
    this.name=name;
}
```

2.
In this design, what happens if we would allow connecting one state to another directly from the State class?
If we did that we would not be able to check if the state we are connecting from and the state we are connecting to exist in the same state machine.

Class Diagram and personal comments:

To design the whole assignment firstly we thought, since the states and the transitions could be of two different types, that we needed two general classes (State, Transition) that contain the shared attributes, between SimpleState and CompositeState, and also between TriggeredTransition and ImmediateTransition.

Then we created the StateMachine class, which contains all the states and events and it is the one that simulates all the assignment.

And the last class was the color enumeration that is required for the color of the ImmediateTransition class.

The main problems we had were mainly related with exercises 3 and 4. With the exercise 3 they were related with the simulate method of the StateMachine class, since we were traversing incorrectly through the Transitions, and because of that, through the states.

Exercise 4 was the most complicated one, we were forced to change the entire State class structure and their new relationship with their new attributes like states in the CompositeState class, and also the simulate method in the StateMachine class, to make it functional with the CompositeStates.

Now everything is working properly so you shouldn't have any problem executing each of the different sections.