

General Notes

- Comments within classes need to be corrected and modified.
- Conventions needed for class name, method name.
- Magic numbers.
- More States needed.
- there need more necessary parameters within functions and classes.
- **Interfaces**
 - ICollectable
 - N/A
 - ICollideable
 - NEXT SPRINT
 - ICommand
 - N/A
 - IController
 - Comments needed for class and methods
 - IDrawable
 - For the “ToggleDrawing” function, what/where is it toggling?
 - IEnemy
 - for the “ChangeDirection” function, the comment should be “call SwitchFaceDirection” state”.
 - IInteractable
 - for the “Interact” function, how does it interact with Mario?
 - IItem
 - N/A
 - IMoveStateMachine
 - N/A
 - IPlayer
 - Do we need UpdateScore in this sprint?
 - IProjectile
 - N/A
 - ISprite
 - N/A
 - IStructure
 - What additional function do we need to add in the interface? It seems bare there.
 - GetHitBox / GetPosition
 - IStructureStateMachine
 - N/A
 - IThinkingStateMachine
 - N/A
 - IUpdateable
 - for the “Update” function, should contain a parameter: gameTime
- **Classes**
 - BackgroundObject
 - NEXT SPRINT
 - BigMarioDecorator
 - should be set as a state

- Comments needed
- Block
 - Comments needed for function “Interact” and “Collision”
- BlockStateMachine
 - Why is the block’s initial position always 400?
 - Line 20 may create an alias of states and may not save the actual states correctly if states get changed.
 - Confusing why Update is included then not implemented.
- BlockSwitchingCommand
 - There should be only 1 switching command class to avoid Duplicated Code and similar classes.
- EnemySwitchingCommand
 - See BlockSwitchingCommand
- FireballStateMachine
 - Similar to ShellStateMachine, There should be only 1 StateMachine to avoid Duplicated Code and similar classes.
- FireballUpdate
 - There could be more collision conditions for fireball, such as interacting with items or players.
- FireMarioDecorator
 - should be set as a state
 - Comments needed
- Flag
 - Comments needed
- FlagStateMachine
 - Comments needed.
 - NEXT SPRINT.
- GameObjectManager
 - There should be only 1 object class to avoid Duplicated Code and similar classes.
 - The StateMachines for most of the Objects should have a Reset() function that resets the initial position already, as well as their states. They also have their very initial positions hardcoded in there. Do we want the GameObjectManager to send the initial conditions to the statemachines for this sprint?
 - When creating the objects, we need to make sure their constructors get passed the correct things they need. such as gametime, initial location.
- GameState
 - NEXT SPRINT.
- Goomba
 - N/A
- GoombaStateMachine
 - Update seems a little long and overly complex. Could remove “FaceRight” or “FaceLeft” state.
 - There are no references between GoombaStateMachine and GoombaUpdate.
- GoombaUpdate

- There are no references between GoombaStateMachine and GoombaUpdate.
- ItemStateMachine
 - Update seems a little long and overly complex. Could remove “FaceRight” or “FaceLeft” state.
 - There are no references between ItemStateMachine and ItemUpdate.
- ItemSwitchingCommand
 - See BlockSwitchingCommand
- ItemUpdate
 - There are no references between ItemStateMachine and ItemUpdate.
- KeyboardController
 - N/A
- Koopa
 - N/A
- KoopaStateMachine
 - Update seems a little long and overly complex. Could remove “FaceRight” or “FaceLeft” state.
 - There are no references between KoopaStateMachine and KoopaUpdate.
- KoopaUpdate
 - There are no references between KoopaStateMachine and KoopaUpdate.
- LittleMarioDecorator
 - should be set as a state
 - Comments needed
- Mario
 - Mario needs more StateMachines: like hState.FunctionName() (Health State) or vState.FunctionName() (Vulnerability State).
 - More implementation needed within Mario functions.
 - Input vector location parameter needed.
- MarioJumpCommand
 - If the space bar is held down, will that continue to execute this command continuously? Or will it activate only once? This affects the statemachine and how it updates Mario’s position on screen.
- MarioLeftCommand
 - See MarioJumpCommand
- MarioRightCommand
 - See MarioJumpCommand
- MarioStateMachine
 - Update seems a little long and overly complex. Could remove “FaceRight” or “FaceLeft” state.
 - Magic numbers
 - Where better set UpdateScore function?
- MarioThrowingCommand
 - See MarioJumpCommand
- MouseController
 - N/A
- Pipe

- N/A
- PipeStateMachine
 - NEXT SPRINT.
- QuitGameCommand
 - N/A
- Shell
 - Magic numbers
 - The speed shouldn't be needed within Move() and Collision()..
- ShellStateMachine
 - See FireballStateMachine above.
- ShellUpdate
 - Comments needed within function.
 - There are no references between ShellStateMachine and ShellUpdate.
- Sprite
 - Is it at all possible to combine the 2nd and 3rd constructors into 1 with the same parameters as the 2nd? And to hold up what the 3rd constructor does we would just have to send in SpriteEffects.None into that last parameter.
- SpriteFactory
 - Quite a long class, is there any way we could simplify/shorten it?
 - I would think changing "HitBlock" to something like "UsedBlock", "FinishedBlock", or "DoneBlock" would help with clarity.
- StarMarioDecorator
 - should be set as a state
 - Comment needed
- Text
 - N/A
- ResetCommand
 - N/A