**Explain how each state class’s methods should be implemented.**

Super Mario game have an interface “MarioState” with four interface methods. There will

be three state class methods:

**1. Standing**

**2. Ducking**

**3. Jumping**

The state class methods will implement the methods of the interface.

**We implement the standing class and interface method as below:**

public class Standing implements MarioState

{

private static Standing instance;

//getInstance()

public static Standing getInstance()

{

if(instance==null)

instance = new Standing();

return instance;

}

Mario marioState;

// Standing

Public Standing() {}

@Override

//method dKeyPushed() implementation

Public void duck()

{

marioState.dKeyReleased();

marioState.setState(newState.getInstance());

}

//similarly rest of the methods of the interface are implemented.

}

**Standing Class**

Methods

**1. duck()**

- Set state.duck(…) to True

- Set state.stand(…) to False

- Set state.jump(…) to False as jumping is not allowed while ducking

-move() method can be called

-dKeyReleased can be called, followed by stand().

**2. stand()**

-State.stand(…) will be true

- Calling stand() will not make any changes.

-Set State.jump(…) to true as jumping is allowed during standing.

-move() can be called

-dKeyPushed() or sKeyPushed() can be called, followed by duck() or jump() respectively

**3. move()**

-Set state.move() to true.

-jump() or duck() can be called, followed by respective change of states.

**4. jump()**

-Set state.jump() to true

-dkeyPushed() can be called to make Mario Dive.

-move() can be called as moving is allowed while jumping

**Ducking Class**

Methods

**1. move()**

-set state.move() to true

-dKeyReleased() can be called, followed by stand()

**2. duck()**

-state.duck() will be true.

-set state.jump() to false

- Calling duck() will not make any changes.

-move() can be called

-dKeyReleased() can be called, followed by stand()

**3. stand()**

-set state.duck() to false

-set state.stand() to true

-move(), jump() or duck() can be called, followed by change of states.

**4. jump()**

-state.duck() will be true

-state.jump() will be false.

Hence jump() cannot be called.

**Jumping Class**

Methods

**1. move()**

-set state.move() to true

**2. duck()**

-set state.duck() to true.

-Mario will Dive

3. **stand()**

-set state.jump() to false

-set state.stand() to true

-move(), jump() or duck() can be called, followed by respective change of states.

4. **jump()**

-state.jump() will be true

- Calling jump() will not make any changes.

-move() can be called, followed by respective change of state.