Name: [redacted] Section (period): [redacted]

6 points each (min 50 if attempted):

1. Finite state machines are basically flowcharts, with a few key differences.
   a. 0 (false)
   (b.) 1 (true) 1

2. There is no base state interface or base state class needed in order to create concrete state classes in the state pattern. need base state
   (a.) 0 (false) 0
   b. 1 (true)

3. The State Pattern ensures that there is only one instance of a class created.
   (a.) 0 (false) 0 singleton
   b. 1 (true)

4. Two major variants of the State Pattern are to have states suggest their own transitions, or to have some sort of StateManager that holds a table of transitions.
   a. 0 (false) 1 the entire pattern is fsm
   (b.) 1 (true) fsm can be stored in object

5. The State Pattern allows us to encapsulate varying behavior for the same object, based on its internal state.
   a. 0 (false) 1
   (b.) 1 (true)

6. An enum representing state is a low-key way to get *some* of the benefits of the state pattern.
   a. 0 (false) 1 doesn't do transitions but still good
   (b.) 1 (true)

7. Unity has a built-in state manager called StateMachineBehavior.
   a. 0 (false) 1 already been implemented
   (b) 1 (true)

8. The Hierarchical State Pattern is a variation wherein every concrete state has the same parent State class.
   a. 0 (false) 1 polymorphism
   (b.) 1 (true)

9. Extra credit to go from 98 to 100, using #1 as the most significant bit and #8 as the least significant bit, what unsigned int do the answers to this test represent?

= 10011111 $1+2+4+8+16+128 = \boxed{159}$

$= 1\cdot2^0 + 1\cdot2^1 + 1\cdot2^2 + 1\cdot2^3 + 1\cdot2^4 + 1\cdot2^7$

2
128
16
8
4
2
+ 1
159