1.23: Some questions imply some ambiguity, or use both sequential and combinational. Points are given for correctly explaining rationale if you disagree with the solution, and points are lost for no explanation at all, regardless of “correctness”.

a. sequential – requires memorizing states of the machine (counter for “preset time”)
b. combinational – takes inputs, provides outputs, no memory required
c. sequential – needs a queue to make sure coins are outputted in order
d. sequential – needs to keep track of current time, needs to keep track of preset time.
e. combinational – just comparing two numbers and outputting 1 or 0. no memory
f. combinational – requires no memory of previous state, just current input state
g. sequential – needs to keep track of even/odd, and takes input one at a time (register/FF)
h. combinational – no memory of previous state, just wiring and combinational logic

1.31

a) inputs: direction of spin (counter or clock), current number being pointed to, “passed 0” bit (useful for both reset and 2nd number)
   outputs: locked, unlocked, (reset)
   Note: Reset isn’t really an “output” persay, more a combination of inputs. See FSM.
b) Drawing: I don’t have access to Visio on my Mac, so the state machine is drawn in a “Paint” equivalent. Hope it’s decently understandable!

2.2 and 2.3 – Should be straightforward, no drawings will be attached