1.) Simplify the following K-maps, and write the corresponding Boolean Equation. You should create the simplest Sum of Products form possible.

a.)

b.)
2.) Using the rules of Boolean Algebra, minimize this function. You do not need to list the rules, but you must show the steps you use.

\[ F = (B \cdot (CDE + D \cdot (\overline{C} + \overline{E}))) \cdot (\overline{AB} + BC) \]
3.) Draw the state diagram for the following circuit. YOU DO NOT HAVE TO DO THE STATE TABLE. YOU DO NOT HAVE TO IMPLEMENT THE CIRCUIT. Your state diagram should be as simple as possible.

Your machine has two inputs: A and B. Your machine should output TRUE once it has seen at least one 1 on input A, and at least one 1 on input B, and continue outputting TRUE thereafter. The 1’s can be seen on the same or different clock cycles. The output reacts as soon as you see the proper pattern, including the current inputs to the circuit.
4.) For the following state diagram, implement the circuit. The state encoding is given to you in each state. You may use premade D-flipflops and any other basic gates. Your implementation should be as simple as possible.
5.) For the following circuit, fill in the timing diagram below. All DFFs are positive-edge-triggered, like we’ve been using in class. Note that the delay for any gate is very small compared to the grid lines in the timing diagram, and you can assume $T_{\text{setup}}$, Clk->Q, and $T_{\text{hold}}$ are 0.

Assume that the value of all DFFs starts at 0.