1. (100 pts.) Regularize this.

For each of the following languages over the alphabet \{0, 1\}, give a regular expression that describes that language, and briefly argue why your expression is correct.

1.A. All strings except 010 and 101.
1.B. All strings that start in 10 and contain 111 as a substring.
1.C. All strings in which every nonempty maximal substring of consecutive 1s is of even length. For instance 10011 is not in the language while 01111000110 is.
1.D. All strings that do not contain the substring 010.
1.E. All strings that do not contain the subsequence 101.

2. (100 pts.) Then, shalt thou count to three.

Let \( L \) be the set of all strings in \( \{0, 1\}^* \) that contain at least three occurrences of the substring 001.

2.A. Describe a DFA that over the alphabet \( \Sigma = \{0, 1\} \) that accepts the language \( L \). Argue that your machine accepts every string in \( L \) and nothing else, by explaining what each state in your DFA means.

You may either draw the DFA or describe it formally, but the states \( Q \), the start state \( s \), the accepting states \( A \), and the transition function \( \delta \) must be clearly specified.

2.B. Give a regular expression for \( L \), and briefly argue why the expression is correct.

3. (100 pts.) Do not affix it.

(This exercise is about writing things formally – it is not difficult once you have cut through the formalism. In short, don’t panic - you can do it!)

3.A. Let \( M = (Q, \Sigma, \delta, s, A) \) be a DFA. A state \( q \in Q \) is bad, if for all strings \( w \in \Sigma^* \) we have that \( \delta^*(q, w) \notin A \). Let \( B(M) \subseteq Q \) be the set of bad states of \( M \). Consider the DFA \( M' = (Q, \Sigma, \delta, s, B(M)) \). What is the language \( L(M') \)? Prove formally your answer!

3.B. Prove that if \( x \in L(M') \) and \( y \in \Sigma^* \), then \( xy \in L(M') \).
3.C. Let \( L_1 \) and \( L_2 \) be two regular languages over \( \Sigma \) accepted by DFAs \( M_1 = (Q_1, \Sigma, \delta_1, s_1, A_1) \),
and \( M_2 = (Q_2, \Sigma, \delta_2, s_2, A_2) \), respectively.
Describe a DFA \( M = (Q, \Sigma, \delta, s, A) \) in terms of \( M_1 \) and \( M_2 \) that accepts
\[
L = \{ w \mid w \in L_2 \text{ and no prefix of } w \text{ is in } L_1 \}
\]
Formally specify the components \( Q, \delta, s, \) and \( A \) for \( M \) in terms of the components of \( M_1 \) and \( M_2 \).