CS 254: Computability and Complexity	**Anonymous submission**
Problem Set #03	*October 1, 2019*

1. We have a DFA for language A and a DFA for language B

   Want to show: language C

   "perfect shuffle": { w|w = $a_1b_1...a_kb_k$, where $a_1...a_k \in$ A $b_1...b_k \in$ B, each $a_ib_i \in \sum^*$}

   D_A = (Q_A, $\sum$, δ_A q_A, F_A) and D_B = (Q_B, $\sum$, δ_B, q_B, F_B)

   To create this DFA, call it D_C I need to construct one that essentially switches between D_A and D_B. To do this D_C will need to keep track of the current state in D_A and D_B as well as whether the character in the string should advance D_A or D_B. The accept state for D_C is if D_A and D_B are both in accept states.

   D_C is defined as follows:

   Q = Q_A x Q_B x $\{A$, B$\}$ This will track the current state for both D_A and D_B as well as whether the character in the string should advance D_A or D_B

   q = (q_A q_B, A) So D_C starts at q_A applies the first character then switches to q_B applies the next and then the following character should be in A

   $\sum = \sum_A \cup \sum_B$ as it can accept the alphabet of either

   δ((x,y,A),i) = (δ_A(x,i),y,B) If the current state of D_A is x while the current state of D_B is y and the next character to be read is i we will change to current state in D_A to δ_A(x,i).

   Following similar logic: δ((x,y,B),i) = (x, δ_B(y,i),A) if i is read into D_B the next state for D_B is δ_B(y,i) and the next character will be read will change D_A

   F = F_A x F_B shows that if we are in an accept state for both D_A and D_B then D_C accepts

   Since we have defined a DFA for C, C is then normal.