5. For languages A and B, let the *perfect shuffle* of A and B be the language

$$\{w \mid w = a_1b_1 \cdots a_kb_k, \text{ where } a_1 \cdots a_k \in A \text{ and } b_1 \cdots b_k \in B, \text{ each } a_i, b_i \in \Sigma\}.$$

Show that the class of regular languages is closed under perfect shuffle.

Answer: Let $D_A = (Q_A, \Sigma, \delta_A, q_A, F_A)$ and $D_B = (Q_B, \Sigma, \delta_B, q_B, F_B)$ be two DFAs that recognize A and B, respectively. Here, we shall construct a DFA $D = (Q, \Sigma, \delta, q, F)$ that recognizes the perfect shuffle of A and B.

The key idea is to design D to alternately switch from running D_A and running D_B after each character is read. Therefore, at any time, D needs to keep track of (i) the current states of D_A and D_B and (ii) whether the next character of the input string should be matched in D_A or in D_B. Then, when a character is read, depending on which DFA should match the character, D makes a move in the corresponding DFA accordingly. After the whole string is processed, if both DFAs are in the accept states, the input string is accepted; otherwise, the input string is rejected.

Formally, the DFA D can be defined as follows:

(a) $Q = Q_A \times Q_B \times \{A, B\}$, which keeps track of all possible current states of D_A and D_B, and which DFA to match.

(b) $q = (q_A, q_B, A)$, which states that D starts with D_A in q_A, D_B in q_B, and the next character read should be in D_A.

(c) $F = F_A \times F_B \times \{A\}$, which states that D accepts the string if both D_A and D_B are in accept states, and the next character read should be in D_A (i.e., last character was read in D_B).

(d) δ is as follows:

i. $\delta((x, y, A), a) = (\delta_A(x, a), y, B)$, which states that if current state of D_A is x, the current state of D_B is y, and the next character read is in D_A, then when a is read as the next character, we should change the current state of A to $\delta_A(x, a)$, while the current state of B is not changed, and the next character read will be in D_B.

ii. Similarly, $\delta((x, y, B), b) = (x, \delta_B(y, b), A)$.

6. For languages A and B, let the *shuffle* of A and B be the language

$$\{w \mid w = a_1 b_1 \cdots a_k b_k, \text{ where } a_1 \cdots a_k \in A \text{ and } b_1 \cdots b_k \in B, \text{ each } a_i, b_i \in \Sigma^*\}.$$

Show that the class of regular languages is closed under shuffle.

Answer: Let $D_A = (Q_A, \Sigma, \delta_A, q_A, F_A)$ and $D_B = (Q_B, \Sigma, \delta_B, q_B, F_B)$ be two DFAs that recognize A and B, respectively. Similar to the previous question, we shall prove by construction. However, the key difference is that D *may* now switch from running D_A and running D_B after each character is read. To allow this flexibility and simplify the construction, we design an NFA $N = (Q, \Sigma, \delta, q, F)$ that recognizes the shuffle of A and B instead of directly designing a DFA.

At any time, N needs to keep track of the current states of D_A and D_B. Then, when a character is read, N may make a move in D_A or D_B accordingly. After the whole string is processed, if both DFAs are in the accept states, the input string is accepted; otherwise, the input string is rejected. In addition, N should also accept the empty string.

Formally, the NFA N can be defined as follows:

(a) $Q = (Q_A \times Q_B) \cup \{q_0\}$, where $Q_A \times Q_B$ keeps track of all possible current states of D_A and D_B, and q_0 denotes the state when nothing is read.

(b) $q = q_0$.

(c) $F = (F_A \times F_B) \cup \{q_0\}$, which states that N accepts the string if both D_A and D_B are in accept states, or N accepts the empty string.

(d) δ is as follows:

   i. $\delta(q_0, \varepsilon) = (q_A, q_B)$, which states that at the start state q_0, N can make D_A in q_A and D_B in q_B without reading anything.

   ii. $(\delta_A(x, a), y) \in \delta((x, y), a)$, which states that if current state of D_A is x, the current state of D_B is y, then when a is read as the next character, we can change the current state of A to $\delta_A(x, a)$, while the current state of B is not changed.

   iii. Similarly, $(x, \delta_B(y, a)) \in \delta((x, y), a)$.