Problem

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.

Step-by-step solution

Step 1 of 2

Consider the two languages *A* and *B*. The language *shuffle* on *A* and *B* is as follows:
$\{w \mid w = a_1 b_1 ... a_k b_k, \text{ where } a_1 ... a_k \in A \text{ and } b_1 ... b_k \in B, \text{ each } a_i, b_i \in \Sigma^*\}$.

Assume, $DFA_A = (Q_A, \Sigma, \delta_A, S_A, F_A)$ and $DFA_B = (Q_B, \Sigma, \delta_B, S_B, F_B)$ be two DFAs that recognize *A* and *B* respectively. $DFA_{shuffle} = (Q, \Sigma, \delta, S, F)$ recognizes the language perfect shuffle on *A* and *B*. For each character read, $DFA_{shuffle}$ may move from running DFA_A to running DFA_B. The *NFA* is more flexible when compared to the *DFA*. In this case, $NFA_{shuffle} = (Q, \Sigma, \delta, S, F)$ has to be constructed to allow more flexibility.

The $NFA_{shuffle}$ keeps track the current states of DFA_A and DFA_B. For each character read, $NFA_{shuffle}$ makes moves in the corresponding DFA (either DFA_A or DFA_B). After the whole string is read, if both DFA_A and DFA_B reaches to the final state, then the input string is accepted by $NFA_{shuffle}$.

Comment

Step 2 of 2

The $NFA_{shuffle}$ can be defined as follows:

- $Q = (Q_A \times Q_B) \cup \{q_0\}$ *:* The set of all possible states of DFA_A and DFA_B which should match with $NFA_{shuffle}$. Here, q_0 denotes the initial state.
- $q = q_0$
- $F = (F_A \times F_B) \cup \{q_0\}$: F_A and F_B are the final states for DFA_A and DFA_B respectively. The $NFA_{shuffle}$ accepts the string if both DFA_A and DFA_B are in accept states or $NFA_{shuffle}$ accepts the empty string.
- δ is as follows:

o $\delta(q_0, \varepsilon) = (q_A, q_B)$: At the start state q_0, the current state of DFA_A is q_A and the current state of DFA_B is q_B without reading anything.

o $(\delta_A(m,a), n) \in \delta((m,n), a)$: Change the current state of *A* to $\delta_A(m,a)$ when the character *a* is read. Here, the current state of D_A is *m* and the current state of D_B is *n*.

o $(m, \delta_B(n,a)) \in \delta((m,n), a)$: Change the current state of *B* to $\delta_B(n,a)$ when the character *a* is read. Here, the current state of D_A is *m* and the current state of D_B is *n*.

The language *L* is said to be regular if there exist an FA that recognizes the language *L*. Here, the $NFA_{shuffle}$ is defined for the language *shuffle*.

Therefore, the class of regular languages is closed under shuffle.

Comment