Problem of the Week – 1

Half-Paths-NFA

Define a Half-Paths-NFA M as a tuple $(Q, \Sigma, \delta, q_0, F)$ with the following properties: Q is a finite set of states, Σ is the (finite) input alphabet, δ is a function from $Q \times \Sigma$ to the set $P(Q) - \{\emptyset\}$, $q_0 \in Q$ is the initial state, and $F \subseteq Q$ is a set of final states. (Recall that $P(q)$ is the power set of Q.) Thus, the transition function of M is the same as that of an NFA in which there are no ε-transitions and at least one transition is defined for every state and every symbol of the input alphabet.

We now define what it means for M to accept an input string w. A *computation of M on input string* $w = w_1 w_2 \cdots w_n$, where $w_i \in \Sigma$ for all i, is a sequence of states $q_0, q_1, \ldots, q_n$ such that $q_i \in \delta(q_{i-1}, w_i)$ for $1 \leq i \leq n$; the computation is said to be an *accepting* computation if $q_n \in F$. The Half-Paths-NFA M is said to *accept* a string w if and only if at least half of the computations of M on w are accepting.

Put another way, a Half-Paths-NFA is an NFA with transitions for every state and input symbol and with no ε-transitions, that accepts only those strings for which at least half of the computation paths lead to a final state.

Compare the power of Half-Paths-NFAs with that of NFAs. That is, determine which of the following four cases is true: the class of languages recognized by Half-Paths-NFAs is the *same as*, or *a subset of*, or *a superset of*, or *incomparable with* the class of regular languages.