Deterministic Finite Automata (DFA)

$M = (Q, \Sigma, \delta, q_0, F)$:

Q is a finite set (states)

Σ is a finite set (alphabet)

$\delta : Q \times \Sigma \rightarrow Q$ (transition function)

q_0: distinguished state (start state)

$F \subseteq Q$ (accepting/final states)

Notation: $\hat{\delta} : Q \times \Sigma^* \rightarrow Q$ defined as follows:

- $\hat{\delta}(q, \epsilon) = q$
- $\hat{\delta}(q, ua) = \delta(\hat{\delta}(q, u), a)$.

Regular Languages

Let M be a DFA. A string $w = w_1 w_2 \cdots w_n$, where each $w_i \in \Sigma$ is *accepted* by M if there exists a sequence of states $r_0, r_1, \cdots, r_n$ such that:

- (start right) $r_0 = q_0$,
- (move right) for all $0 \leq i \leq n - 1$, $\delta(r_i, w_{i+1}) = r_{i+1}$, and
- (finish right) $r_n \in F$.

M *recognizes* language A if $A = \{w \mid M \text{ accepts } w\}$.

A language is *regular* if and only if a DFA recognizes it. We denote the language of a DFA M as $L(M)$.