**Alphabet:** Finite, non-empty set of distinct symbols, denoted $\Sigma$.

e.g. $\Sigma = \{0, 1\}$ (binary alphabet)
$\Sigma = \{a, b, c, \ldots, z\}$ (lower-case letters alphabet)

**String:** (also called word) finite (ordered) sequence of symbols chosen from some alphabet.

e.g. 010011101011

Empty string: $\varepsilon$, $|\varepsilon| = 0$, is in any $\Sigma$ by default.

$|w|$ length of the string $w$.

e.g. $|010011101011| = 12$

$\Sigma^k$ is the set of strings over $\Sigma$ of length *exactly* $k$.

e.g. If $\Sigma = \{0, 1\}$, then $\Sigma^0 = \{\varepsilon\}$, $\Sigma^1 = \Sigma$, $\Sigma^2 = \{00, 01, 10, 11\}$, etc.

How many strings are in $\Sigma^k$?

**Kleene's Star:** $\Sigma^*$ is the set of all stings over $\Sigma$.

$$\Sigma^* = \Sigma^0 \cup \underbrace{\Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \ldots}_{=\Sigma^+}$$

**Concatenation:** Let $x, y$ be strings.

If $x = a_1 a_2 \ldots a_m$ and $y = b_1 b_2 \ldots b_n$, then

$xy = a_1 a_2 \ldots a_m b_1 b_2 \ldots b_n$

Note $w\varepsilon = \varepsilon w = w$.

**Language:** A collection of strings. $L$ is a language over the alphabet $\Sigma$ if $L \subseteq \Sigma^*$.

e.g., $L = \{\varepsilon, 01, 0011, 000111, \ldots\}$

Note that $\{\varepsilon\} \neq \emptyset$; one is the language consisting of the single string $\varepsilon$, and the other is the empty language.

**Problem:** Is a given string in a given language?

e.g. $L = \{\underbrace{10}_{2}, \underbrace{11}_{3}, \underbrace{101}_{5}, \underbrace{111}_{7}, \ldots\}$.

$L \subseteq \{0,1\}^*$

$w \in L$ iff $w \in \{0,1\}^*$ encodes a prime number in binary notation.

**Deterministic Finite Automaton (DFA):**

1. Finite set of states $Q$
2. Finite set of input symbols $\Sigma$
3. Transition function $\delta : Q \times \Sigma \longrightarrow Q$
4. Start state $q_0$
5. A set of final (accepting) states, $F$.

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

Given $q, a$, $\delta(q, a) = p$.

To see whether $M$ accepts a string $w$, we "run" $M$ on $w = a_1 a_2 \ldots a_n$ as follows:

$\delta(q_0, a_1) = q_1$, $\delta(q_1, a_2) = q_2$, until $\delta(q_{n-1}, a_n) = q_n$.

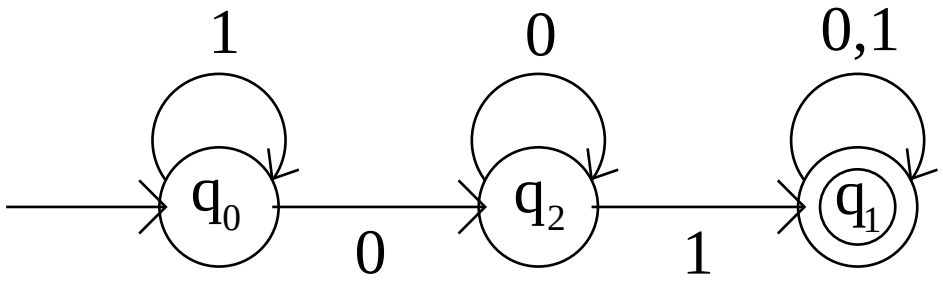
Accept iff $q_n \in F$.

Consider $L = \{w|\ w$ is of the form $x01y \in \Sigma^*\ \}$ where $\Sigma = \{0, 1\}$.

We want to specify a DFA $M = (Q, \Sigma, \delta, q_0, F)$ that accepts all and only the strings in $L$.

$\Sigma = \{0, 1\}$, $Q = \{q_0, q_1, q_2\}$, and $F = \{q_1\}$.

**Transition Diagram:**




**Transition Table:**

| | 0 | 1 |
|---|---|---|
| $q_0$ | $q_2$ | $q_0$ |
| $q_1$ | $q_1$ | $q_1$ |
| $q_2$ | $q_2$ | $q_1$ |

**Extended Transition Function:** Given $\delta$, its ETF is $\hat{\delta}$ defined inductively:

Basis Case: $\hat{\delta}(q, \varepsilon) = q$

Induction Step: if $w = xa$, $w, x \in \Sigma^*$ and $a \in \Sigma$, then

$$\hat{\delta}(q, w) = \hat{\delta}(q, xa) = \delta(\hat{\delta}(q, x), a)$$

Thus: $\hat{\delta} : Q \times \Sigma^* \longrightarrow Q$.

$w \in L(M) \iff \hat{\delta}(q_0, w) \in F$

Here $L(M)$ is the set of all those strings (and only those) which are accepted by $M$.

**Language of a DFA:** $L(M) = \{w | \hat{\delta}(q_0, w) \in F\}$.

**Regular Languages:** $L$ is *regular* iff there exists a DFA $M$ such that $L = L(M)$.

We say that $M$ *recognizes* a language $A$ iff $L(M) = A$.

Let $A, B$ be languages.

**Union:** $A \cup B = \{x | x \in A \text{ or } x \in B\}$

**Concatenation:** $A \circ B = \{xy | x \in A \text{ and } y \in B\}$

**Star:** $A^* = \{x_1 x_2 \ldots x_k | k \geq 0 \text{ and each } x_i \in A\}$

Regular Languages closed under all three.

Union: $Q = Q_1 \times Q_2$ and $\delta((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))$.