REs:

A *regular expression* or **RE** R (over some alphabet Σ) is one of:

- $\varnothing$ (the *null* RE);
- ε (the *empty* RE);
- a (a *literal*, for any $a \in \Sigma$);
- $R_1 \mid R_2$ (the *union* of REs R_1 and R_2);
- $R_1 \cdot R_2$ (the *concatenation* of REs R_1 and R_2); or
- $R_1{}^*$ (the *Kleene star* or *iteration* of RE R_1).

An **RE** R (over Σ) *matches* a string $w = x_1 \ldots x_n$ (where each $x_i \in \Sigma$) iff:

- $R \equiv \varnothing$ and … well, uhh, then it just *doesn't* match, ever: oh well!
- $R \equiv \varepsilon$ and n=0 (i.e., w is the empty string);
- $R \equiv a$ and n=1, with $x_1 \equiv a$;
- $R \equiv R_1 \mid R_2$ and either R_1 matches w or R_2 matches w;
- $R \equiv R_1 \cdot R_2$ and w can be split $w = w_1 w_2$ so that both R_1 matches w_1 **and** R_2 matches w_2; or
- $R \equiv R_1{}^*$ and either n=0 (i.e., w is the empty string) or
  w can be split $w = w_1 w_2$ so that both R_1 matches w_1 **and** $R_1{}^*$ matches w_2.

DFAs:

A *deterministic finite automaton* or **DFA** $M = \langle Q, \Sigma, \delta, q_0, F\rangle$ where

- Q is a finite set of *states;*
- Σ is an *alphabet* (finite set of symbols);
- $\delta \in Q \times \Sigma \rightarrow Q$ is called the *transition function;*
- $q_0 \in Q$ is the *initial state;* and
- $F \subseteq Q$ is the set of *final states.*

A DFA M *accepts* a string $w = x_1 \ldots x_n$ (where each $x_i \in \Sigma$) iff
there are states $q_1, \ldots, q_n \in Q$ such that:

- $q_{i+1} = \delta(q_i, x_{i+1})$ for every i, $0 \leq i < n$; and
- $q_n \in F$.

The *language of a DFA* $L(M) = \{ w \in \Sigma^* \mid M \text{ accepts } w \}$.

NFAs:

A *non-deterministic finite automaton* or **NFA** $N = \langle Q, \Sigma, \delta, q_0, F\rangle$ where

- Q is a finite set of *states;*
- Σ is an *alphabet* (finite set of symbols);
- $\delta \in Q \times \Sigma_\varepsilon \rightarrow \wp(Q)$ is a transition function (where $\Sigma_\varepsilon \triangleq \Sigma \cup \{\varepsilon\}$);
- $q_0 \in Q$ is the *initial state;* and
- $F \subseteq Q$ is the set of *final states.*

An NFA N *accepts* a string $w = x_1 \ldots x_n$ (where each $x_i \in \Sigma$) iff
w can be written as $w = y_1 \ldots y_m$ where each $y_i \in \Sigma_\varepsilon$ and there are states $q_1, \ldots, q_m \in Q$ such that:

- $q_{i+1} \in \delta(q_i, x_{i+1})$ for every i, $0 \leq i < m$; and
- $q_m \in F$.

The *language of an NFA* $L(N) = \{ w \in \Sigma^* \mid N \text{ accepts } w \}$.

GNFAs:

A *generalized non-deterministic finite automaton* or **GNFA** N = $\langle Q, \Sigma, \delta, q_i, q_f \rangle$ where

- Q is a finite set of *states;*
- Σ is an *alphabet* (finite set of symbols);
- $\delta \in (Q - q_i) \times (Q - q_f) \rightarrow$ **RE** is a transition function;
- $q_i \in Q$ is the *initial state;* and
- $q_f \in Q$ is the *final state.*

A GNFA N *accepts* a string $w = w_0 \ldots w_n$ (where each $w_i \in \Sigma^*$) iff there are states $q_0, \ldots, q_n \in Q$ such that:

- $q_0 = q_i$;
- $q_n = q_f$; and
- $w_i \in L(\delta(q_i, q_{i+1}))$ for every i, $0 \leq i < n$.

The *language of a GNFA* $L(N) = \{ w \in \Sigma^* \mid N \text{ accepts } w \}$.