A formalism for describing a certain class of languages.

**Definition:** given an alphabet \( \Sigma \), regular expressions are
strings over the alphabet \( \Sigma \cup \{ +, \cdot, ( ), \epsilon, \emptyset \} \) defined
inductively as follows:

- **basis:** \( \epsilon, \emptyset \), and each \( a \in \Sigma \) is a R.E.
- **inductive step:** if \( E \) and \( F \) are R.E., then so are:
  - \( E + F \) (union)
  - \( E \cdot F \) (concatenation)
  - \( E^* \) (closure)
  - \( (E) \) (parentheses)

**Example:** \( a \cdot (a + b)^* \cdot b^* \cdot a \)

**Definition:** language \( L(E) \) defined by a R.E. \( E \)
in also defined inductively:

- \( L(\epsilon) = \{ \} \) empty word
- \( L(\emptyset) = \emptyset \) empty language
- \( L(E + F) = L(E) \cup L(F) \)
- \( L\left( (E) \right) = L(E) \)
- \( L(E \cdot F) = L(E) \cdot L(F) \) concatenation

**Example:** \( E = \epsilon + 1 \) \( \Rightarrow L(E) = \{ \epsilon, 1 \} \)
\( F = \epsilon + 0 + 1 \) \( \Rightarrow L(F) = \{ \epsilon, 0, 1 \} \)
\( G = E \cdot F \) \( \Rightarrow L(G) = \{ \epsilon, 0, 1, 10, 11 \} \)
\( = (\epsilon + 1) \cdot (\epsilon + 0 + 1) \)
closure of a language \( L \) ?
we first define the powers of a language \( L \):
- \( L^0 = \{ \varepsilon \} \)
- \( L^k = L^{k-1} \cdot L \)

Hence \( L^k = \{ w \mid w = \varepsilon, \ldots, \varepsilon \text{ with } \forall i, \pi_i \in L \} \)

closure of \( L \):
\[
L^* = L^0 \cup L^1 \cup L^2 \cup \ldots = \bigcup_{k \geq 0} L^k
\]

Example:
- \( E = 0 + 1 \) \( \Rightarrow S(E) = \{0, 1\} \)
- \( F = E^* \) \( \Rightarrow S(F) = \) set of all binary strings
- \( E = 0.0 \) \( \Rightarrow S(E^*) = \{ \varepsilon, 00, 0000, 000000, \ldots \} \)
  = all even-length strings of 0's

Positive closure of a language \( L \)

\[
L^* = L^0 \cup L^1 \cup L^2 \cup \ldots
\]

We can introduce a positive closure operator in R.E.

\[
S(E^*) = (S(E))^*
\]

Note: we have to distinguish between an expression \( E \) and the language \( S(E) \) defined by \( E \)

When we write \( E = F \), we usually mean not syntactic equality, but equality of the corresponding languages, i.e. \( S(E) = S(F) \).

In other words, equality is in the algebra of R.E.

Precedence of operators:

\[
\text{high} \quad \ast \\
\quad \cdot \\
\text{low} \quad \downarrow \\
\quad +
\]

Example:

\( E + F \cdot G^* = E + (F \cdot (G^*)) \)
Algebraic laws for R.E.

Similar to the laws for arithmetic expressions, we can express laws for R.E.:

- As addition, we have:
  \[(E \cdot F) \cdot G = E \cdot (F \cdot G) = E \cdot F \cdot G\]
  \[(E + F) + G = E + (F + G) = E + F + G\]

- As commutativity of +
  \[E + F = F + E\]

Note: * is not commutative: \[E \cdot F \neq F \cdot E\]

- As distributivity:
  1. Left distributive law of \(*\) over +:
     \[F \cdot (F + G) = E \cdot F + E \cdot G\]
  2. Right:
     \[(F + G) \cdot E = F \cdot E + G \cdot E\]

Proof of 1: The law actually holds for arbitrary languages and does not require \(E, F, G\) to be R.E.

Hence, we prove: for arbitrary languages \(L, M, N\):

\[L \cdot (M \cup N) = L \cdot M \cup L \cdot N\]

We show that for a string \(w\) we have \(w \in L \cdot (M \cup N)\) if \(w \in L \cdot M \cup L \cdot N\)

"Only if": \(w \in L \cdot (M \cup N) \Rightarrow w = x \cdot y\) with \(x \in L, y \in M \cup N\)

Since \(y \in M \cup N\), either \(y \in M\) or \(y \in N\) (or both).

If \(y \in M\), then \(w = x \cdot y \in L \cdot M\), hence \(w \in L \cdot M \cup L \cdot N\)

(Similarly for \(y \in N\)).

"If": \(w \in L \cdot M \cup L \cdot N\), hence either \(w \in L \cdot M\) or \(w \in L \cdot N\)

If \(w \in L \cdot M\), then \(w = x \cdot y\) with \(x \in L, y \in M\). (Similarly)\n
Hence \(y \in M \cup N\), and \(w = x \cdot y \in L \cdot (M \cup N)\). (for \(w \in L \cdot (M \cup N)\))
Example: \(0.0 + 0.1^* = 0.(0 + 1^*)\)

we can factor out \(e \in D\) from the union

What about \(0 + 0.1^*\)?

if we factor out \(e \in D\), what remains after the summation on the left?

\[0 + 0.1^* = 0.0 + 0.1^* = 0.(\epsilon + 1^*) = 0.1^*\]

identity

\(- \epsilon \in \Sigma^*\)

- identities and annihilators (hold for arbitrary languages)
  \(- \emptyset + E = E + \emptyset = E\)
  \(- \epsilon \cdot E = E \cdot \epsilon = E\)
  \(- \emptyset \cdot E = E \cdot \emptyset = \emptyset\)

- idempotency
  \(- E + E = E\)
  \(- (E^*)^* = E^*\) \(\text{proof: Exercise 3.4.1}\)

- other laws for closure (already seen)
  \(- \emptyset^* = \epsilon\)
  \(- \epsilon^* = \epsilon\)
  \(- E^+ = E \cdot E^* \cdot E^* \cdot E\)
  \(- E^* = E^+ \cdot \epsilon\)

note: if \(\epsilon \in \Sigma E\), then \(E^* = E^+\)

Exercise 3.4.4

Prove that \((E^*F^*)^* = (E + F)^*\)
Exercise 3.1.1 Write R.E.s for the following languages:

a) \( \{ w \in \{a, b, c\}^* | w \text{ contains at least one } a \text{ and at least one } b \} \)

b) \( \{ w \in \{0, 1\}^* | w \text{'s tenth symbol from the right is } 1 \} \)

c) \( \{ w \in \{0, 1\}^* | w \text{ contains at most one pair of consecutive } 1\text{'s} \} \)

Exercise 3.1.2 Write R.E.s for the following languages:

a) The set of all strings over \( \{0, 1\} \) such every pair of adjacent 0's appears before any pair of adjacent 1's.

b) The set of strings of 0's and 1's whose number of 0's is divisible by 5.

Solutions:

3.1.1  
\( a) \quad c^*a(2+c)^*b(2+b+c)^*c^*b(2+b+c)^*a(2+a+b)^*c^* \)

\( b) \quad (0+1)^* \underbrace{1 \cdot (0+1) \cdots (0+1)}_{3 \text{ times}} \)

\( c) \quad 0^* \cdot (1.0^+) \cdot 1 \cdot 0^* \cdot (0.1^+) \cdot 0^* + 0^* \cdot (1.0^+) \)

3.1.2  
\( a) \quad 0^* \cdot (1.0^+) \cdot 1^* \cdot (0.1^+) \quad \text{no pair of adjacent 1's} \quad \text{no pair of adjacent 0's} \)

\( b) \quad (1^*0.1^* \cdot 0.1^* \cdot 0.1^* \cdot 0.1^* \cdot 0.1^*)^* \)
Regular languages

What is the relationship between the classes of languages studied so far?

\[ \varepsilon\text{-NFA} \iff \text{NFA} \iff \text{DFA} \]

Regular languages \leftrightarrow \text{R.E.} \leftrightarrow \text{？}

**Theorem:** $(\text{R.E.} \rightarrow \varepsilon\text{-NFA})$

For every R.E. $E$ there is an $\varepsilon$-NFA $A_E$ such that $L(E) = L(A_E)$.

**Proof:** let us call an $\varepsilon$-NFA simple if

- it has only one final state
- the initial state has no incoming arcs
- the final state has outgoing arcs

We show by structural induction that for each R.E. $E$ there is a simple $\varepsilon$-NFA $A_E$ such that $L(E) = L(A_E)$.

**Basis:** $E = \varepsilon$, $E = \emptyset$, $E = \varepsilon$ for some $\varepsilon \in \Sigma$

\[
\begin{align*}
A_\varepsilon &\xrightarrow{\varepsilon} q_0 \xrightarrow{\varepsilon} q_1 \\
A_\emptyset &\xrightarrow{\varepsilon} q_0 \xrightarrow{\varepsilon} q_0 \\
A_{\varepsilon} &\xrightarrow{\varepsilon} q_0 \xrightarrow{\varepsilon} q_0
\end{align*}
\]

**Inductive step:**

1. $E = F \cup G$
2. $E = F \cdot G$
3. $E = F^*$
4. $E = (F)$
By I.H., there are simple $\varepsilon$-NFA's for $F$ and $G$.

1) $E = F + G$

\[ L(A_E) = L(A_F) \cup L(A_G) = L(F) \cup L(G) = L(F + G) = L(E) \]

by I.H.

2) $E = F \cdot G$

\[ L(A_E) = L(A_F) \cdot L(A_G) = L(F) \cdot L(G) = L(F \cdot G) = L(E) \]

by I.H.

3) $E = F^*$

\[ A_E = A_F \]

q.e.d.
Example: \( E = 2^* + b, c \)

Diagram:

```
O --- e ----> O --- e ----> O --- e ----> O
|            |            |            |
v            v            v            v
O ----> O ----> O ----> O
```

Theorem (DFA \( \rightarrow \) R.E.):

For every DFA \( A \) there is a R.E. \( E_A \) not \( L(E_A) = L(A) \)

Proof: Let \( A = (Q, \Sigma, \delta, q_0, F) \)

Assume without loss of generality \( \mu = \) \ where \( Q = \{ q_0, q_1, \ldots, q_m \} \)

Let us define \( L_{ij} = \{ w \mid \delta(q_i, w) = q_j \} \) where \( V_{ij} \in \{ 1, \ldots, n \} \)

Note that \( L_{ij} = L(A_{ij}) \) with \( A_{ij} = (Q, \Sigma, \delta, q_i, \{ q_j \}) \)

We aim at constructing R.E.'s \( E_{ij} \) for \( L_{ij} \)

Then we can take \( E_A = \sum_{q_3 \in F} E_{q_3} \), since \( L(E_A) = \bigcup_{q_3 \in F} \bigcup_{q_3 \in F} \{ w \mid \delta(q_3, w) \in F \} = L(A) \)

How can we compute \( E_{ij} \)?

Let us define \( V_{ij} \in \{ 1, \ldots, n \} \), \( V_k \in \{ 0, \ldots, n \} \)

\( L_{ij} = \{ w \mid A \text{ goes from } q_i \text{ to } q_j \text{ on input } w \),

passing only through \( q_1, \ldots, q_n \text{ as intermediate states} \} \)
Example:

\[ L_{12} = \{e, d\} \]
\[ L_{13} = \{abc, dbc\} \]
\[ L_{14} = L_{15} = \{abc, dbc, def, def\} \]

Note: \( L_{ij} = L_{ji} \)

Hence, we are done if we can construct R.E.s \( E_{ij}^k \) for \( L_{ij} \).

We can simply take \( E_{ij} = E_{ij}^\infty \), and hence \( E_A = \sum_{q_j \in F} E_{ij}^\infty \).

We construct \( E_{ij}^k \) by induction on \( k \):

**Basis:** we construct \( E_{ij}^0 \), for all \( i, j \in \{1, \ldots, n\} \).

Since \( k = 0 \), we cannot go through any intermediate state.

2 cases, each with 2 sub-cases:

1. \( i \neq j \)
   - \( q_i \)
   - \( q_j \)
   - \( q_i \)
   - \( q_j \)
   - \( q_i \)
   - \( q_j \)
   - \( q_i \)
   - \( q_j \)
   - \( q_i \)

   \[ E_{ij}^0 = \varepsilon + \cdots + a_n \]

2. \( i = j \)
   - \( q_i \)
   - \( q_i \)
   - \( q_i \)
   - \( q_i \)
   - \( q_i \)
   - \( q_i \)
   - \( q_i \)
   - \( q_i \)
   - \( q_i \)
   - \( q_i \)

   \[ E_{ii}^0 = \varepsilon \]

   \[ E_{ii}^0 = \varepsilon + a_n + \cdots + a_1 \]
Induction: since we have constructed $E_{i,j}^{h-1}$ for $i, j \in \{1, \ldots, n\}$, we show how to construct $E_{i,j}^h$.

Observe:

- $L_{i,j}^h$ will include $L_{i,j}^{h-1}$.
- It additionally will contain those words that lead through $q_h$ at least once, when going from $q_i$ to $q_j$.

$w = x_1 x_2 \ldots x_n$, where $\Rightarrow$ represents transitions going at most through $\{x_1, \ldots, x_{h-1}\}$.

Then $x_n \in L_{i,k}^{h-1}$,

$k_2, \ldots, k_{n-1} \in L_{h,k}^{h-1}$

$x_0 \in L_{h,j}^{h-1}$

$\Rightarrow w \in L_{i,k}^{h-1} \cdot \left(L_{h,k}^{h-1}\right)^* \cdot L_{h,j}^{h-1}$

$\Rightarrow E_{i,j}^h = E_{i,j}^{h-1} + E_{i,k}^{h-1} \cdot \left(E_{h,k}^{h-1}\right)^* \cdot E_{h,j}^{h-1}$

Sample:

- $E_{11}^1 = E_{11}^{h-1}$
- $E_{12}^1 = E_{12}^{h-1}$
- $E_{21}^1 = \emptyset$
- $E_{22}^1 = E_{22}^{h-1} + E_{12}^{h-1} \cdot \left(E_{21}^{h-1}\right)^* \cdot E_{12}^{h-1}$

$\delta(A) = \delta(E_{12}^2) = \delta(E_{21}^2)$

$E_{12}^1 = E_{12}^{h-1} = O^* \cdot (0^*1)^*$

$\Rightarrow$ Optional
Theorem (DFA → R.E.)

For every DFA $A$, there is a R.E. $E_A$ s.t. $L(E_A) = L(A)$

Proof sketch: We show how to construct $E_A$ by eliminating states of $A$.

Consider the elimination of a state $q$:

If there was a path from state $p$ to state $q$ over $s$, after eliminating $q$, the path does no longer exist.

So we have to compensate for that.

We add a regular expression "connecting" $p$ and $q$ and capturing the missing path.

We can eliminate in this way all states except initial and final states:

Strategy:

a) For each final state $q$, eliminate all states except $q, q_0$

b) If $q \neq q_0$, we are left with:

$$R + P \cdot S^* \cdot (q_0 \rightarrow q)$$

(c) If $q_0 \notin E$, we must eliminate all states except $q_0$.

We are left with:

$$R^*$$

(d) We take the union of all derived R.E.s.

$$R + P \cdot S^* \cdot (q_0 \rightarrow q)$$
Example

We view all edge labels as R.E.N. (missing labels mean $\text{\textbf{\varnothing}}$)

Eliminate $B$:

$$E_B = \emptyset + 1 \cdot \emptyset^*(0 \cdot a) = 1 \cdot \emptyset^*(0 \cdot a) = 1 \cdot (0 \cdot a)$$

Eliminate $C$:

$$E_C = \emptyset + 1 \cdot (0 \cdot a) \cdot \emptyset^* \cdot (0 \cdot a) = 1 \cdot (0 \cdot a) \cdot (0 \cdot a)$$

$$E_1 = (0 \cdot a)^* \cdot E_C = (0 \cdot a)^* \cdot 1 \cdot (0 \cdot a) \cdot (0 \cdot a)$$

$D$:

$$E_D = (0 \cdot a)^* \cdot 1 \cdot (0 \cdot a)$$

$$E = E_1 + E_2 = (0 \cdot a)^* \cdot 1 \cdot (0 \cdot a) \cdot (0 \cdot a) + (0 \cdot a)^* \cdot 1 \cdot (0 \cdot a)$$

$$= (0 \cdot a)^* \cdot 1 \cdot (0 \cdot a) \cdot (3 + 0 \cdot a)$$