Regular expressions

- $\emptyset$
- $a$ for all $a \in \Sigma$
- If $r_1$, $r_2$ are regular expressions then so are $r_1 \cup r_2$ and $r_1 r_2$
- If $r$ is a regular expression then so is $r^*$
- Nothing else is a regular expression

If $w$ is a string, then $w$ is a regular expression.
Regular expressions

\( \mathcal{L}(r) \) — language denoted by \( r \)

<table>
<thead>
<tr>
<th>( r )</th>
<th>( \mathcal{L}(r) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \emptyset )</td>
<td>( \emptyset )</td>
</tr>
<tr>
<td>( a )</td>
<td>( {a} )</td>
</tr>
<tr>
<td>( r_1 \cup r_2 )</td>
<td>( \mathcal{L}(r_1) \cup \mathcal{L}(r_2) )</td>
</tr>
<tr>
<td>( r_1r_2 )</td>
<td>( \mathcal{L}(r_1)\mathcal{L}(r_2) )</td>
</tr>
<tr>
<td>( r^* )</td>
<td>( \mathcal{L}(r)^* )</td>
</tr>
</tbody>
</table>

The operator precedence is \( * \succ \circ \succ \cup \)

\( \mathcal{L}(ab^* \cup aa) = (\{a\} \circ \{b\}^*) \cup \{aa\} \)
Deterministic Finite Automaton (DFA)

\[ M = (Q, \Sigma, \delta, q_0, F) \]

- \( Q \) = set of states of the finite automaton
- \( \Sigma \) is a finite alphabet
- \( \delta : Q \times \Sigma \rightarrow Q \) is the transition function
- \( q_0 \in Q \) is the initial state
- \( F \subseteq Q \) is the set of accepting states
Acceptance by DFA

\[ \delta^* : Q \times \Sigma^* \rightarrow Q \]

\[ \delta^*(q, \epsilon) = q \]

\[ \delta^*(q, ax) = \delta^*(\delta(q, a), x) \]

\( w \) is accepted by \( M \) iff \( \delta^*(q_0, w) \in F \)

\[ L(M) = \{ w \mid M \text{ accepts } w \} \]

\( M \text{ recognizes } L \) iff \( L = L(M) \)
Dead states

\[ \forall w : \delta^* (q, w) \notin F \]

\[ q_2 \text{ is a dead state} \]
Product construction

\[ M_1 = (P, \Sigma, \delta_1, p_0, F_1) \]
\[ M_2 = (Q, \Sigma, \delta_2, q_0, F_2) \]

\((P \times Q, \Sigma, \delta, (p_0, q_0), \ldots)\)

where

\[ \delta((p, q), b) = (\delta_1(p, b), \delta_2(q, b)) \]

for all \((p, q) \in P \times Q, b \in \Sigma\)

<table>
<thead>
<tr>
<th>Aim</th>
<th>( F )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \cap )</td>
<td>( F_1 \times F_2 )</td>
</tr>
<tr>
<td>( \cup )</td>
<td>{ (p, q)</td>
</tr>
<tr>
<td>( \setminus )</td>
<td>{ (p, q)</td>
</tr>
</tbody>
</table>