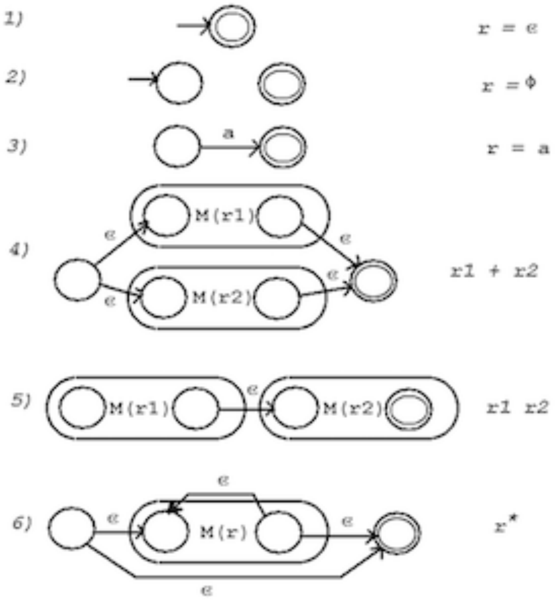
2017-10-05

- Definition: a language is **regular** iff there is a DFA that accepts it.
- **Regular expressions** over Σ include (1) $\emptyset$, (2) ϵ, (3) $a \in \Sigma$, (4) $e_1 \cup e_2$, (5) $e_1 e_2$, (6) e^*.
- A regular expression e over Σ defines a language $L(e)$ as follows: (1) $L(\emptyset) = \emptyset$, (2) $L(\epsilon) = \{\epsilon\}$, (3) $L(a) = \{a\}$, (4) $L(e_1 \cup e_2) = L(e_1) \cup L(e_2)$, (5) $L(e_1 e_2) = L(e_1)L(e_2)$, (6) $L(e^*) = L(e)^*$
- Examples: (1) $L(e \cup \emptyset) = L(e)$, (2) $L(e\emptyset) = \emptyset$, (3) $L(e \cup \epsilon) = L(e) \cup \{\epsilon\}$, (4) $L(e\epsilon) = L(e)$, (5) $L(\emptyset^*) = \{\epsilon\}$, (6) $L(\epsilon^*) = \{\epsilon\}$.
- DFA, NFA, and regular expression:
  - Equivalent in their descriptive power.
  - Closed under complement, union, intersection, concatenation, Kleene star.
- Theorem: For every regular expression e, there is an NFA A s.t. $L(e) = L(A)$.
- $L(e) \subseteq L(A)$ proved by construction:



- $L(A) \subseteq L(e)$ proved by induction:
  - Definition: $A = \langle \Sigma, Q, q_0, F, \delta \rangle$, where $Q = \{q_0, q_1, q_2, \ldots, q_m\}$.
  - Definition: $L(i, j, k) = \{w|$ there is a run from state i to state j passing only through state $\leq k\}$, where $i, j, k \in \{0, 1, 2, \ldots, m\}$.
  - *Hypothesis*: For any $k \in \{0, 1, 2, \ldots, m\}$, there is a valid regular expression $e_{i,j,k}$ s.t. $L(i, j, k) = L(e_{i,j,k})$.
  - *Basis*: When $k = 0$, $e_{i,j,0}$ is always a valid regular expression s.t. $L(i, j, 0) = L(e_{i,j,0})$.
  - *Induction*: Assume the hypothesis hold for $k = m - 1$ and consider $k = m$. $L(i, j, m) = L(i, j, m-1) \cup L(i, m, m-1)L(m, m, m-1)^*L(m, j, m-1) = L(e_{i,j,m-1} \cup e_{i,m,m-1}\, e^*_{m,m,m-1}\, e_{m,j,m-1})$. Because $e_{i,j,m-1}$, $e_{i,m,m-1}$, $e_{m,m,m-1}$, and $e_{m,j,m-1}$ are all regular expressions, $e_{i,j,m} = e_{i,j,m-1} \cup e_{i,m,m-1}\, e^*_{m,m,m-1}\, e_{m,j,m-1}$ is also a valid regular expression.
  - *Conclusion*: $L(A) = \bigcup_{q_j \in F} L(0, j, k) = \bigcup_{q_j \in F} L(e_{0,j,k})$, and $\bigcup_{q_j \in F} e_{0,j,k}$ is the equivalent regular expression for any $k \in \{0, 1, 2, \ldots, m\}$.
- Again, since the set of all languages is uncountably infinite, and the set of regular languages is countably infinite, there is a language which is *not* regular.
- **Pumping lemma** is a useful tool to prove languages that are *not* regular.

- An example of proof using the *pumping lemma*: Prove $L = \{a^n b^n | n \in \mathbb{N}\}$ is *not* regular.
  - Suppose L is regular, and $A = \langle \Sigma, Q, q_0, F, \delta \rangle$, where $\Sigma = \{a, b\}$, is its DFA.
  - Consider the following word $w = a^k b^k$, where $k \geq |Q|$.
  - Then, a^k can be divided into three part $a^k = a^u a^v a^w$, and $k = u + v + w$ s.t. $a^u (a^v)^n a^w b^k \in L(A)$ for every nonnegative integer $n \geq 0$.
  - However, there is a contradiction since $a^u (a^v)^n a^w b^k = a^{u+nv+w} b^k \notin L(A)$ for nonnegative integers $n \neq 1$.
  - Therefore, L is *not* regular.