Nondeterministic Finite Automata (NFA)

- allow choices (and $\varepsilon$-transitions)

**Ex. 1**
all strings ending with 01

![NFA Diagram]

not valid DFA!!

$\delta(q_0, 0) = q_0 \lor q_1$ ??

but valid NFA

$\delta(q_1, 0)$

but what does it mean for an NFA to accept an input?

Idea - accept iff $\exists$ path from start state to some accepting state

reject iff $\forall$ path from start state end in non-accept state

not realistic machine!

ability to guess

**Ex. 2**
$0^* (01)^* 1^*$

![NFA Diagram]

$\varepsilon$-transitions don't consume input

Formal Defn
An NFA is $M = (Q, \Sigma, s, \delta, A)$

like before
except $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow 2^Q$
except \( \delta : Q \times (\Sigma \cup \{\varepsilon\}) \to \mathcal{P}(Q) \)

\[ \text{power set} \]
\[ = \text{set of all} \]
\[ \text{Subsets of Q} \]

Ex1: \( \delta(q_0,0) = \{q_0,q_1\} \)
\( \delta(q_1,0) = \emptyset \)

Ex2: \( \delta(q_0,\varepsilon) = \{q_1\} \)

**Def**

For \( q \in Q \), define \( \varepsilon\text{-reach}(q) \) recursively:

(i) \( q \) is in \( \varepsilon\text{-reach}(q) \)

(ii) if \( q' \) is in \( \varepsilon\text{-reach}(q) \) & \( q'' \in \delta(q',\varepsilon) \), then \( q'' \) is in \( \varepsilon\text{-reach}(q) \)

(iii) nothing else is in.

Ex2: \( \varepsilon\text{-reach}(q_0) = \{q_0,q_1,q_3\} \)
\( \varepsilon\text{-reach}(q_1) = \{q_1,q_3\} \).

**Def**

Define extended transition fn \( \delta^* : Q \times \Sigma^* \to \mathcal{P}(Q) \) recursively:

(i) \( \delta^*(q,\varepsilon) = \varepsilon\text{-reach}(q) \)

(ii) if \( x = ay \) (\( a \in \Sigma, y \in \Sigma^* \)),

\[ \delta^*(q,x) = \bigcup_{q' \in \delta^*(q,a)} \bigcup_{q'' \in \delta(q',\varepsilon)} \delta^*(q'',y) \]

Ex1: \( \delta^*(q_0,100) = \delta^*(q_0,00)^* = \delta^*(q_0,0) \cup \delta^*(q_0,0)^* \)

**Def**

\( M \) accepts \( x \) if \( \delta^*(s,x) \cap A \neq \emptyset \).

\[ L(M) = \{ x \in \Sigma^* : M \text{ accepts } x \} \]

Ex1: \( (q_0,0) \cup (q_1,0)^* \in \Sigma^* \)
Ex  a) \((\varepsilon + 0)(01 + 001)^* (1)^*\)

b) all strings whose 5th last symbol is 0
   (DFA with 32 states)

c) all strings not ending with 0

Wrong.

Regular \downarrow
NFA \downarrow
DFA

Regular \rightarrow NFA

Thm If \(L_1\) is accepted by some NFA \(M_1\),
\(L_2 \leftarrow \ldots \leftarrow \ldots \leftarrow M_2\),
**Thm**

If \( L_1 \) is accepted by some NFA \( M_1 \) and \( L_2 \) is accepted by some NFA \( M_2 \), then:

(i) \( L_1 \cup L_2 \) is accepted by some NFA \( M' \)

(ii) \( L_1 \cdot L_2 \) is accepted by some NFA \( M' \)

(iii) \( L_1^* \) is accepted by some NFA \( M' \)

**PF:**

Given \( M_1 = (Q_1, \Sigma, s_1, \delta_1, A_1) \) and \( M_2 = (Q_2, \Sigma, s_2, \delta_2, A_2) \),

(i) **union**

(ii) **concat**

(iii) **star**
If $L$ is regular, then $L$ is accepted by some NFA.

Pf: By recursion (induction).
Base cases: $\emptyset, \{e\}, \{a\}$