Nondeterministic Finite Automata (NFA)

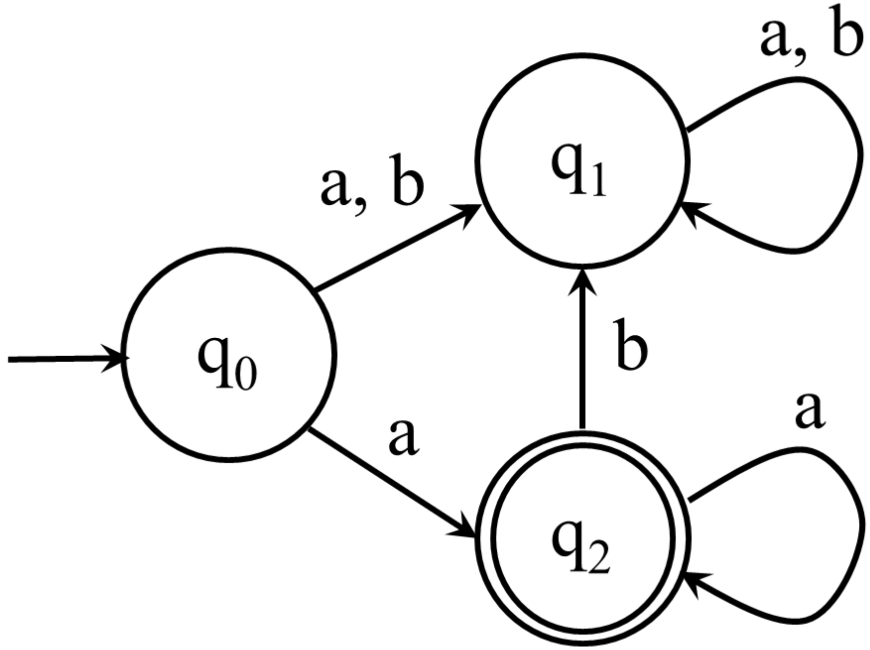
- NFAs are just like DFAs, except that:
  - There may be multiple transitions from the same state with the same character label
  - A transition may be an ε-transition
- Thus, the formal definition of NFAs are the same as DFAs, except for the transition function:

$$\delta : Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow \mathcal{P}(Q)$$

Power set

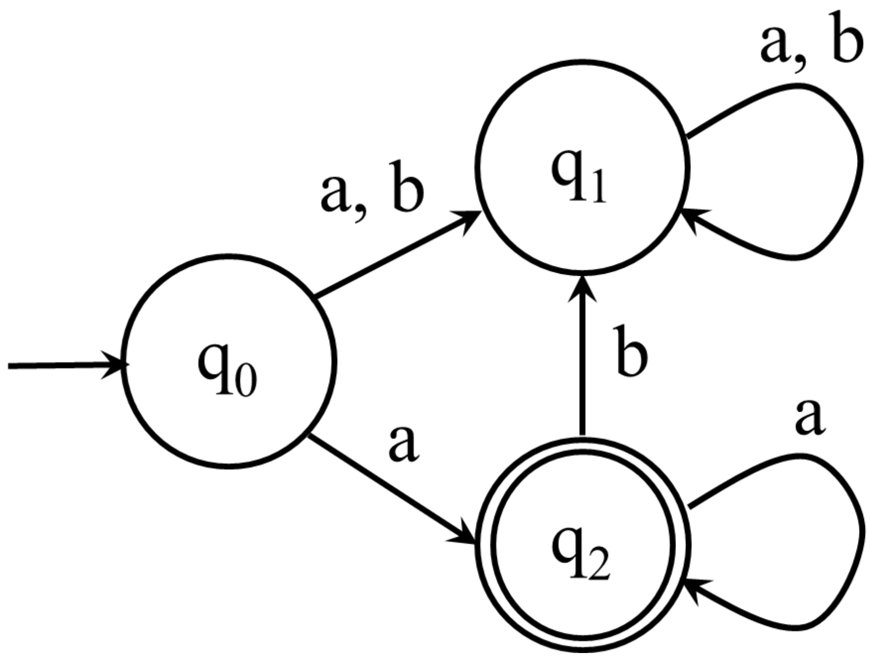
A Quick Question

- Does the following NFA accept the word aaa?



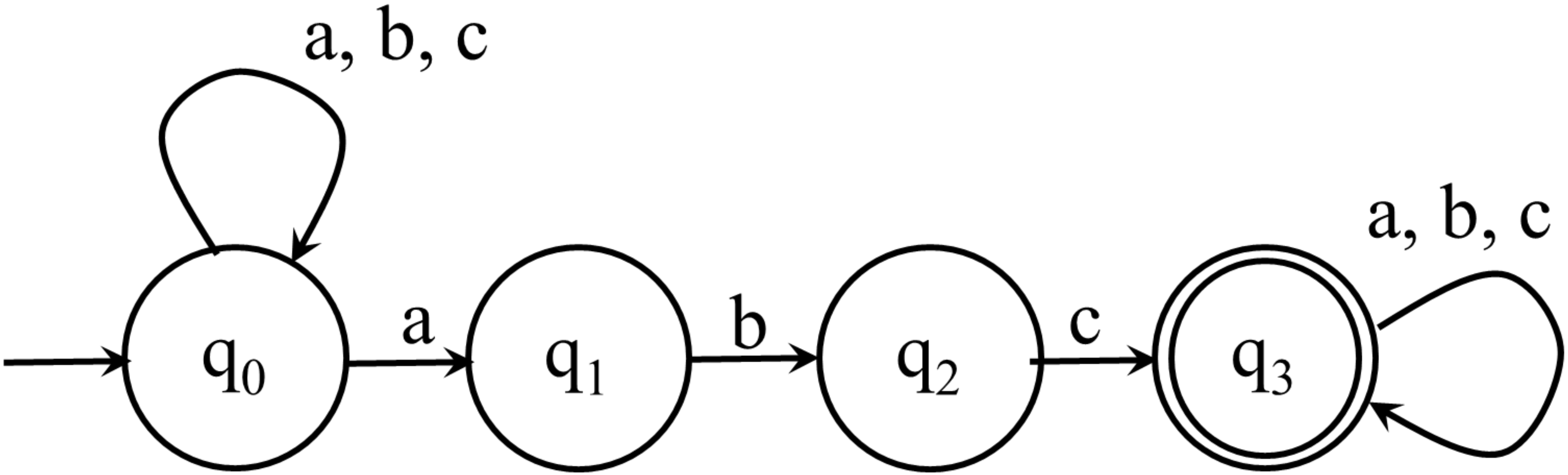
A Quick Question

- Does the following NFA accept the word aaa?



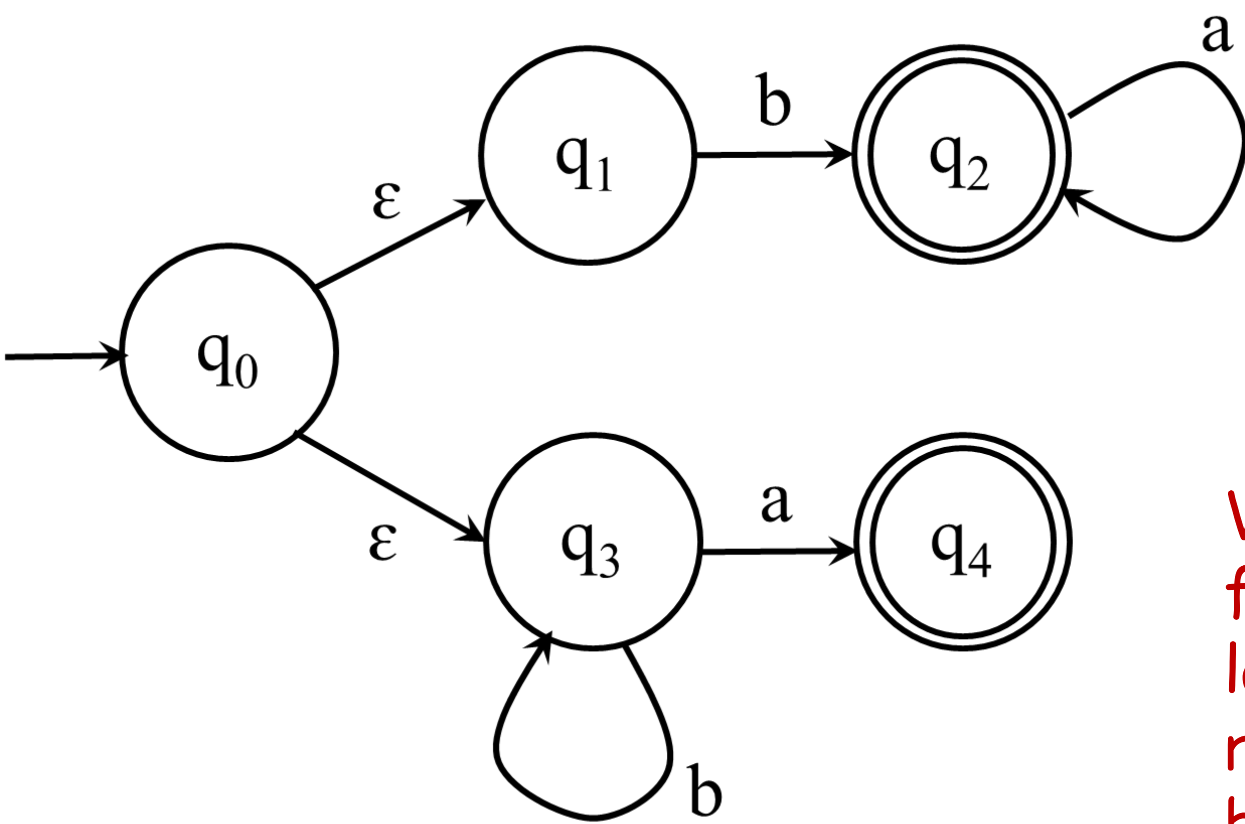
- Yes – An NFA accepts a word w if there exists at least one transition path for w that ends in an accept state

Another Example



- What language does this NFA recognize?

Yet Another Example - ε-transitions

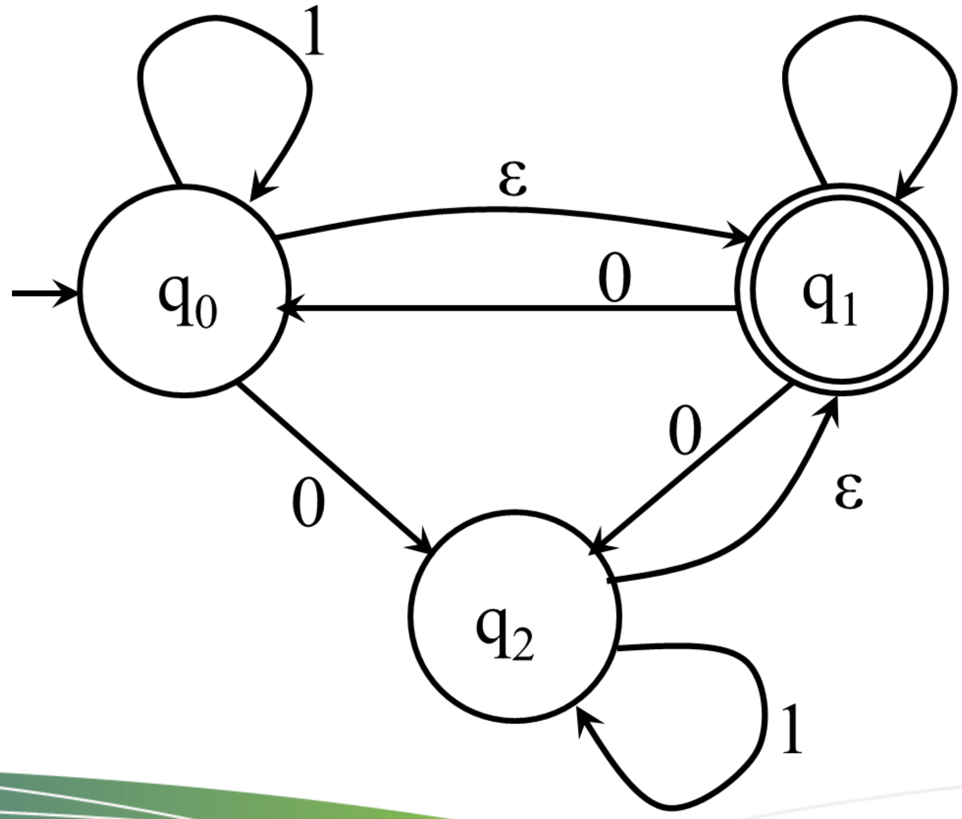


Very useful for unioning languages recognized by DFAs

- ε-transitions allow us to automatically move into the next state without matching a character

One More Example

- What language does this NFA recognize?



NFAs are Equivalent to DFAs!

- In other words, they both recognize the same class of languages
- Proof idea: Show that you can convert any NFA into an equivalent DFA
  - The other direction (DFA → NFA) follows automatically from the definitions

Proof (Idea) by Construction

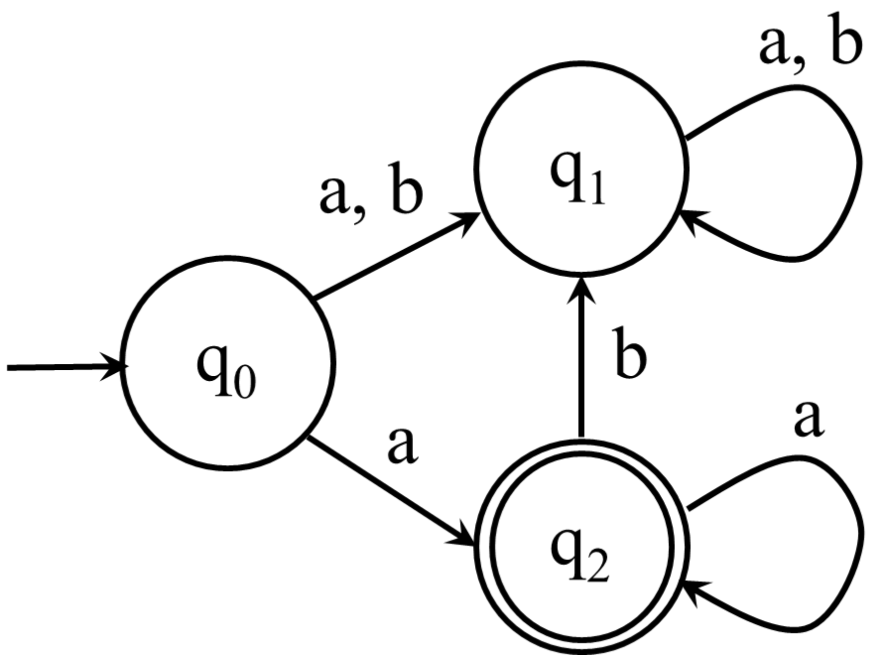
- Converting an NFA into a DFA:
  - States: $\mathcal{P}(Q)$ (represent new DFA states as subsets of the original set of NFA states Q)
  - Alphabet Σ stays the same
  - Transition function: $\delta(\{q_n, q_{n1}, \ldots, q_{nk}\}, c)$ maps to the set of states of the NFA that are reachable from any of $q_n, q_{n1}, \ldots, q_{nk}$ by reading the character c
  - Start state $\{ q_0 \}$
  - Accept/final states are those subsets of Q that contain any final state of the NFA

Proof (Idea) by Construction

- Converting an NFA into a DFA:
  - States: $\mathcal{P}(Q)$ (represent new DFA states as subsets of the original set of NFA states Q)
  - Alphabet Σ stays the same
  - Transition function: $\delta(\{q_n, q_{n1}, \ldots, q_{nk}\}, c)$ maps to the set of states of the NFA that are reachable from any of $q_n, q_{n1}, \ldots, q_{nk}$ by reading the character c
  - Start state $\{ q_0 \}$
  - Accept/final states are th[...] any final state of the NF[...]

One c transition, along with any number of ε-transitions before and after

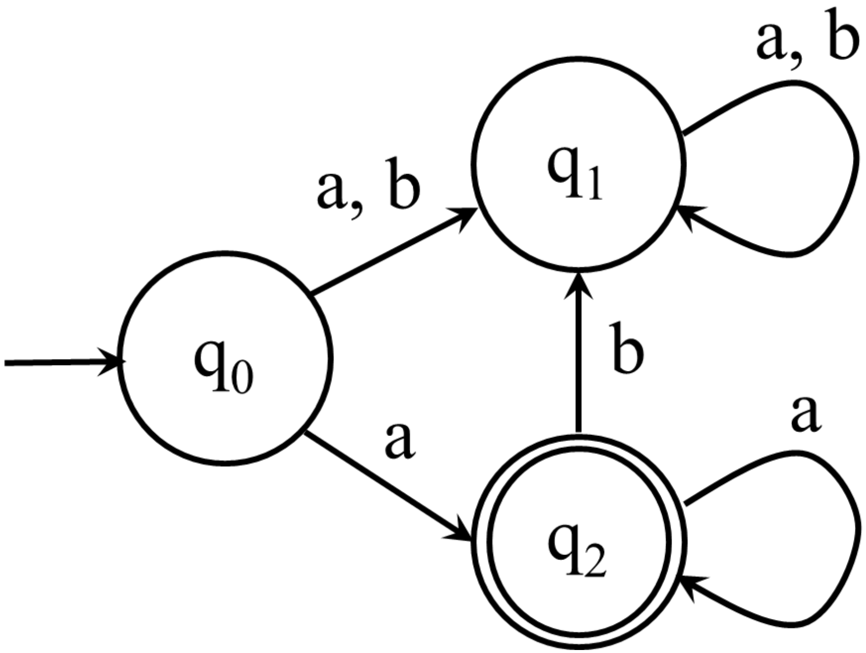
Conversion Example



	a	b
$\{q_0\}$	$\{q_1,q_2\}$	$\{q_1\}$
$\{q_1\}$	$\{q_1\}$	$\{q_1\}$
$\{q_1,q_2\}$	$\{q_1,q_2\}$	$\{q_1\}$

- Step 1 – Create the transition table, starting with $\{q_0\}$, and keep generating subsets of reachable states for each character for each subset

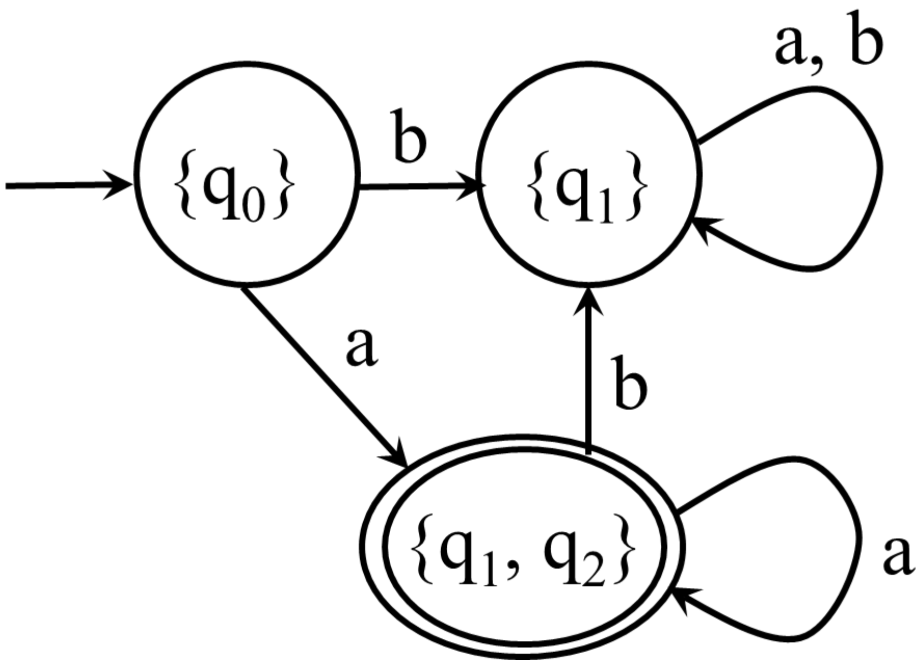
Conversion Example



	a	b
$\{q_0\}$	$\{q_1,q_2\}$	$\{q_1\}$
$\{q_1\}$	$\{q_1\}$	$\{q_1\}$
$\{q_1,q_2\}$	$\{q_1,q_2\}$	$\{q_1\}$

- Step 2 – Flag each subset that contains an accept state – these will represent our new accept states

Conversion Example



	a	b
$\{q_0\}$	$\{q_1,q_2\}$	$\{q_1\}$
$\{q_1\}$	$\{q_1\}$	$\{q_1\}$
$\{q_1,q_2\}$	$\{q_1,q_2\}$	$\{q_1\}$

- Step 3 – Use the table to create the new DFA