Languages (Not Programming Languages)

- An alphabet is a finite set of characters, usually denoted Σ
- A string or word is a finite sequence of characters
- The empty string (ε) has no characters
- A language is a set, which could be infinite, of strings over a given alphabet

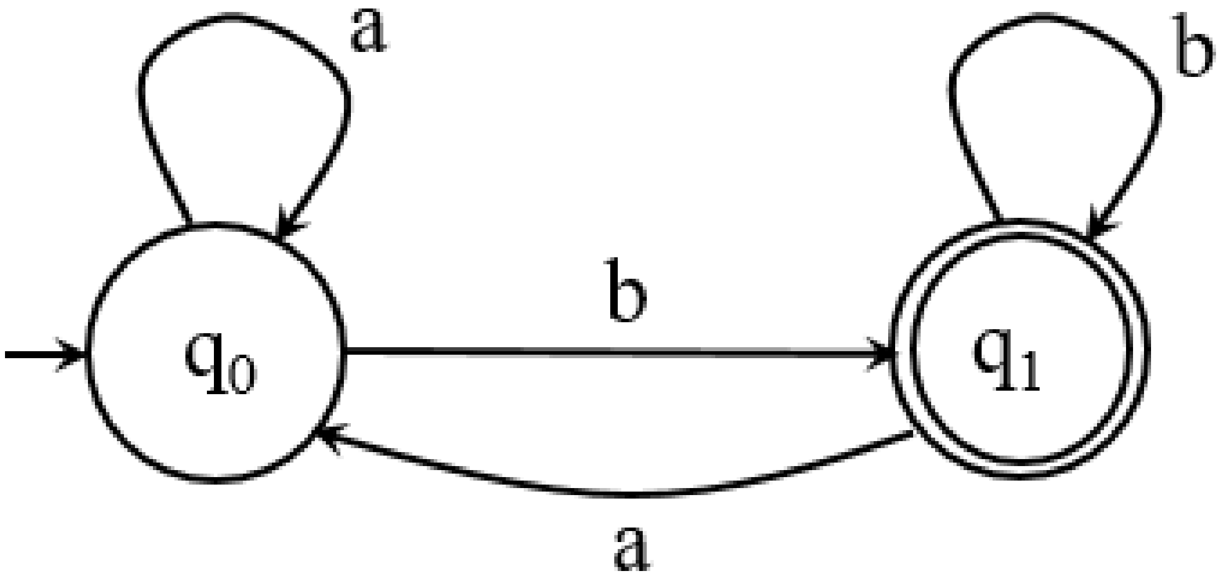
Deterministic Finite Automata (DFA)

- A DFA is a model of computation that can be used to define a language
- Formal parts of a DFA:
  - States: $Q = \{q_0, q_1, q_2, \ldots, q_k\}$
  - Alphabet: Σ
  - Transition function: $\delta : Q \times \Sigma \rightarrow Q$
  - Start state: $q_0 \in Q$
  - Accept/final states: $F \subseteq Q$

DFA Characterizations

- A DFA could be thought of as:
  - An oracle that answers the question "Does a given string w belong to the language L?"
  - A language generator, which can be used to create all of the words in a language L by doing a breadth-first traversal over transition sequences

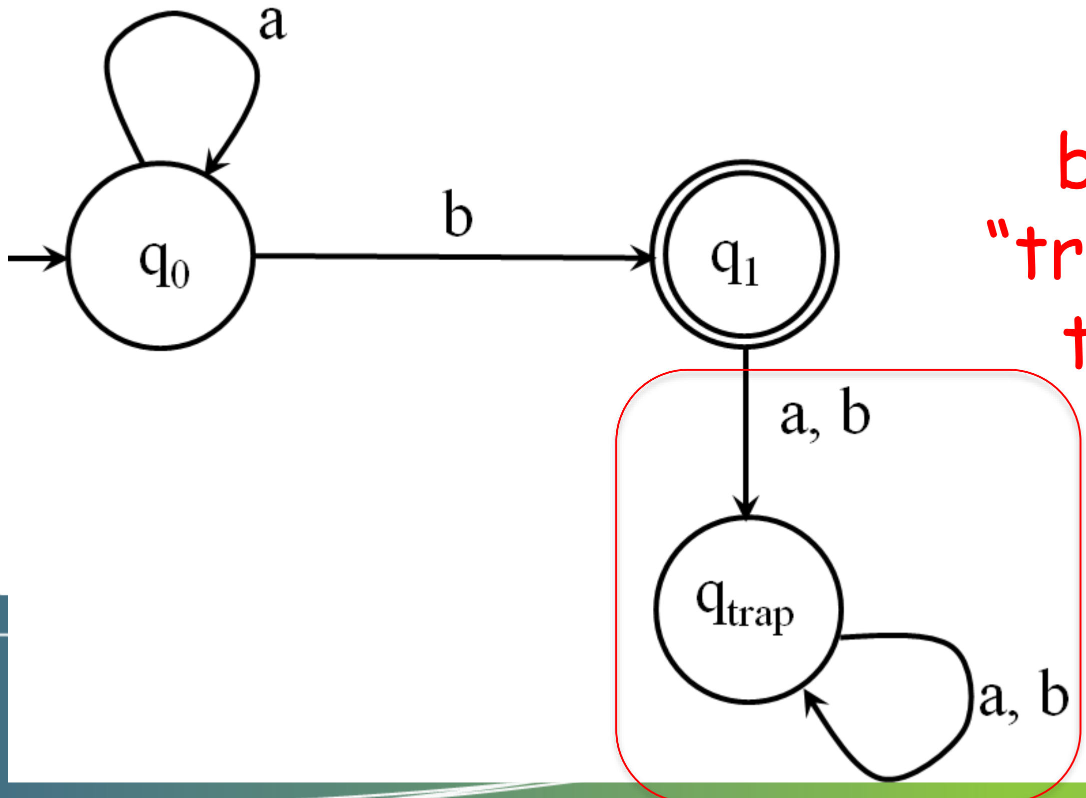
A Simple Example



- This DFA accepts words such as ab, aab, and aaabbb, but also ababbaab and b

Another Example

- Create a DFA that only accepts those words that begin with zero or more a's, and ends with a single b



People usually don't bother to include this "trap state" part, but it technically makes the DFA complete by showing all possible transitions

More Terminology

- If a DFA M ends in an accepting state after processing a given word w, we say that M accepts w
- Otherwise, we say that M rejects w
- If M accepts all words in a language L, and rejects all others, then we say that M accepts L or M recognizes L