Deterministic Finite Accepters

A *deterministic finite accepter* M is:

Q, a list of internal states

Σ, the input alphabet

$\delta : Q \times \Sigma \rightarrow Q$, the transition function.

$q_0 \in Q$, the initial state

$F \subseteq Q$, the final states

Behavior of DFAs

Start in state q_0.

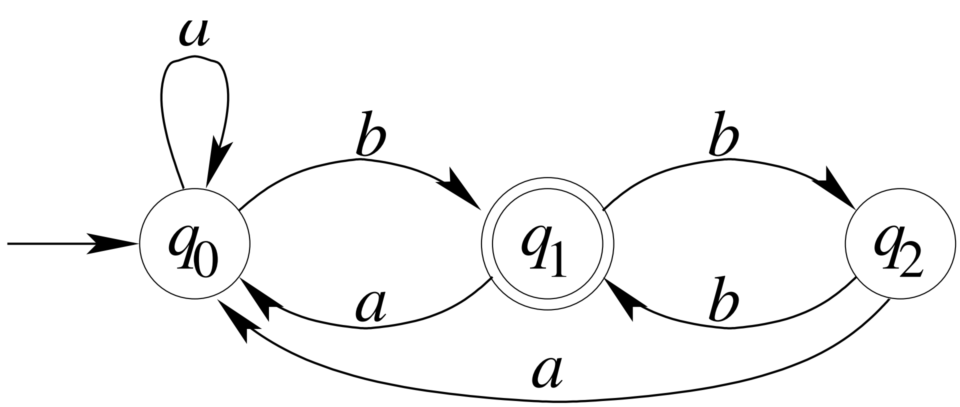
Repeatedly:

read the next input symbol and
move to the next state given by δ.

Accept if the last state is a final state.

Representations of DFAs

Transition Graph:



Transition Table:

	a	b
q_0	q_0	q_1
q_1	q_0	q_2
q_2	q_0	q_1

DFAs and Languages

Let $\delta^* : Q \times \Sigma^* \to Q$ be the *extended transition function.*

$\delta^*(q, w) = q'$ if a DFA in state q goes to state q' after reading w.

Here is a recursive definition for δ^*:

$$\delta^*(q, \lambda) = q$$
$$\delta^*(q, aw) = \delta^*(\delta(q, a), w)$$

DFAs and Languages Continued

The language accepted by a DFA M is the set of all strings accepted by M.

$$\mathcal{L}(M) = \{w \in \Sigma^* : \delta^*(q_0, w) \in F\}$$

The *family* of ***regular languages*** is the set of languages that can be accepted by DFAs. That is, L is a regular language iff there is a DFA M such that $L = \mathcal{L}(M)$.

Some Properties of DFAs

A DFA can be simulated in time $O(n)$ where n is the length of input string.

Repetition of States:
Let M be a DFA with m states.
Let M read a string w with $|w| \geq m$.
Then M visits at least one state twice.
Proof: M visits $|w| + 1 \geq m + 1$ states.
Conclusion follows from Pigeonhole Principle.

Properties Continued

Closure under complementation:
If L is a regular language, then so is $\overline{L}$.
Proof: Let M be a DFA s.t. $L = \mathcal{L}(M)$.
Let Q and F be M's states and final states.
Let $M' = M$, but with final states $Q - F$.
M' accepts w iff M rejects w,
so $\mathcal{L}(M') = \overline{\mathcal{L}(M)} = \overline{L}$

Finite Languages:
Every finite language is a regular language.
Construction (in class)