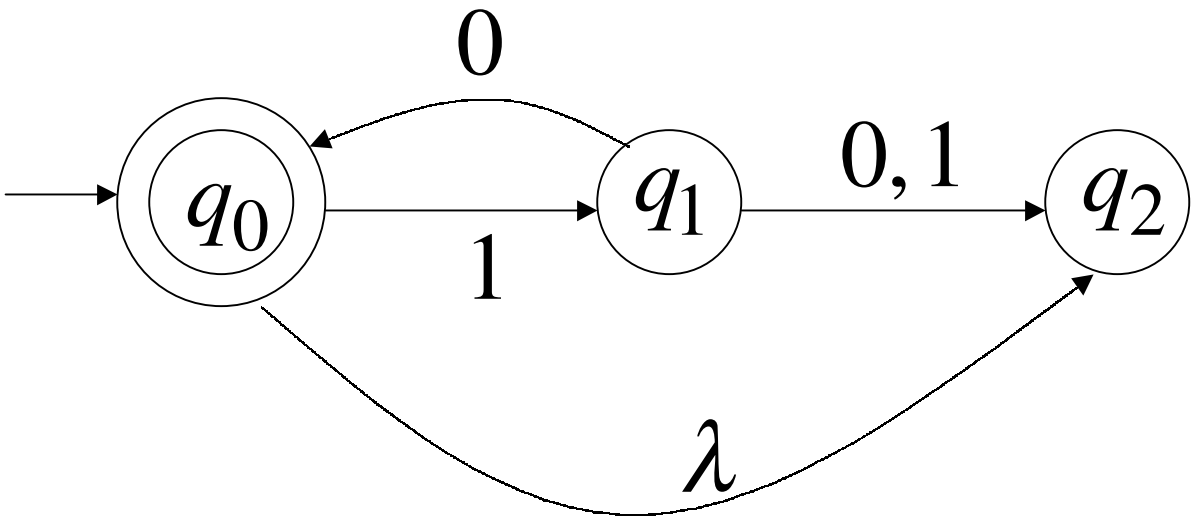
Non-Deterministic Automata

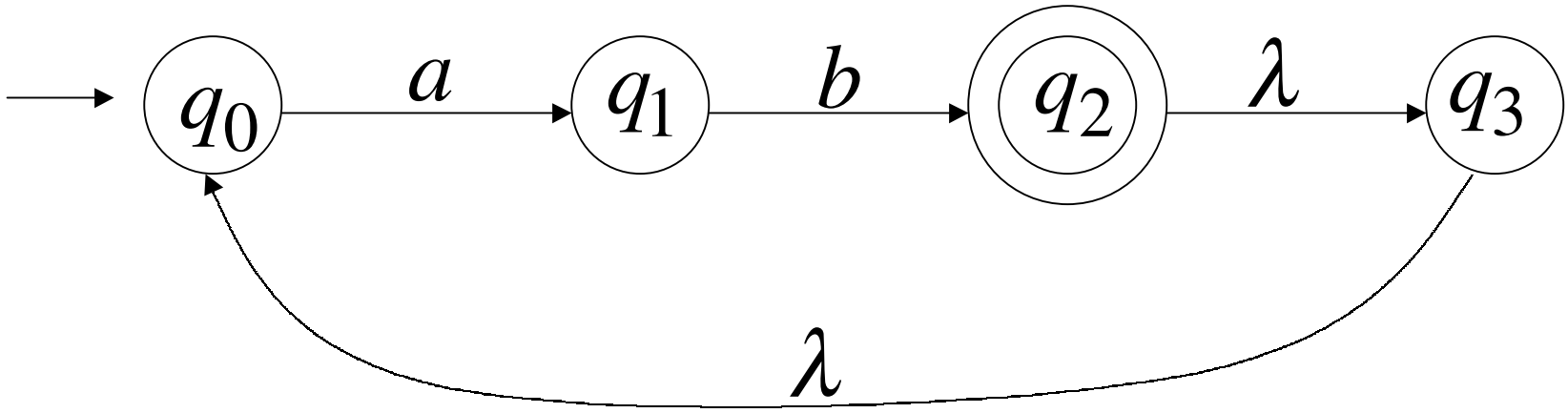
Regular Expressions

NFA Example



$$L = \{\lambda,\ 10,\ 1010,\ 101010,\ ...\}$$
$$= \{10\}^*$$

NFA Example



$$L = \{ab,\ abab,\ abab,\ ...\}$$

$$= \{ab\}^{+}$$

Formal Definition of NFAs

$$M = (Q,\ \Sigma,\ \delta,\ q_0,\ F)$$

Q: Set of states, i.e. $\{q_0, q_1, q_2\}$

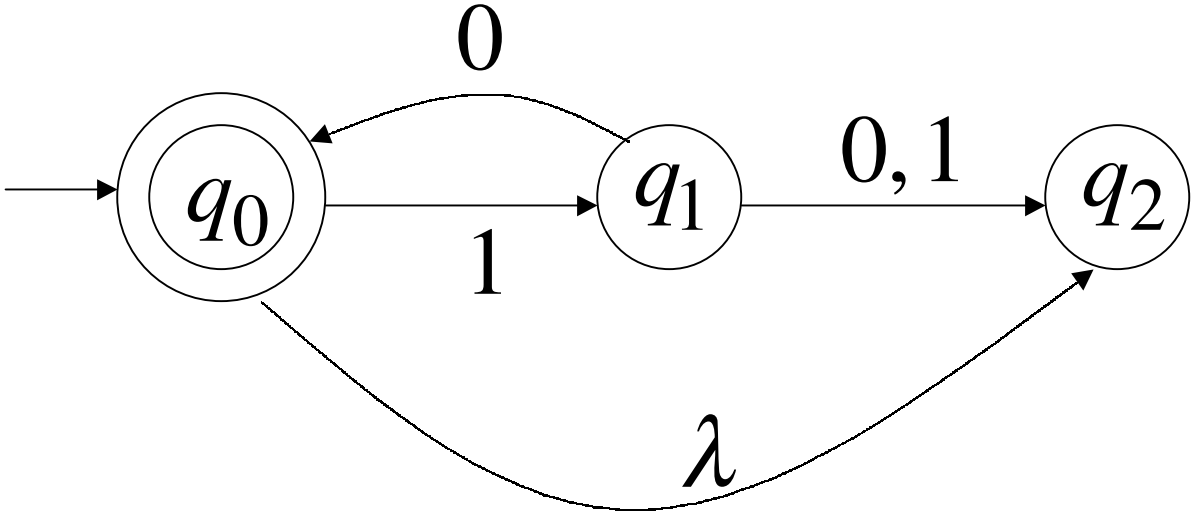
Σ: Input aplhabet, i.e. $\{a, b\}$

δ: Transition function

q_0: Initial state

F: Final states

Transition Function δ



$$\delta(q_0, 1) = \{q_1\}$$

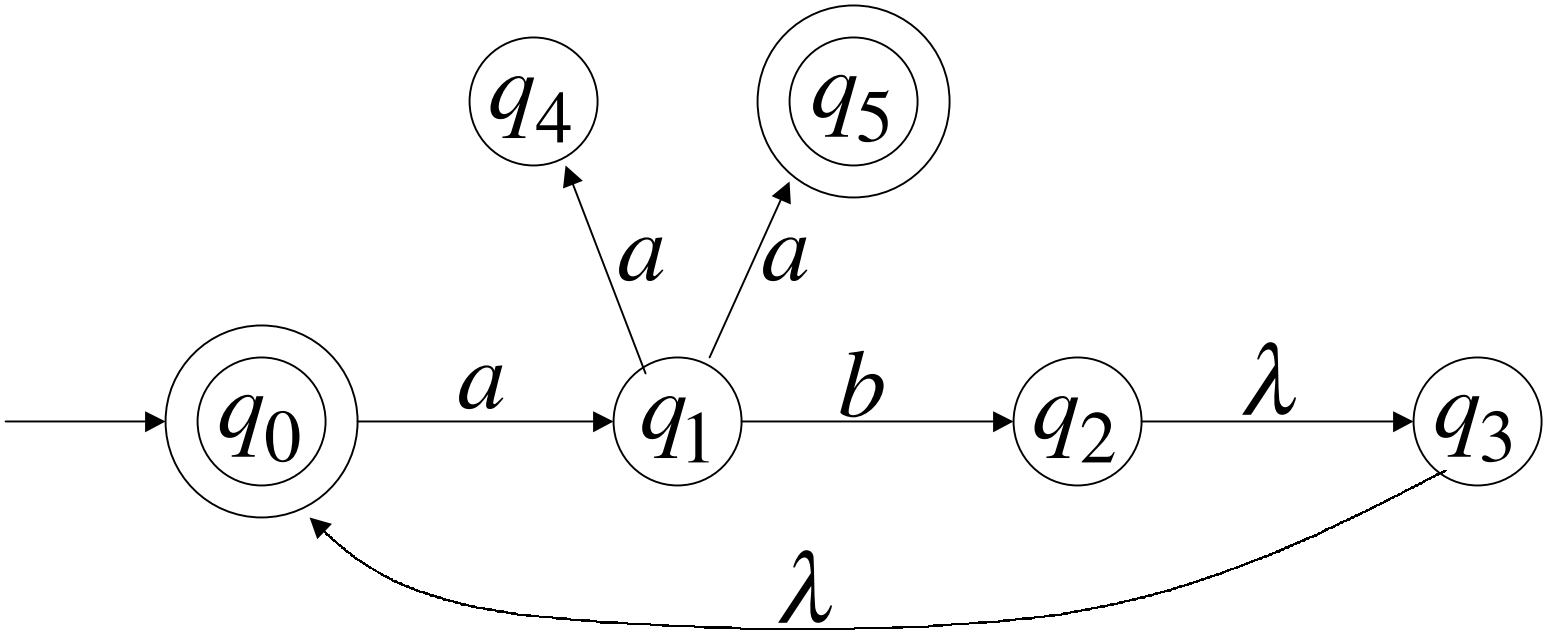
$$\delta(q_0, 0) = \varnothing$$

$$\delta(q_0, \lambda) = \{q_0, q_2\}$$

$$\delta(q_1, 0) = \{q_0, q_2\}$$

$$\delta(q_2, 1) = \varnothing$$

Extended Transition Function δ^*



$$\delta^*(q_0, a) = \{q_1\}$$

$$\delta^*(q_0, aa) = \{q_4, q_5\}$$

$$\delta^*(q_0, ab) = \{q_2, q_3, q_0\}$$

Formally

It holds

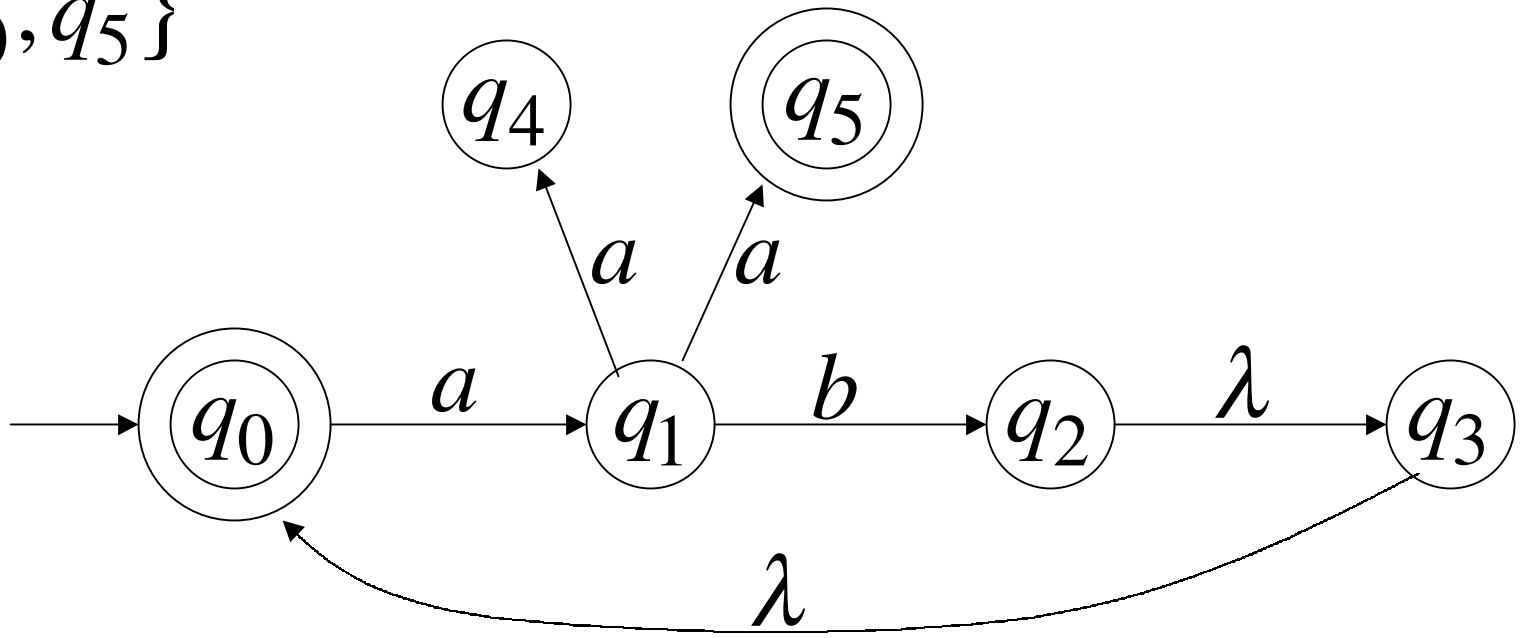
$$q_j \in \delta^*(q_i, w)$$

if and only if

there is a walk from q_i to q_j
with label w

The Language of an NFA M

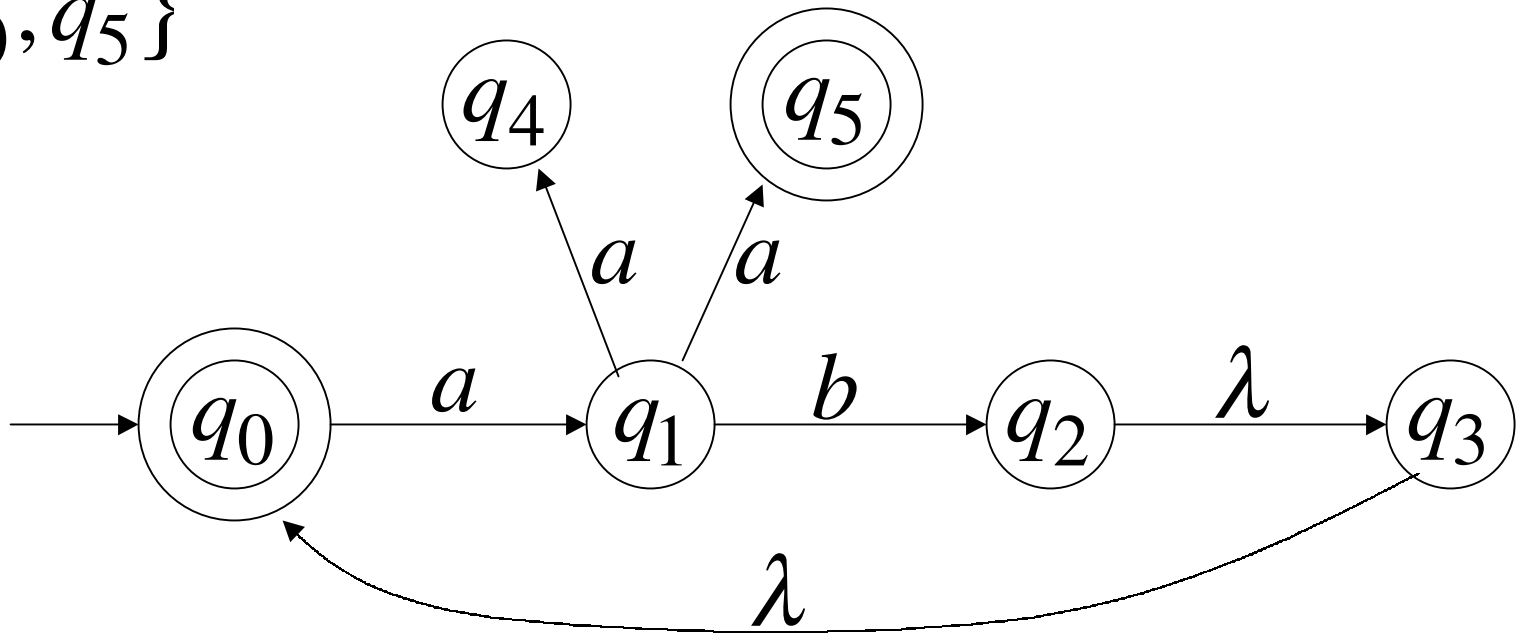
$F = \{q_0, q_5\}$



$\delta^*(q_0, aa) = \{q_4, \underline{q_5}\}$ $\quad aa \in L(M)$

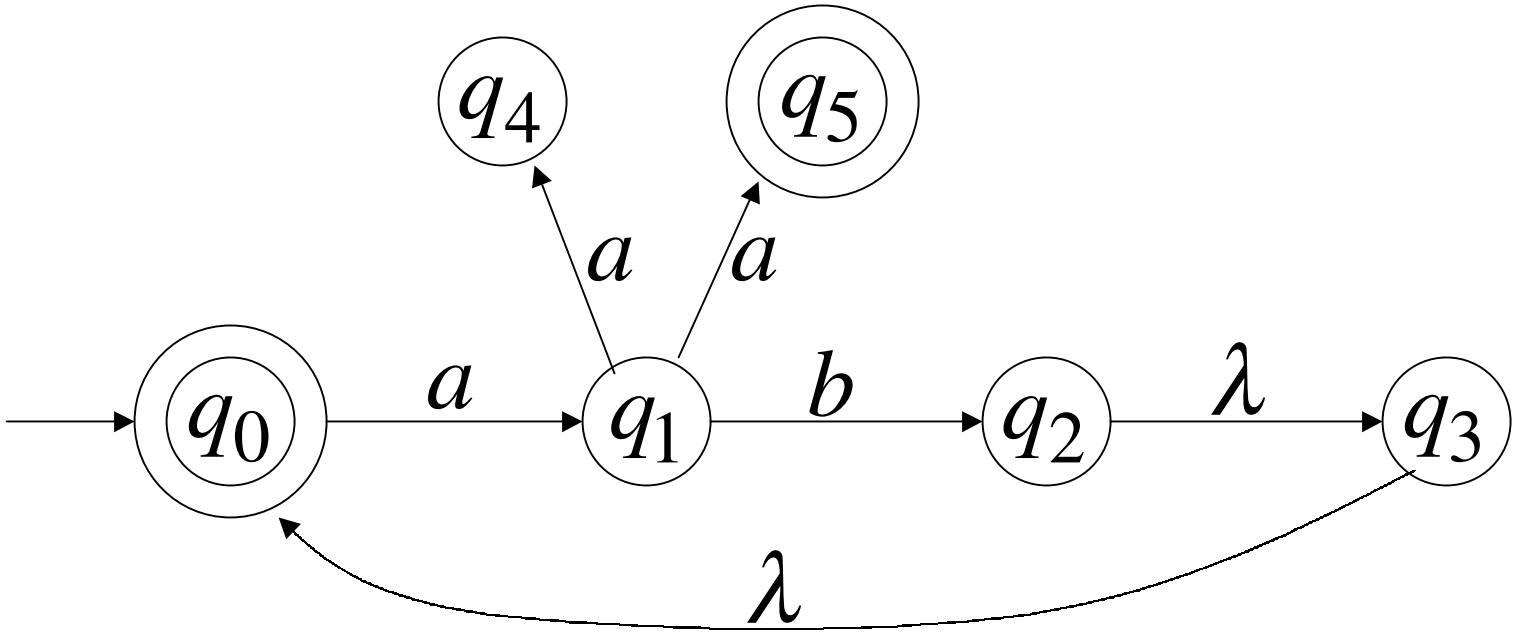
$\delta^*(q_0, ab) = \{q_2, q_3, \underline{q_0}\}$ $\quad ab \in L(M)$

$F = \{q_0, q_5\}$



$\delta^*(q_0, abaa) = \{q_4, \underline{q_5}\}$ $\quad aaba \in L(M)$

$\delta^*(q_0, aba) = \{q_1\}$ $\quad aba \notin L(M)$



$$L(M) = \{aa\} \cup \{ab\}^* \cup \{ab\}^+\{aa\}$$

Formally

The language $L(M)$
accepted by NFA M is:

$$L(M) = \{w_1, w_2, w_3, \ldots\}$$

where

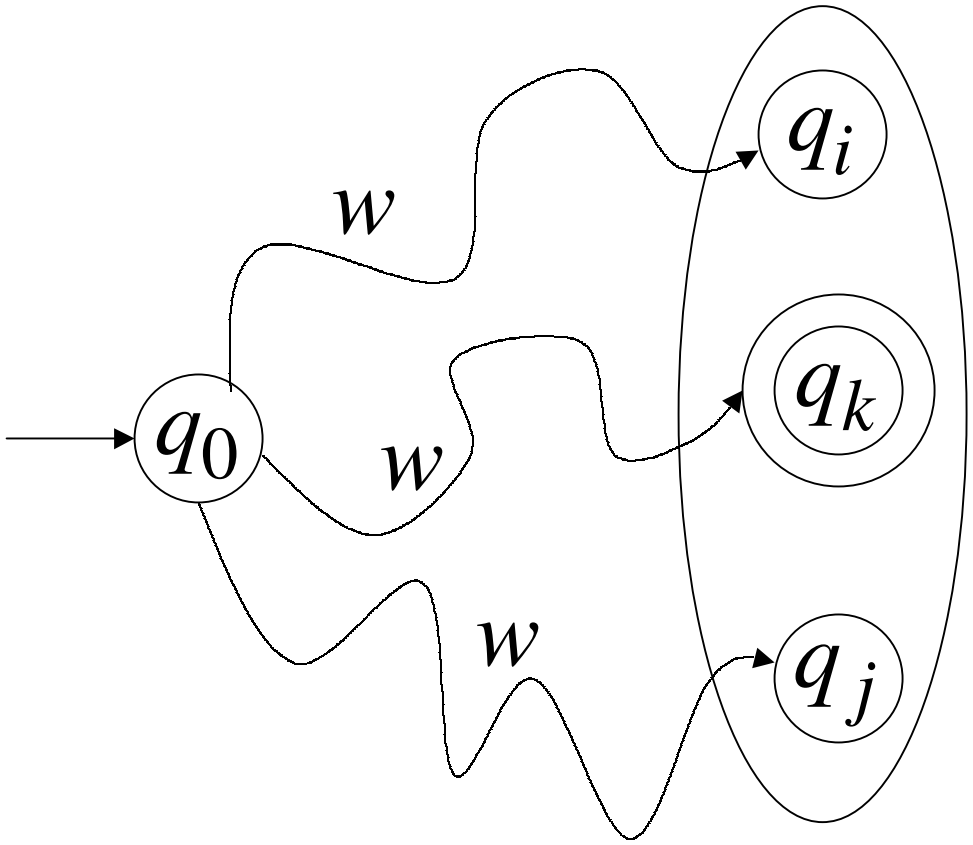
$$\delta^*(q_0, w) = \{q_i, q_j, \ldots\}$$

and there is some

$$q_k \in F \qquad \text{(final state)}$$

$w \in L(M)$

$\delta^*(q_0, w)$



$q_k \in F$

Equivalence of NFAs and DFAs

For DFAs or NFAs:

Machine M_1 is equivalent to M_2

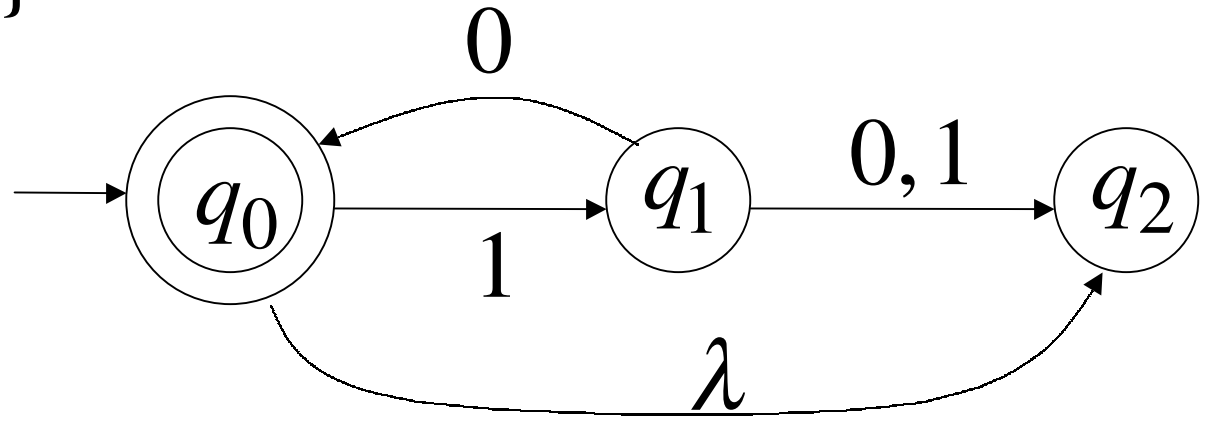
if

$$L(M_1) = L(M_2)$$

Example

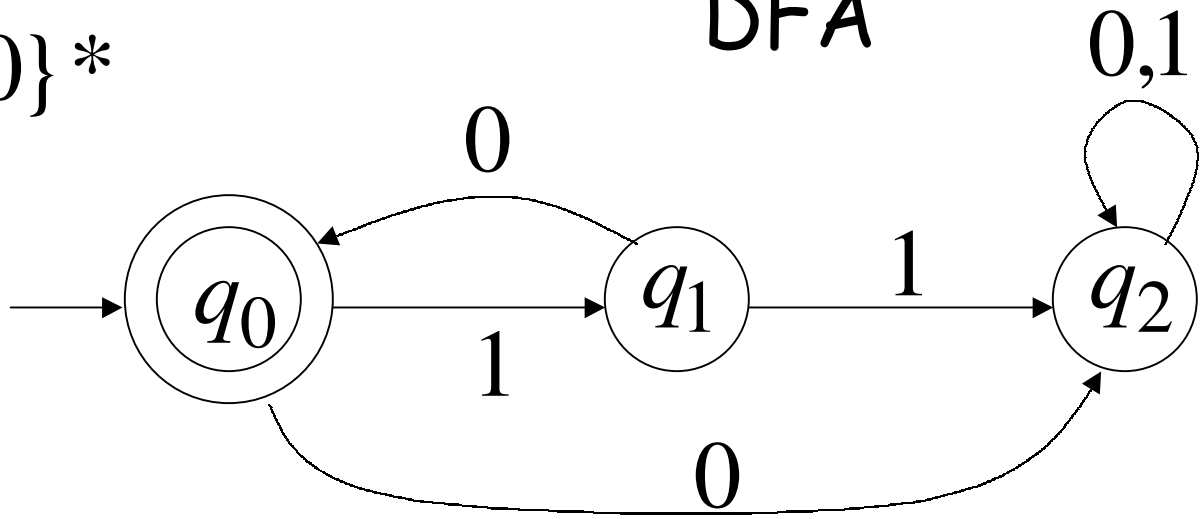
$L(M_1) = \{10\}^*$

NFA



$L(M_2) = \{10\}^*$

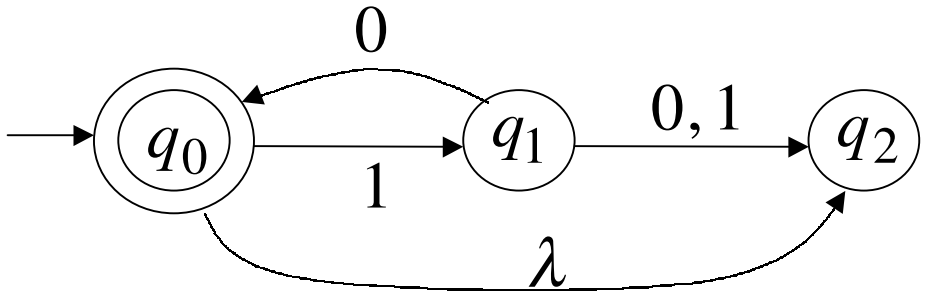
DFA



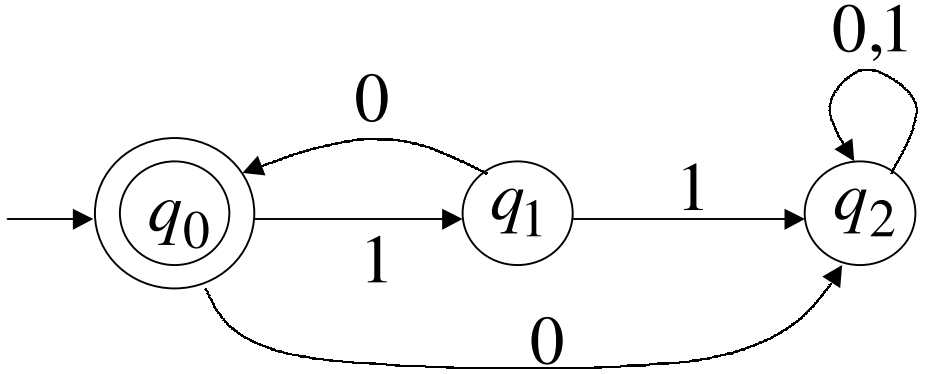
Since $L(M_1) = L(M_2) = \{10\}^*$

machines M_1 and M_2 are equivalent

NFA M_1



DFA M_2



Equivalence of NFAs and DFAs

Every DFA is also an NFA

A language accepted by a DFA will be accepted by an NFA

An NFA is as least as powerful as a DFA

Is an DFA as powerful as an NFA?

Is a DFA as Powerful as an NFA?

Answer: **YES!**

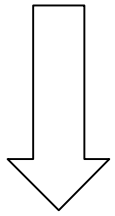
A language accepted by an NFA will be accepted by some DFA

For every NFA there is an equivalent DFA that accepts the same language

NFAs Accept the Regular Languages

For every NFA there is an equivalent DFA

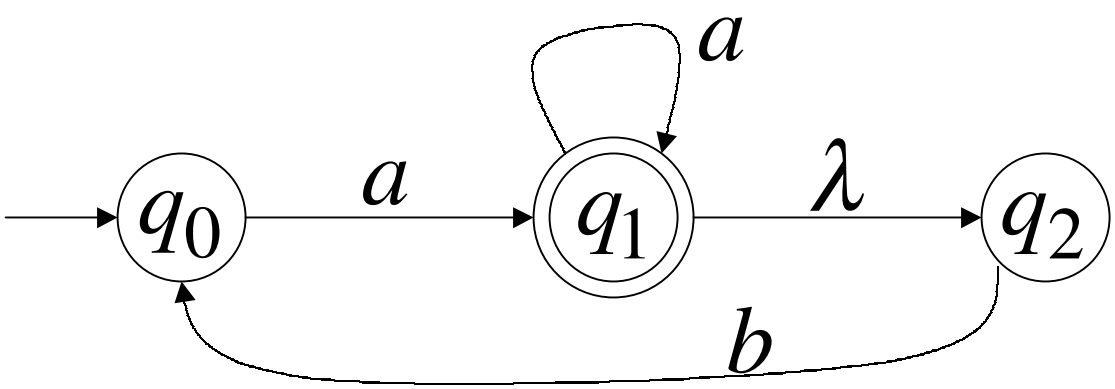
The language accepted by a DFA is regular



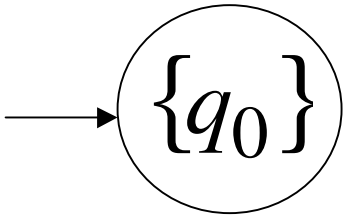
The language accepted by an NFA is regular

NFA to DFA

NFA

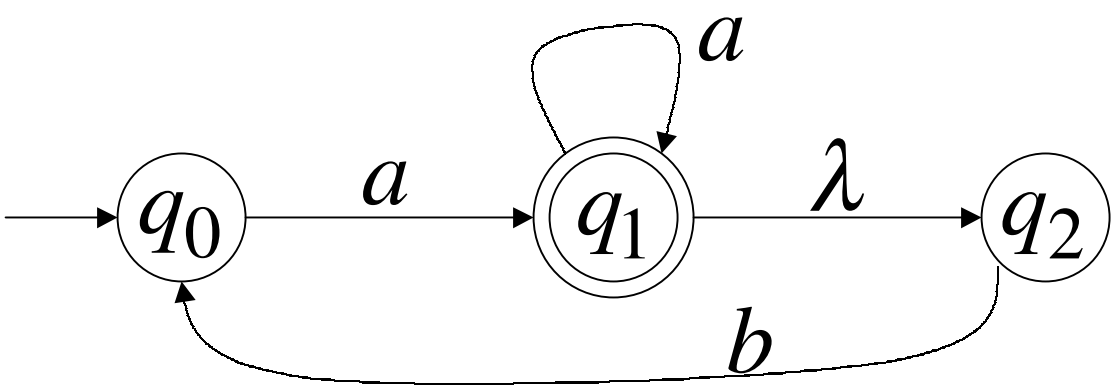


DFA

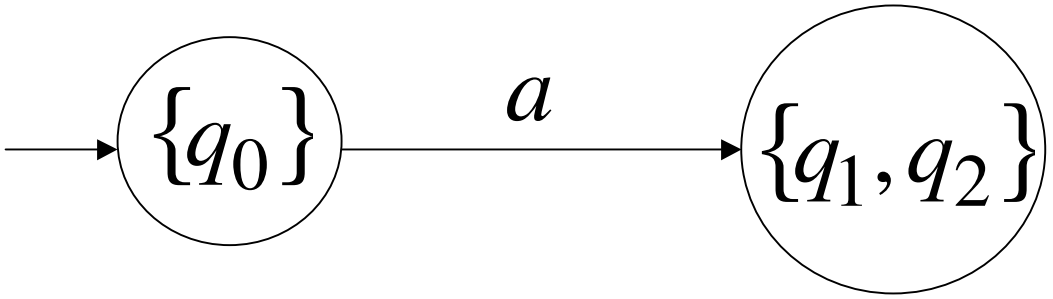


NFA to DFA

NFA

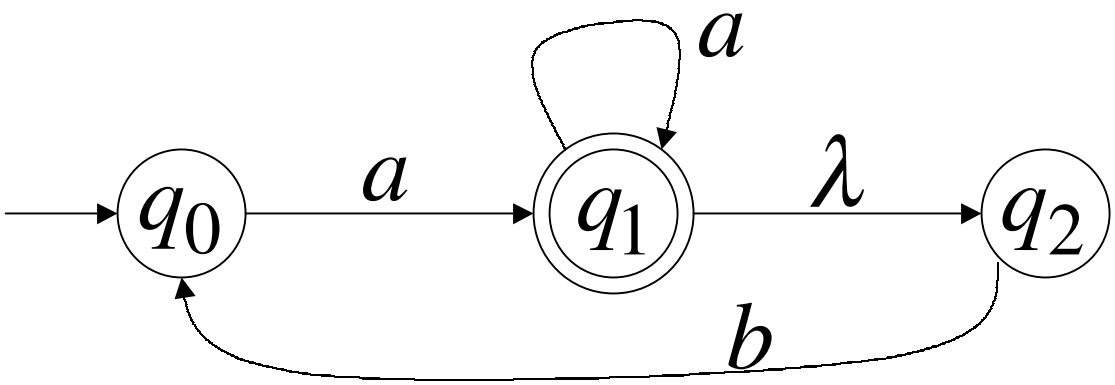


DFA

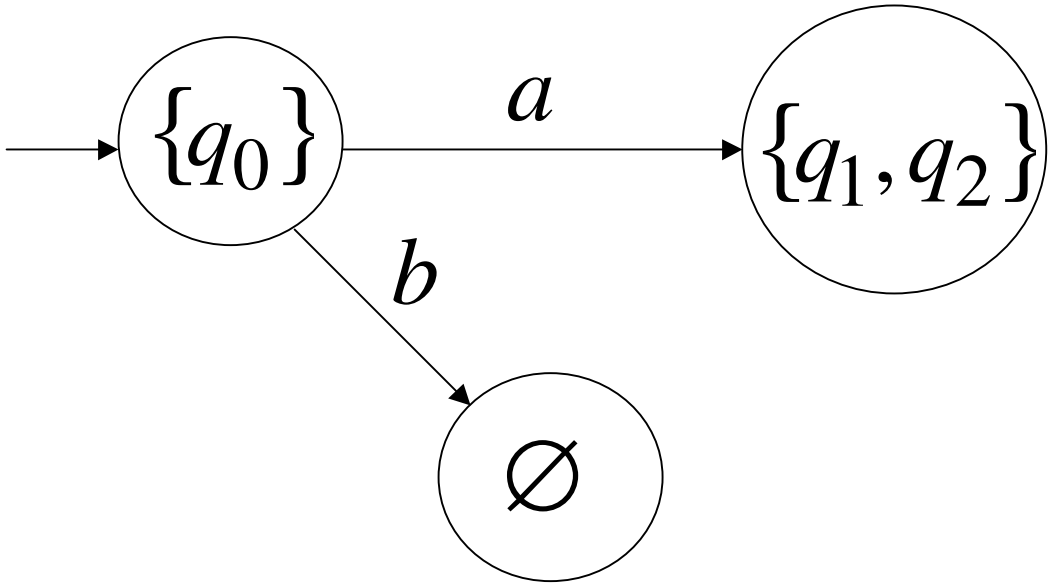


NFA to DFA

NFA

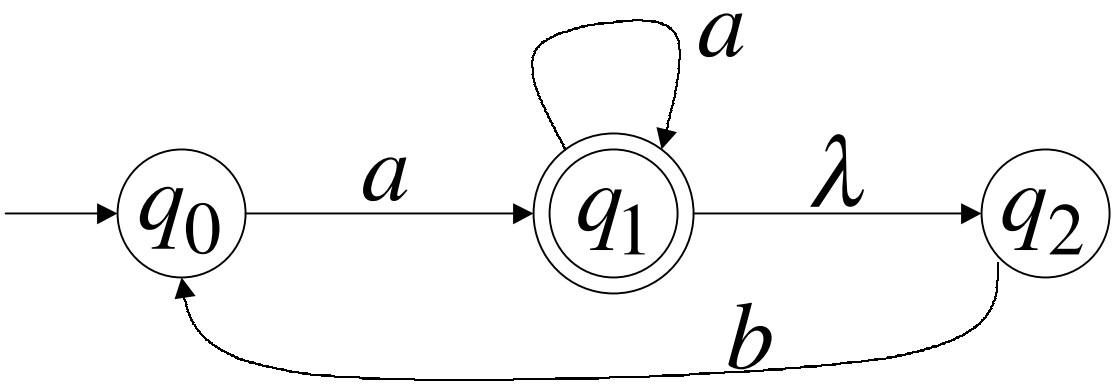


DFA

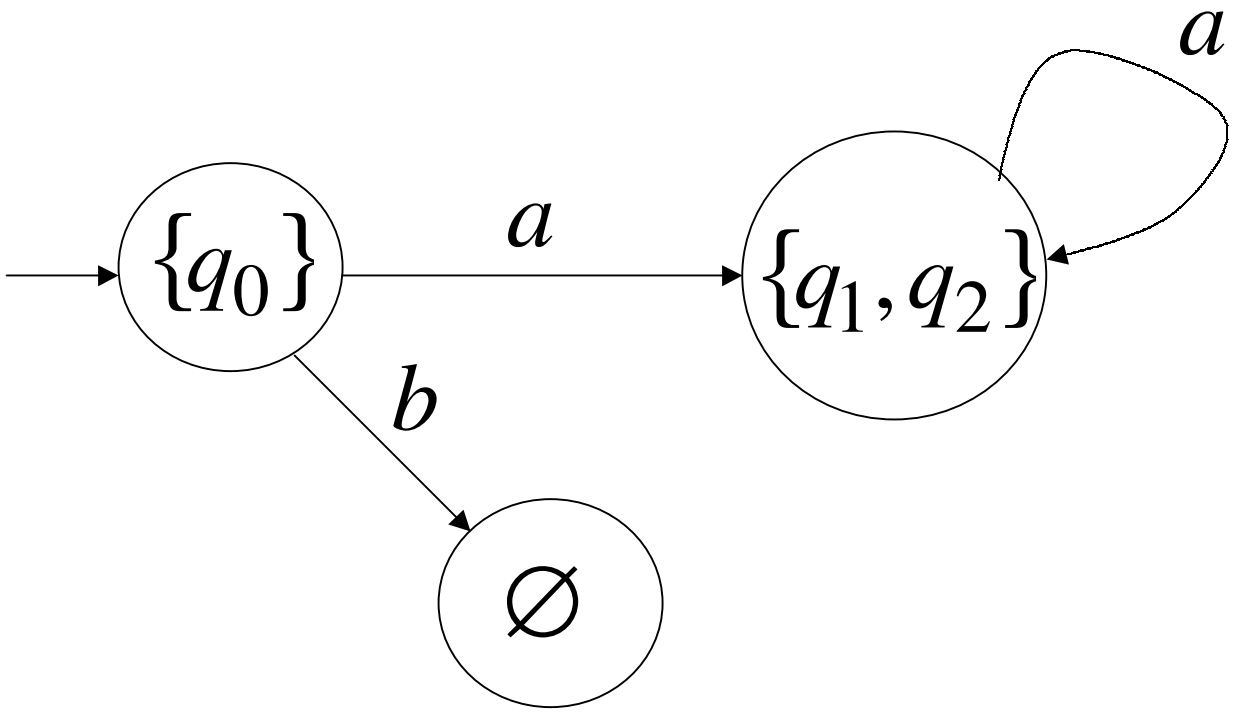


NFA to DFA

NFA

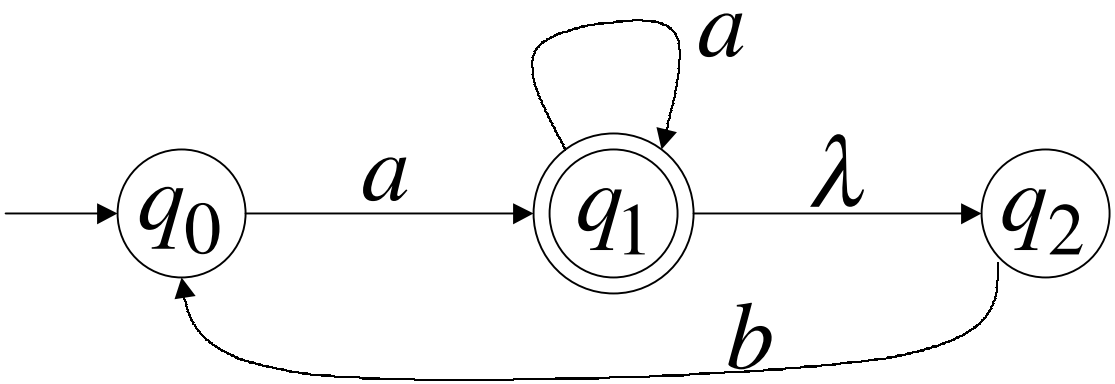


DFA

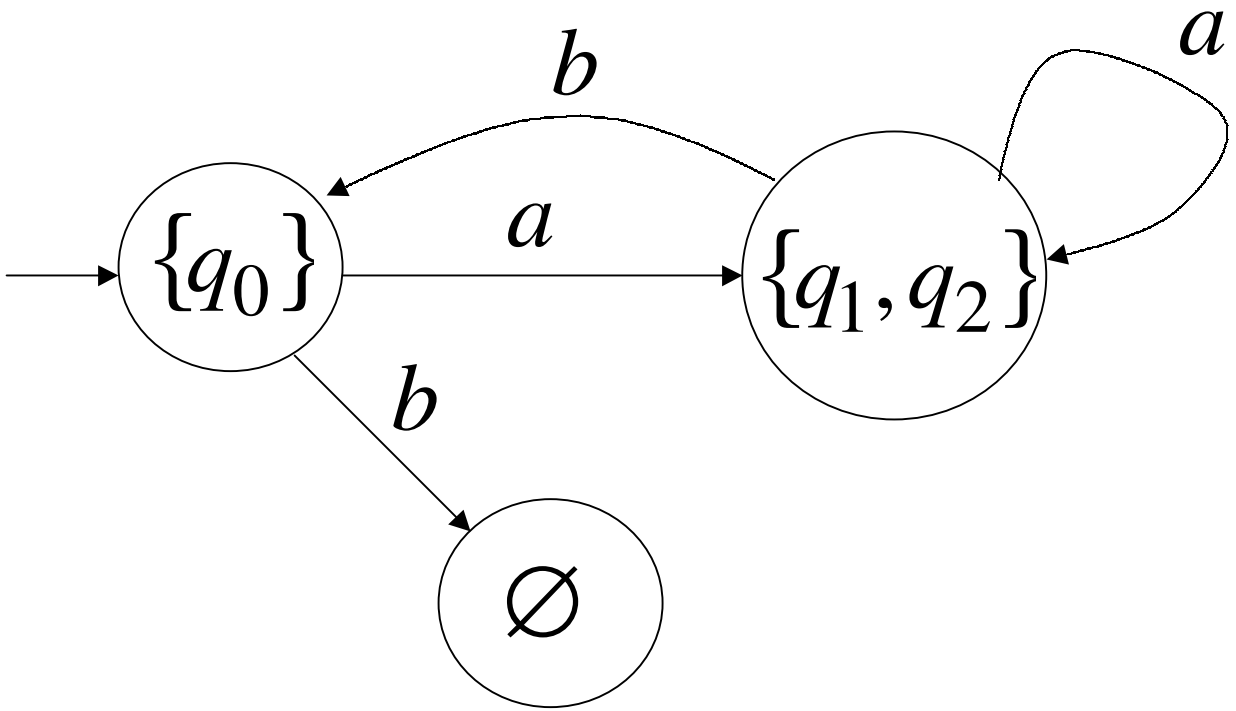


NFA to DFA

NFA

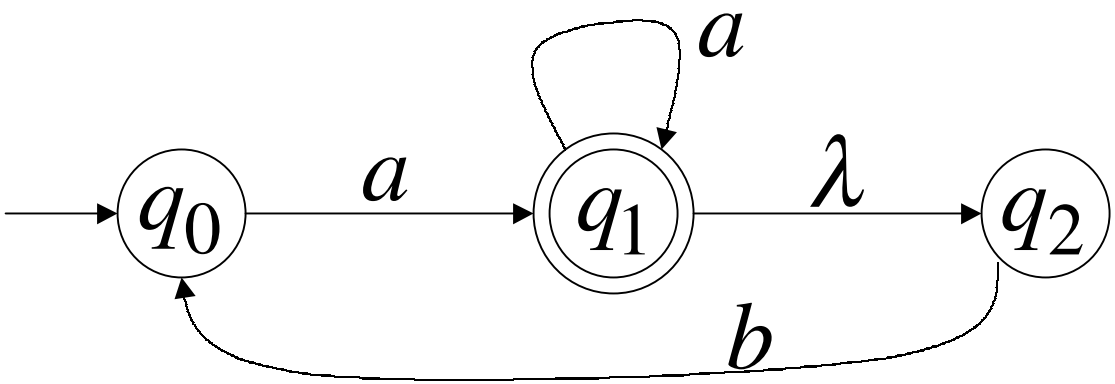


DFA

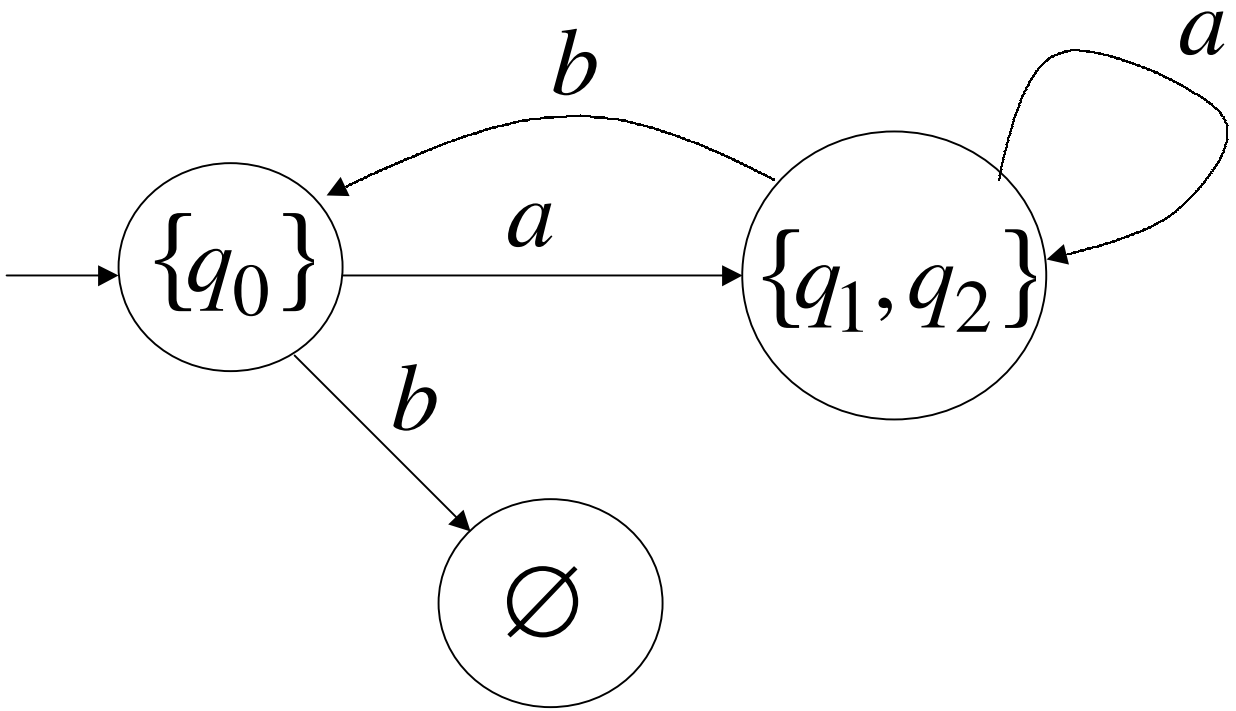


NFA to DFA

NFA

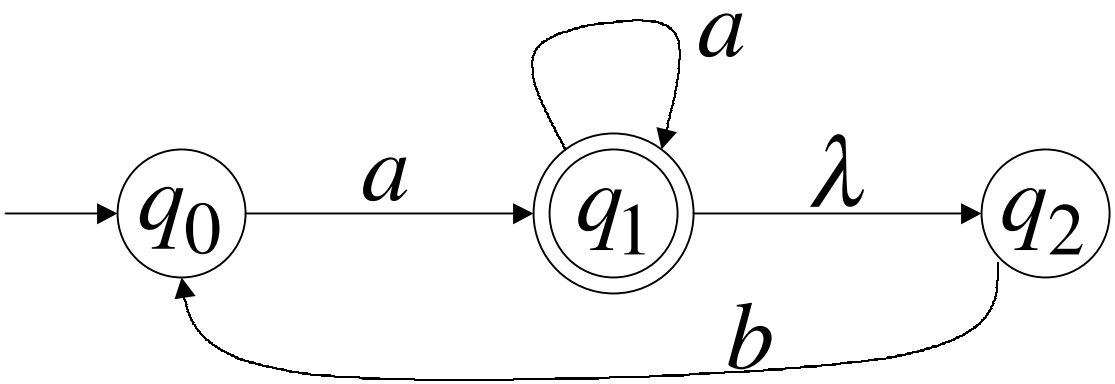


DFA

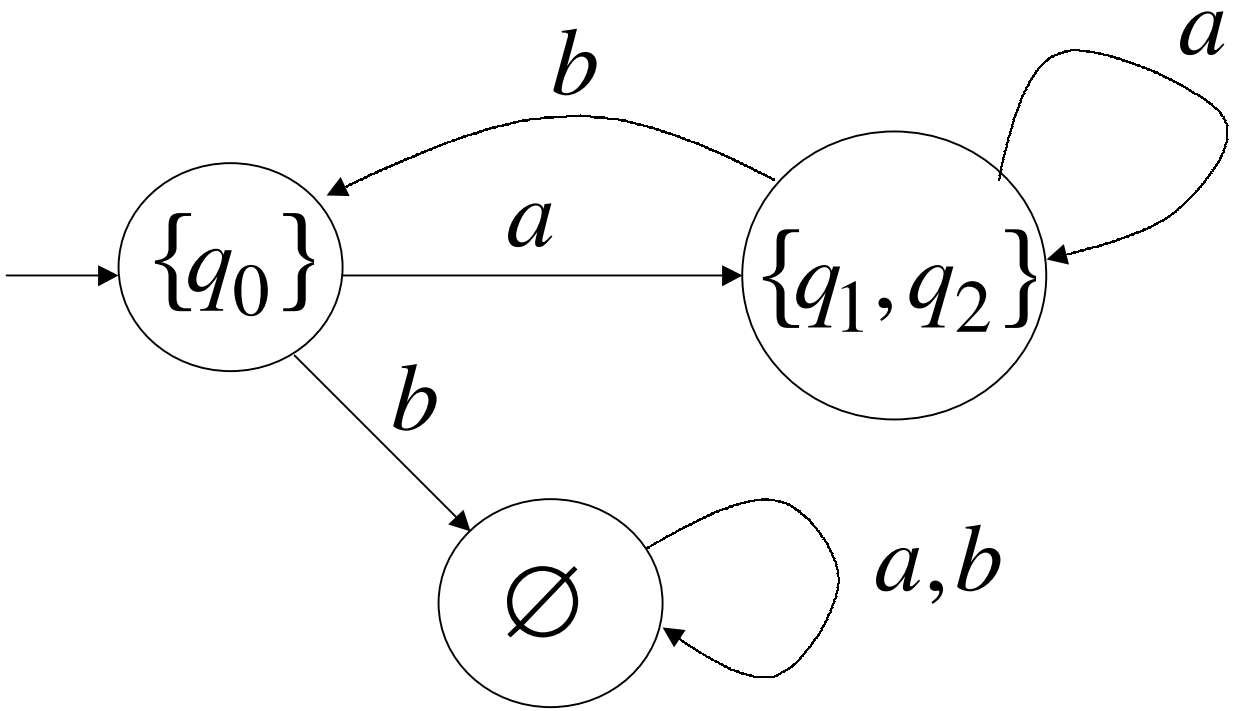


NFA to DFA

NFA

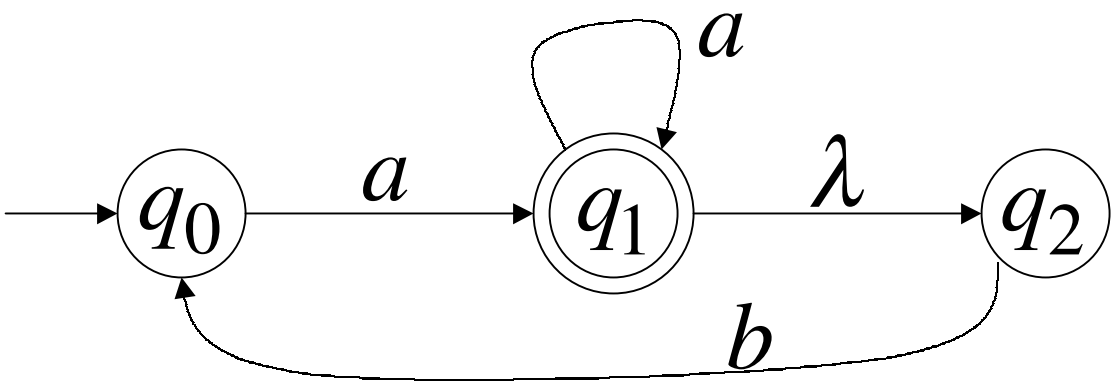


DFA

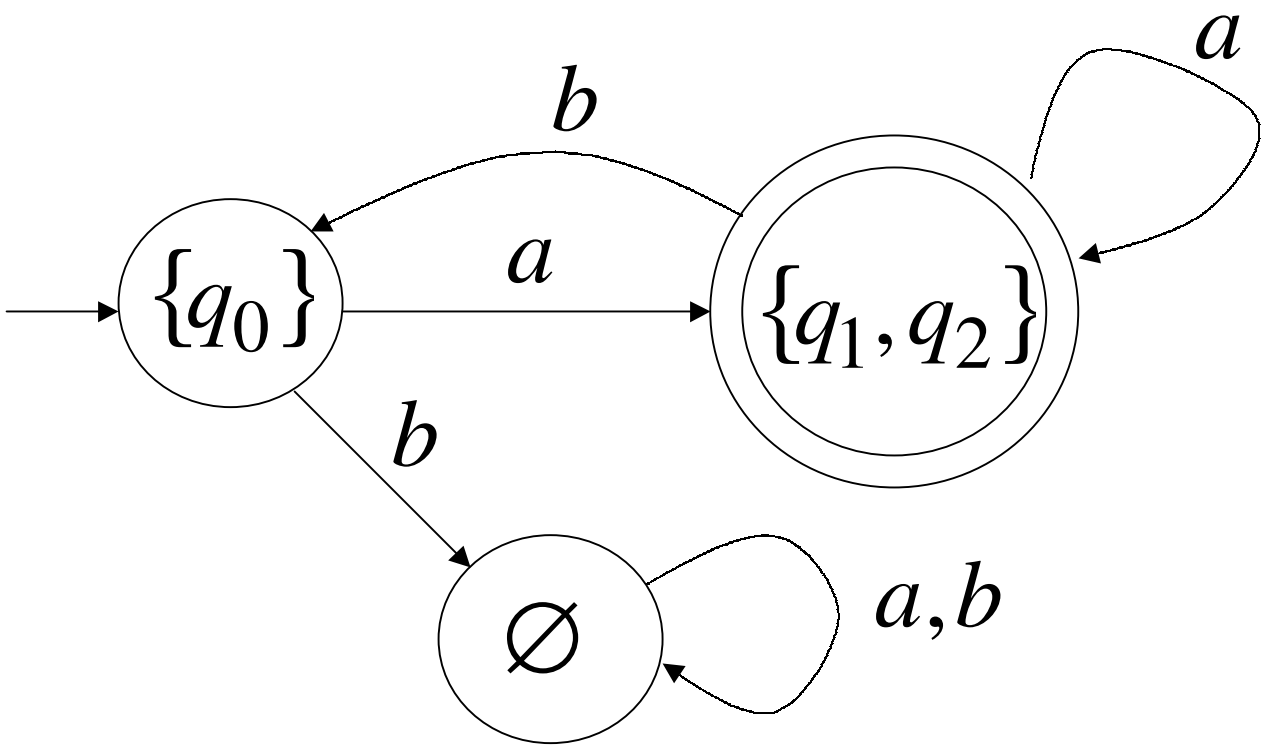


NFA to DFA

NFA



DFA



NFA to DFA Observations

We are given an NFA M

We want to convert it
to an equivalent DFA M'

With $L(M) = L(M')$

If the NFA has states

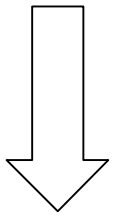
$$q_0, q_1, q_2, \cdots$$

The DFA has states in the powerset

$$\varnothing, \{q_0\}, \{q_1\}, \{q_1, q_2\}, \{q_3, q_4, q_7\}, \ldots.$$

Procedure NFA to DFA

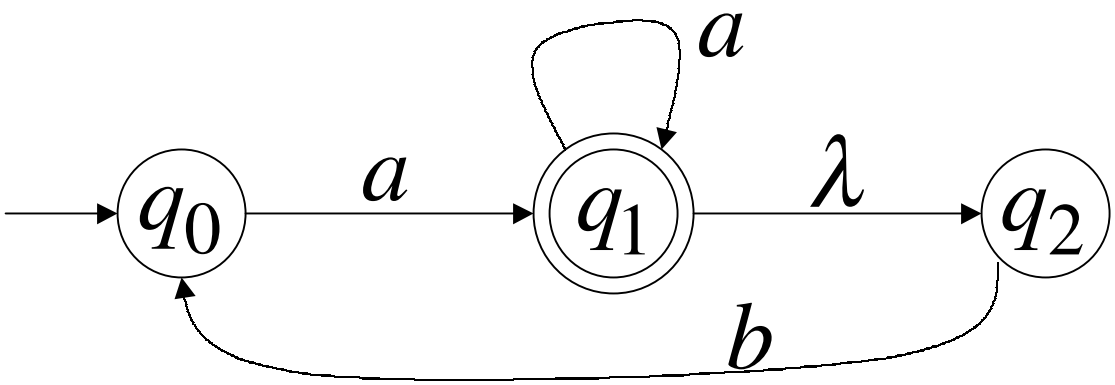
1. Initial state of NFA: q_0



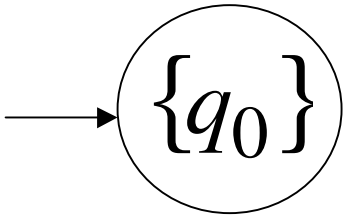
Initial state of DFA: $\{q_0\}$

Example

NFA



DFA



Procedure NFA to DFA

2. For every DFA's state $\{q_i, q_j, \ldots, q_m\}$

Compute in the NFA $\delta^*(q_i, a)$,
$\delta^*(q_j, a)$, ...

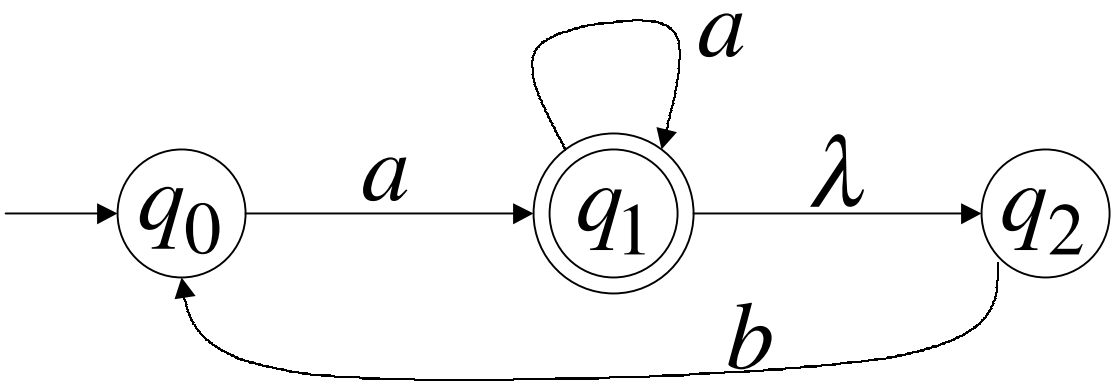
Giving union $\{q'_i, q'_j, \ldots, q'_m\}$

Add a transition

$$\delta(\{q_i, q_j, \ldots, q_m\}, a) = \{q'_i, q'_j, \ldots, q'_m\}$$

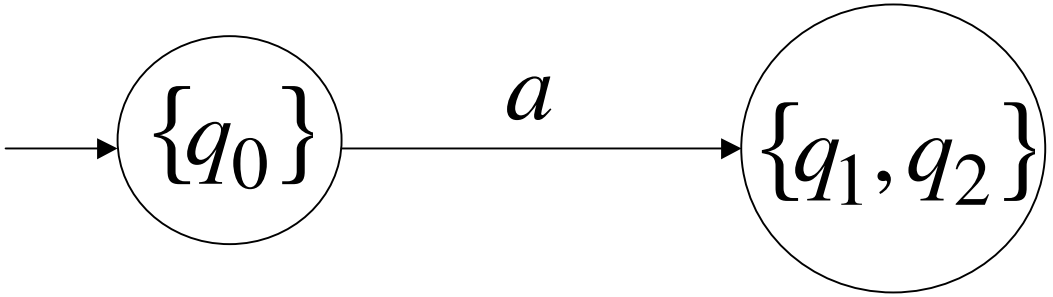
Exampe

NFA



$$\delta^*(q_0, a) = \{q_1, q_2\}$$

DFA



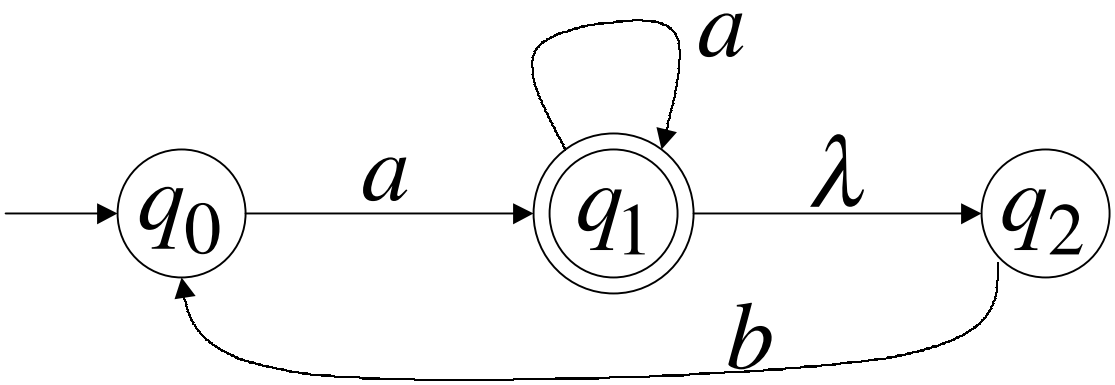
$$\delta(\{q_0\}, a) = \{q_1, q_2\}$$

Procedure NFA to DFA

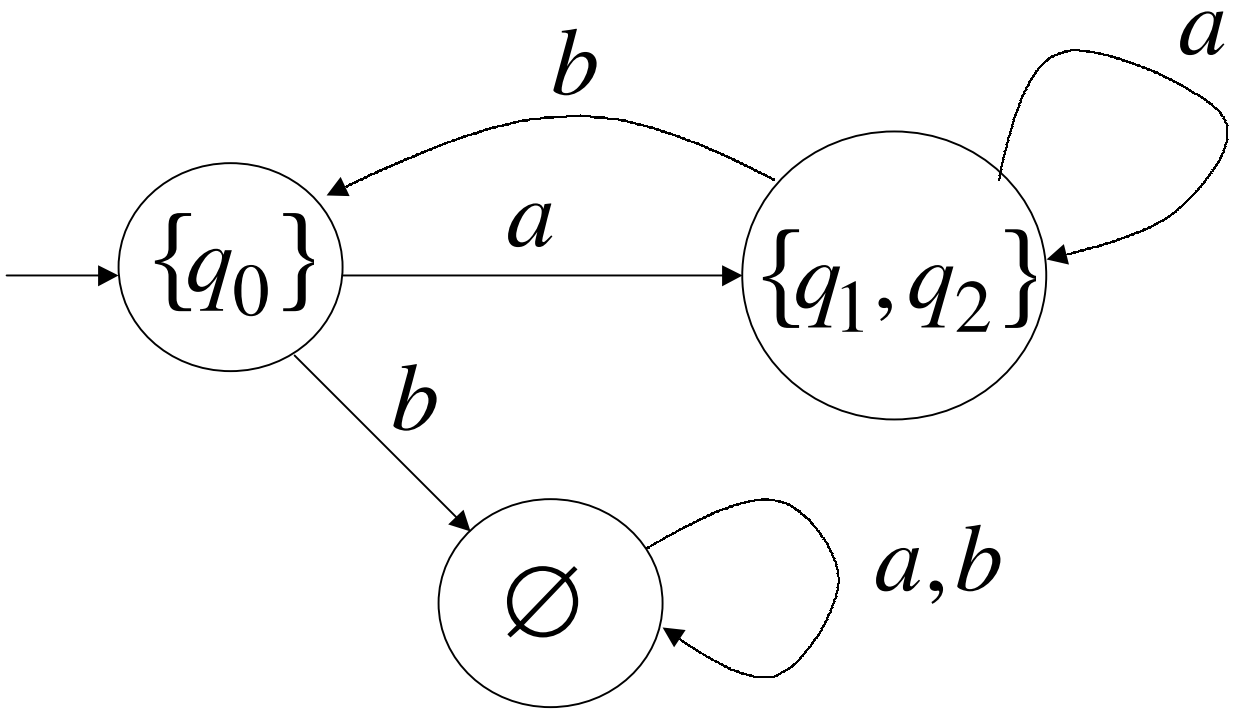
Repeat Step 2 for all letters,
until
no more transitions can be added.

Example

NFA



DFA



Procedure NFA to DFA

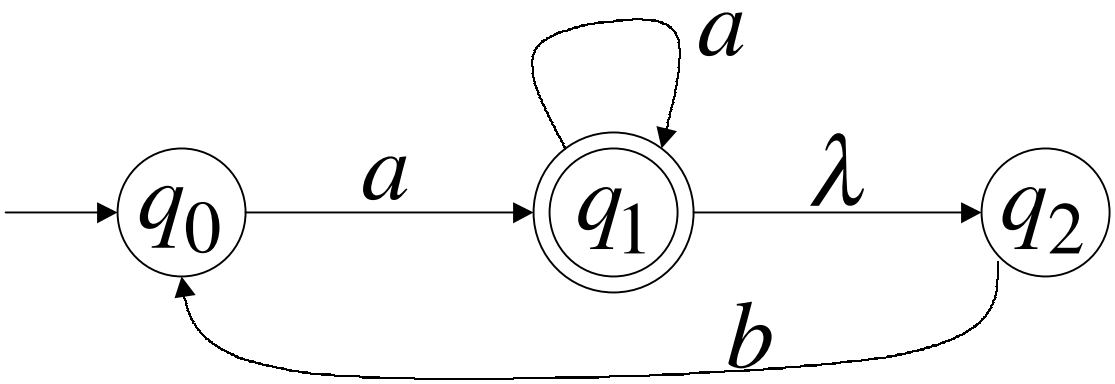
3. For any DFA state $\{q_i, q_j, \ldots, q_m\}$

If some q_j is a final state for the NFA

Then, $\{q_i, q_j, \ldots, q_m\}$
is a final state for the DFA

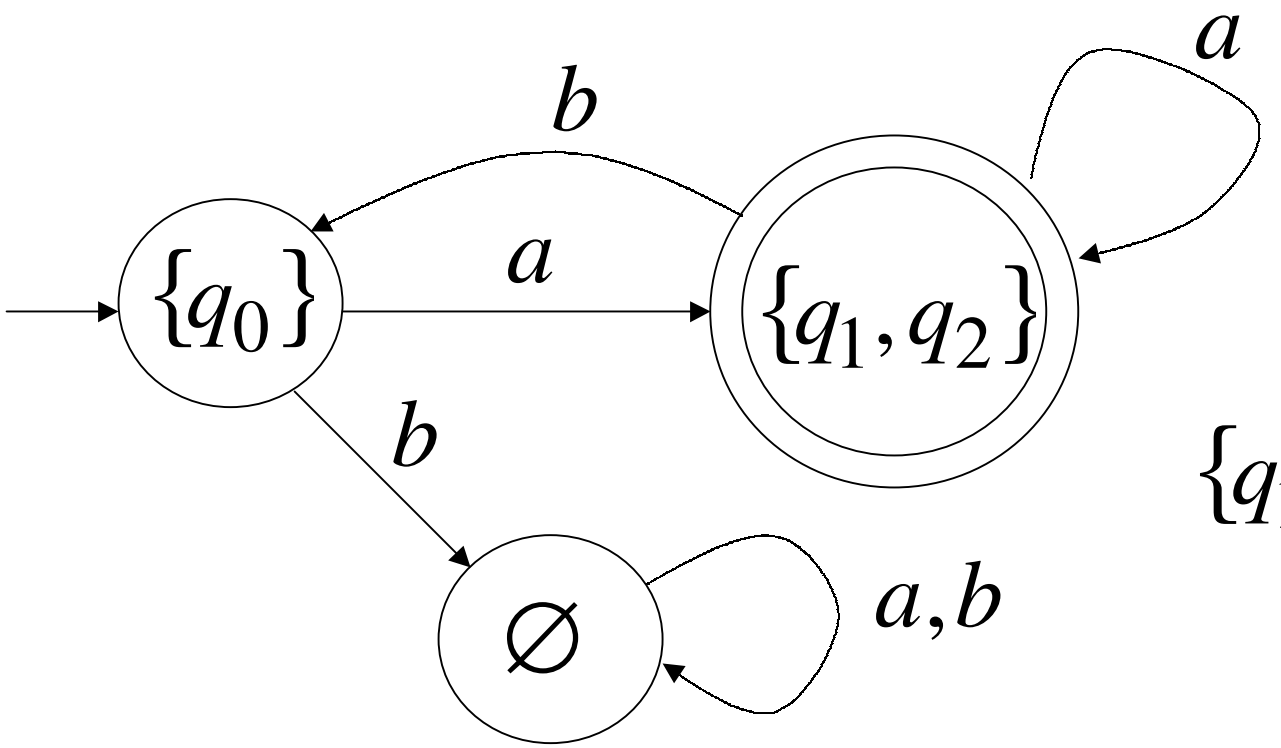
Example

NFA



$q_1 \in F$

DFA



$\{q_1, q_2\} \in F'$

Theorem

Take NFA M

Apply procedure to obtain DFA M'

Then M and M' are equivalent :

$$L(M) = L(M')$$