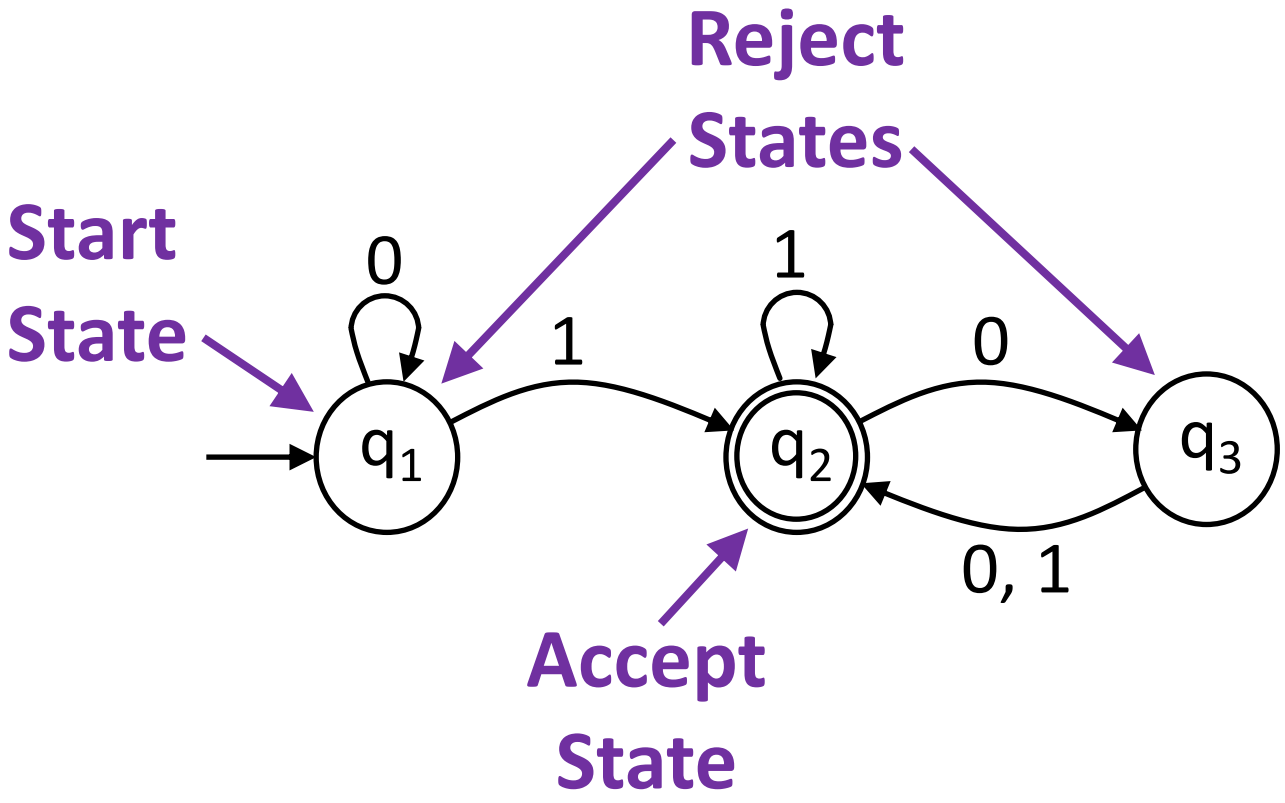
Finite Automata
CSCI 338

Deterministic Finite Automaton (DFA)

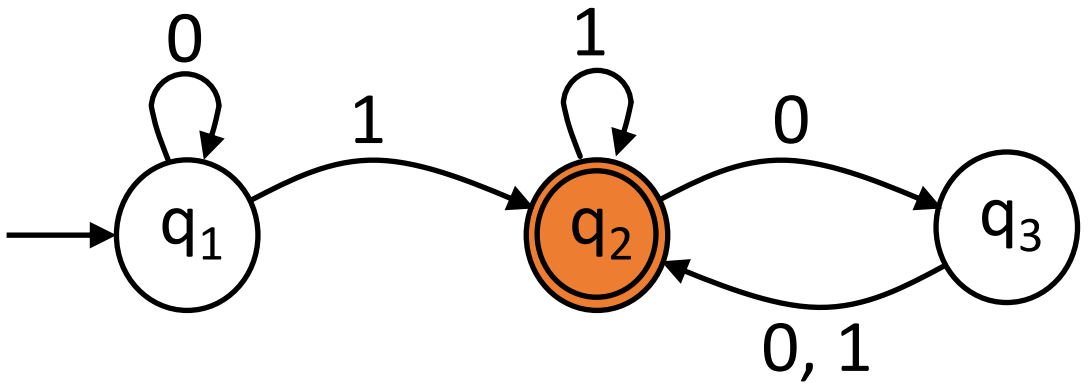


DFAs either accept or reject strings.

Deterministic Finite Automaton (DFA)

DFA string processing:

1. Start at start state.
2. Select first character in string.
3. Update state by following transition that corresponds to character.
4. Select next character in string.
5. Repeat step 3 and 4 until no characters remain.
6. If final state is accept state, accept. Otherwise, reject.

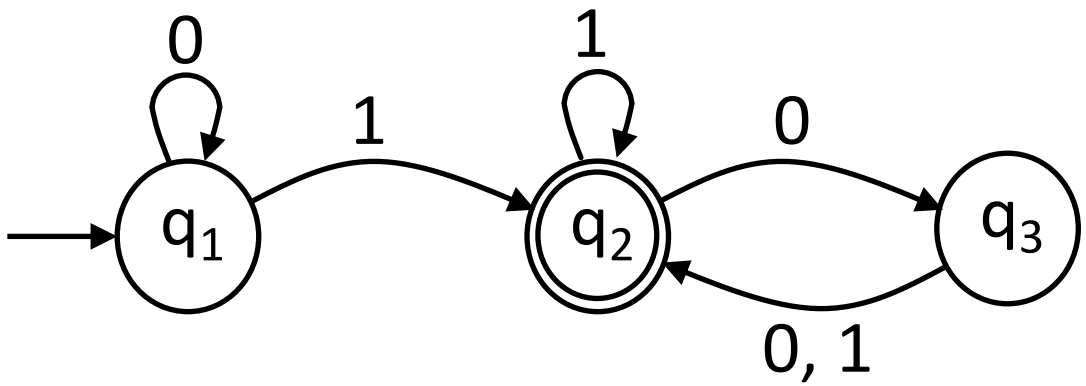


DFAs either accept or reject strings.

Given string $\omega = 01101$, does this DFA **accept** or reject?

DFA Formal Definition

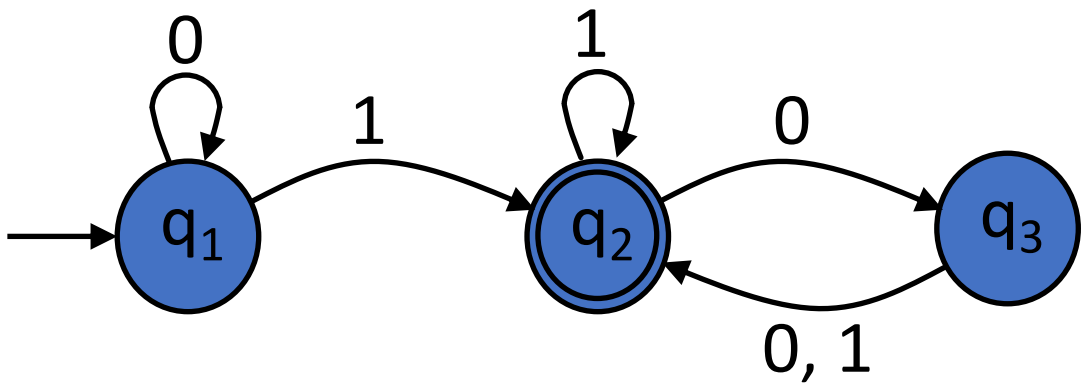
DFAs consist of:



DFA Formal Definition

DFAs consist of:

1. Finite set of states, Q.

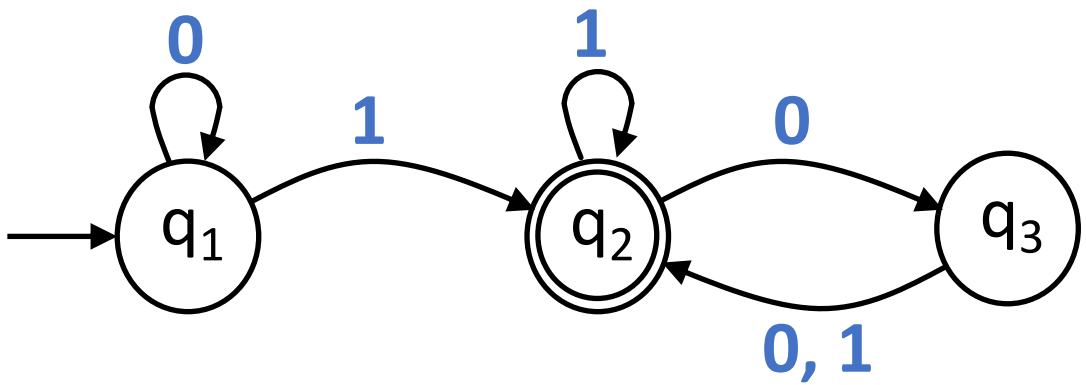


DFA Formal Definition

DFAs consist of:

1. Finite set of states, Q.
2. Finite alphabet, $\boldsymbol{\Sigma}$.

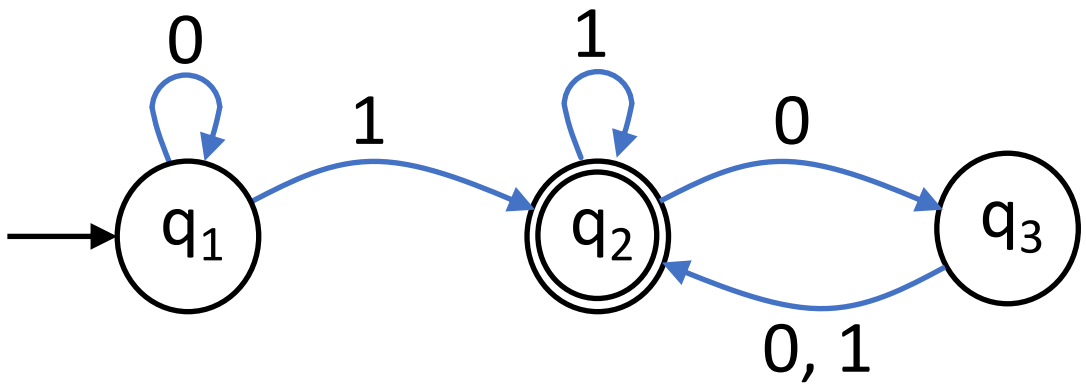
Σ consists of the transition characters (i.e. characters in the strings the DFA processes).



DFA Formal Definition

DFAs consist of:

1. Finite set of states, Q.
2. Finite alphabet, Σ.
3. Transition function, $\boldsymbol{\delta: Q \times \Sigma \rightarrow Q}$.

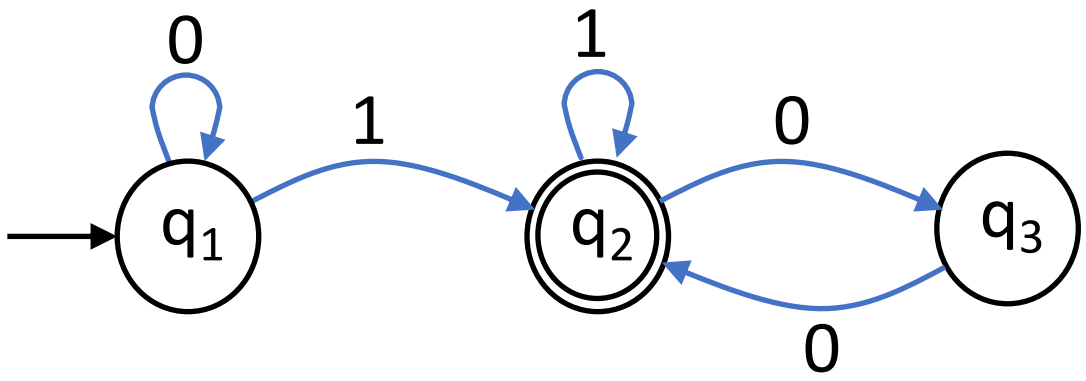


DFA Formal Definition

DFAs consist of:

1. Finite set of states, Q.
2. Finite alphabet, Σ.
3. Transition function, $\boldsymbol{\delta: Q \times \Sigma \rightarrow Q}$.

$\delta: Q \times \Sigma \rightarrow Q$ means that ***every*** state needs to handle ***every*** element of the alphabet.

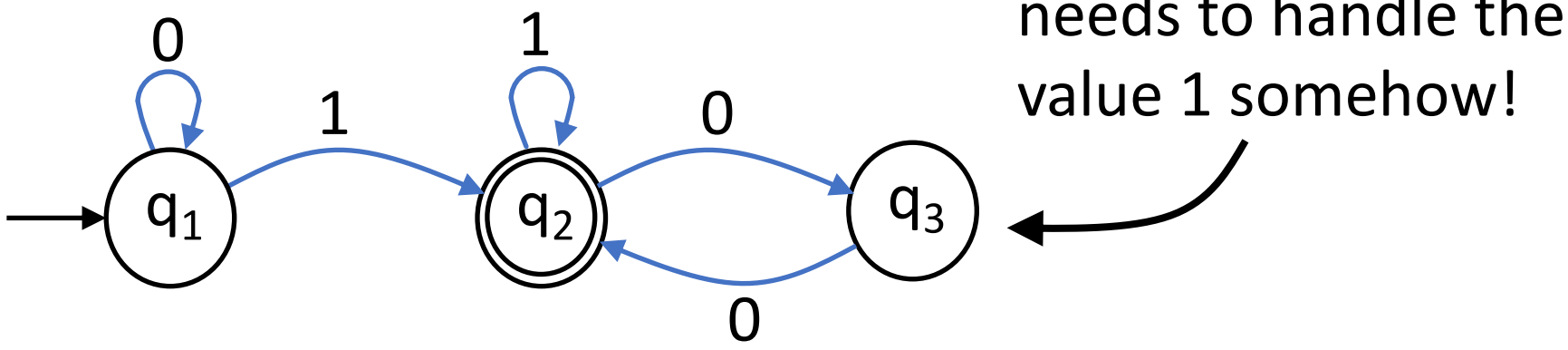


DFA Formal Definition

DFAs consist of:

1. Finite set of states, Q.
2. Finite alphabet, Σ.
3. Transition function, $\boldsymbol{\delta: Q \times \Sigma \rightarrow Q}$.

$\delta: Q \times \Sigma \rightarrow Q$ means that ***every*** state needs to handle ***every*** element of the alphabet.



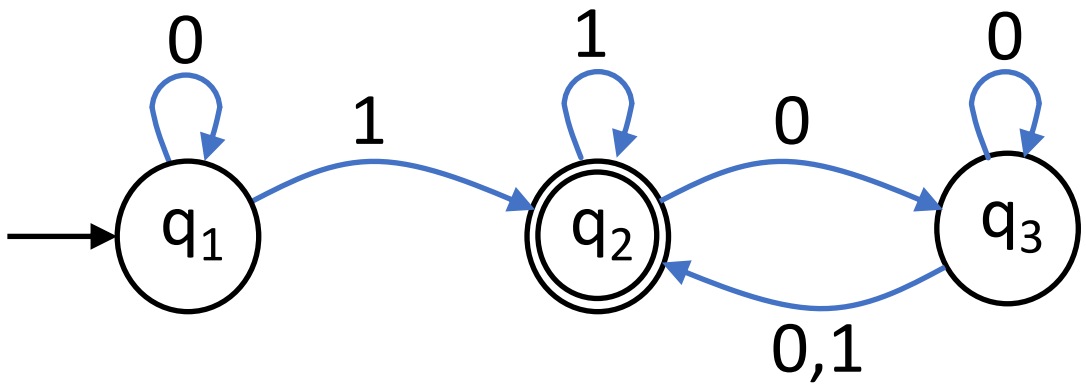
DFA Formal Definition

DFAs consist of:

1. Finite set of states, Q.
2. Finite alphabet, Σ.
3. Transition function, $\boldsymbol{\delta: Q \times \Sigma \rightarrow Q}$.

$\delta: Q \times \Sigma \rightarrow Q$ means that ***every*** state needs to handle ***every*** element of the alphabet.

??

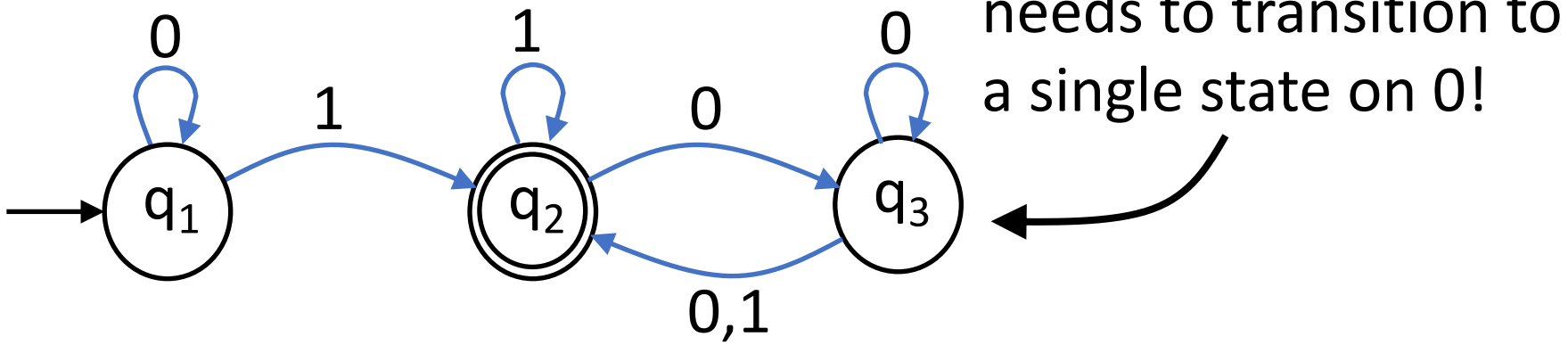


DFA Formal Definition

DFAs consist of:

1. Finite set of states, Q.
2. Finite alphabet, Σ.
3. Transition function, $\boldsymbol{\delta: Q \times \Sigma \rightarrow Q}$.

$\delta: Q \times \Sigma \rightarrow Q$ means that ***every*** state needs to handle ***every*** element of the alphabet.

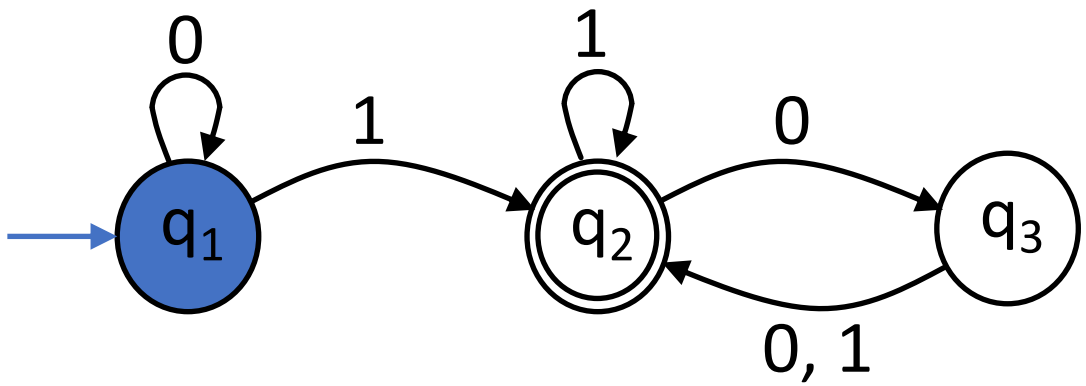


DFA Formal Definition

DFAs consist of:

1. Finite set of states, Q.
2. Finite alphabet, Σ.
3. Transition function, $\delta: Q \times \Sigma \rightarrow Q$.
4. Start state, $\boldsymbol{q_0} \in Q$.

Exactly one start state needed.

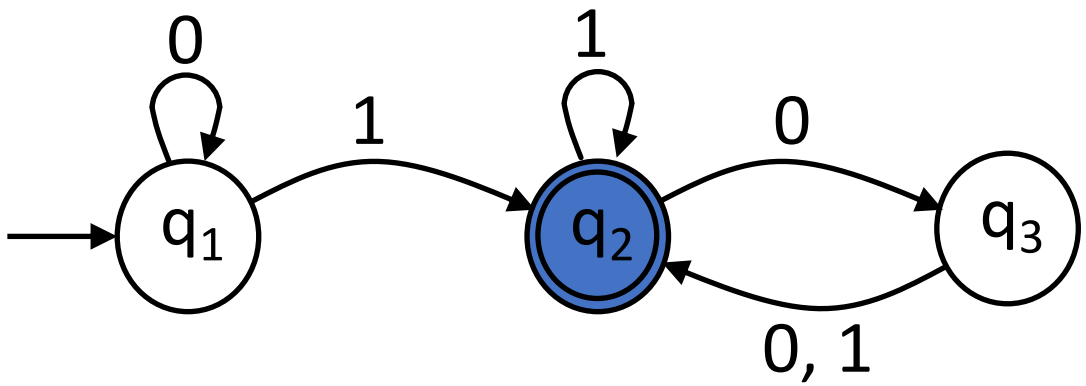


DFA Formal Definition

DFAs consist of:

1. Finite set of states, Q.
2. Finite alphabet, Σ.
3. Transition function, $\delta: Q \times \Sigma \rightarrow Q$.
4. Start state, $q_0 \in Q$.
5. Set of accept states, $\boldsymbol{F} \subseteq Q$.

F is allowed to equal Q or be empty.



DFA Formal Definition

DFAs consist of:

1. Finite set of states, Q.
2. Finite alphabet, Σ.
3. Transition function, $\delta: Q \times \Sigma \rightarrow Q$.
4. Start state, $q_0 \in Q$.
5. Set of accept states, $F \subseteq Q$.

$Q = \{q_1, q_2, q_3\}$

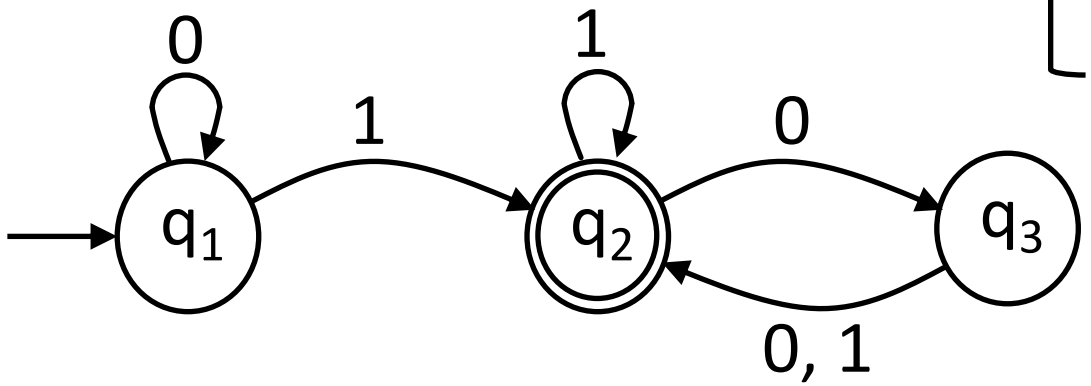
$\Sigma = \{0,1\}$

δ:

	0	1
q_1	q_1	q_2
q_2	q_3	q_2
q_3	q_2	q_2

Start state = q_1

$F = \{q_2\}$



DFA Language

Definitions:

The set of all strings A that a DFA M accepts is called its language, $L(M) = A$.

M recognizes A.

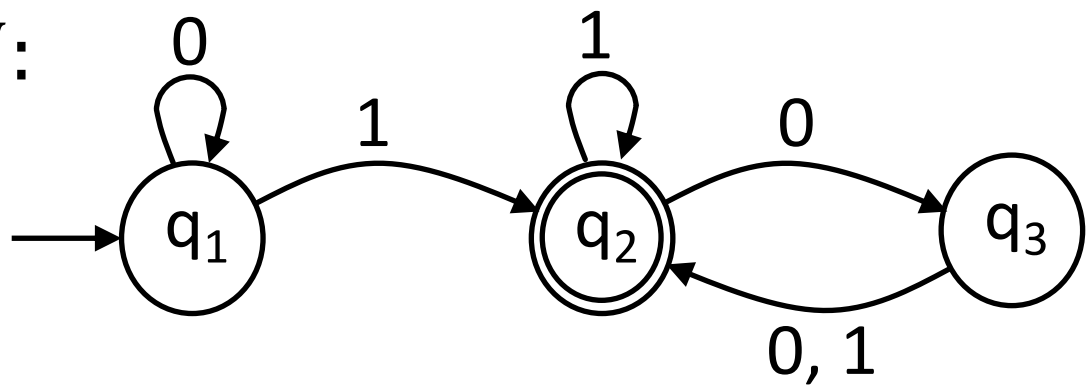
DFA Language

Definitions:

The set of all strings A that a DFA M accepts is called its language, $L(M) = A$.

M recognizes A.

M:



$L(M) =$ $\{\omega: \omega$ contains at least one 1 and an even number of 0s following the final 1$\}$

DFA Language

Definitions:

The set of all strings A that a DFA M accepts is called its language, $L(M) = A$.

M recognizes A.

A language is called a regular language if some DFA recognizes it.

DFA Language

Definitions:

The set of all strings A that a DFA M accepts is called its language, $L(M) = A$.

M recognizes A.

A language is called a regular language if some DFA recognizes it.

How do you prove a language is regular?

DFA Language

Definitions:

The set of all strings A that a DFA M accepts is called its language, $L(M) = A$.

M recognizes A.

A language is called a regular language if some DFA recognizes it.

How do you prove a language is regular?

Make a DFA that recognizes it.

DFA Language

Definitions:

The set of all strings A that a DFA M accepts is called its language, $L(M) = A$.

M recognizes A.

A language is called a regular language if some DFA recognizes it.

How do you prove a language is regular?

Make a DFA that recognizes it.

Set of regular languages are “things we can do” with DFAs.

DFA Practice

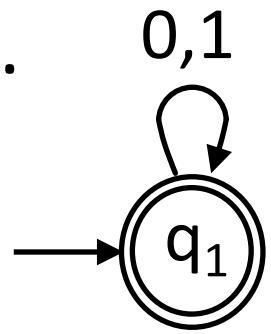
Prove that the following languages are regular:

1. Set of all strings over {0,1}.

DFA Practice

Prove that the following languages are regular:

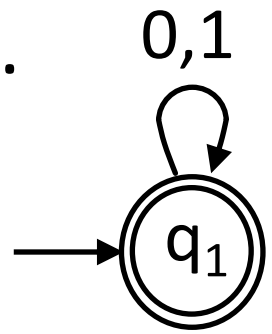
1. Set of all strings over {0,1}.



DFA Practice

Prove that the following languages are regular:

1. Set of all strings over {0,1}.

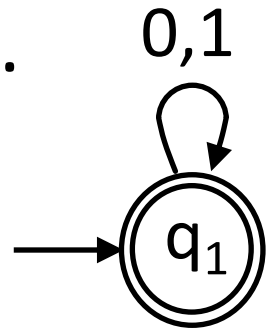


2. Set of all strings with an even number of 0s.

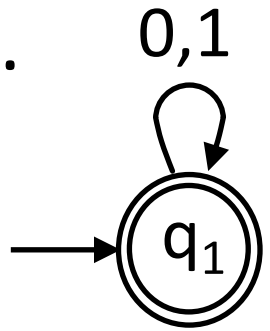
DFA Practice

Prove that the following languages are regular:

1. Set of all strings over {0,1}.



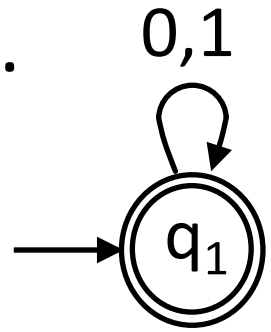
2. Set of all strings with an even number of 0s.



DFA Practice

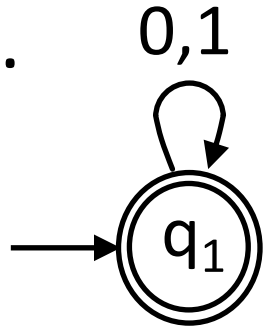
Prove that the following languages are regular:

1. Set of all strings over {0,1}.



2. Set of all strings with an even number of 0s.

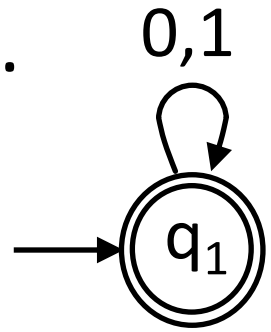
The set of **all** strings A that a DFA M accepts is called its language, $L(M) = A$.



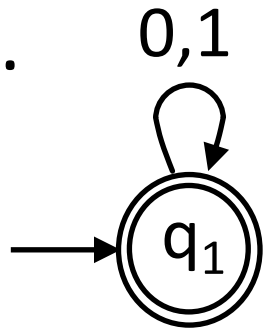
DFA Practice

Prove that the following languages are regular:

1. Set of all strings over {0,1}.



2. Set of all strings with an even number of 0s.



The set of **all** strings A that a DFA M accepts is called its language, $L(M) = A$.

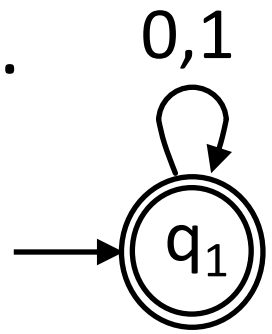
DFA Language Rules:

1. If the DFA accepts it, it is in the language.
2. If it is in the language, the DFA must accept it.

DFA Practice

Prove that the following languages are regular:

1. Set of all strings over {0,1}.

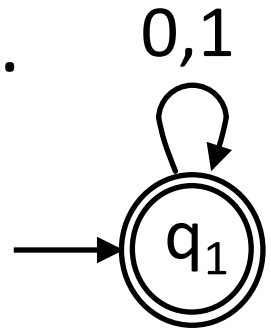


2. Set of all strings with an even number of 0s.

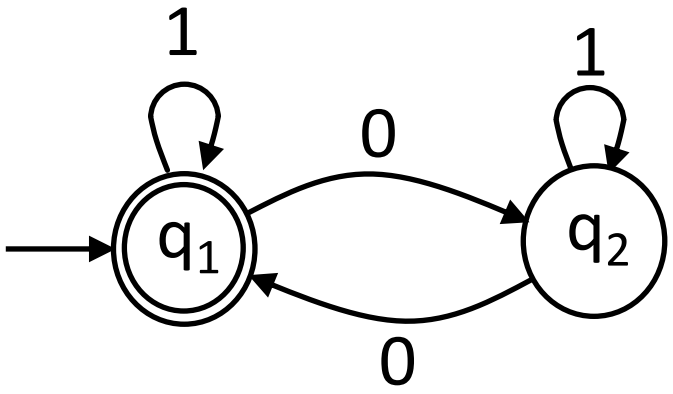
DFA Practice

Prove that the following languages are regular:

1. Set of all strings over {0,1}.



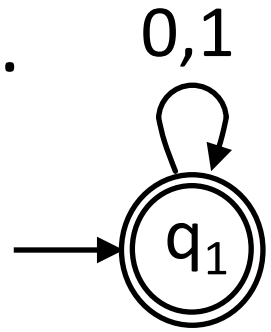
2. Set of all strings with an even number of 0s.



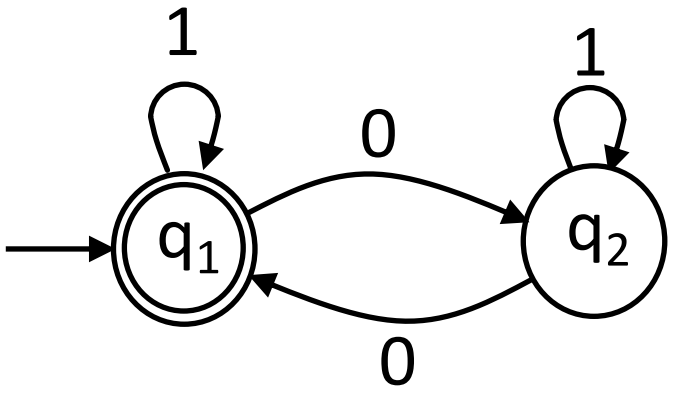
DFA Practice

Prove that the following languages are regular:

1. Set of all strings over {0,1}.



2. Set of all strings with an even number of 0s.

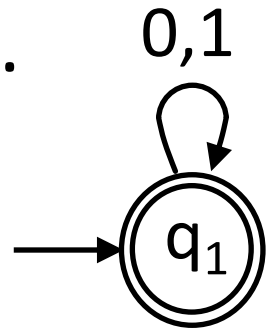


3. Set of all strings that contain the substring: 10.

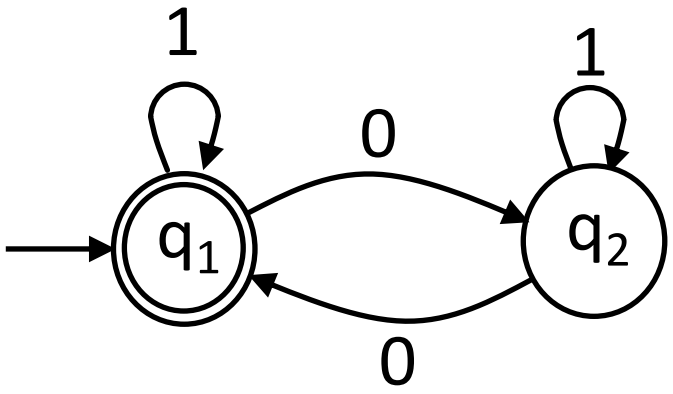
DFA Practice

Prove that the following languages are regular:

1. Set of all strings over {0,1}.



2. Set of all strings with an even number of 0s.



3. Set of all strings that contain the substring: 10.