BITS Pilani
Pilani Campus

Deterministic Finite Automata (DFA)

Shashank Gupta
Assistant Professor
Department of Computer Science and Information Systems

More Examples

Construct a DFA which accepts the following language

- $L = \{x \in \{0, 1\}^* \mid x$ begins and ends with the same symbol$\}$

DFA = ?

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

Example (Continued…..)

Construct a DFA which accepts the following language

- L = {x ∈ $\{0, 1\}^*$ |x begins and ends with the same symbol}

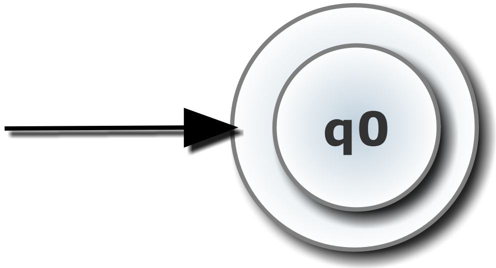
Step 1: Initially, construct the language 'L' over $\Sigma = \{0, 1\}^*$ starting with string of minimum length.

- L = {∈, **0, 1, 00, 11**, **0**1**0**, **1**0**1**, **0**101**0**, **1**100**1**,----}

Example (Continued…..)

Step 2: Secondly construct the FA for the minimal length string (i.e. '∈') from the language 'L'.

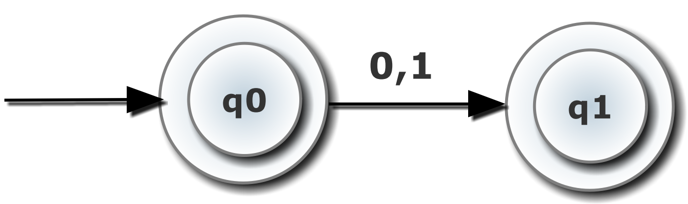
- L = {∈, **0**, **1**, **00**, **11**, **0**1**0**, **1**0**1**, **0**101**0**,----}

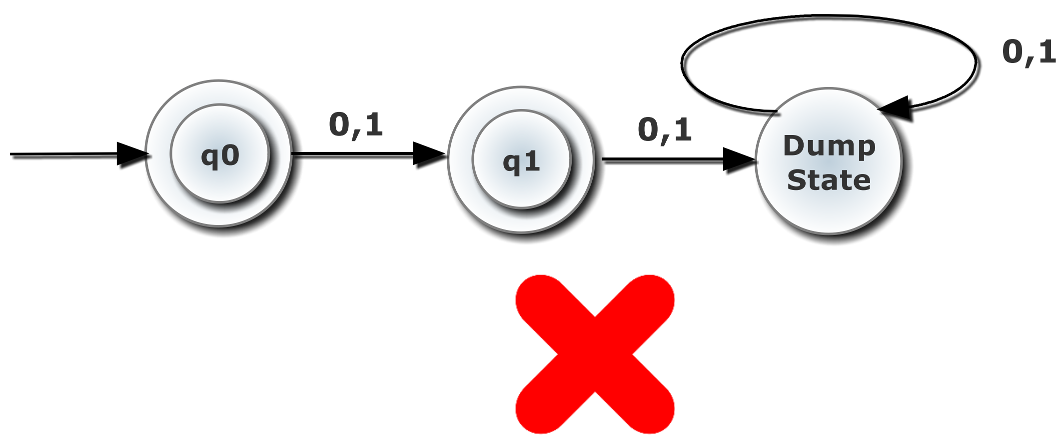


Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

- L = {∈, 0, 1, 00, 11, 010, 101, 01010,----}

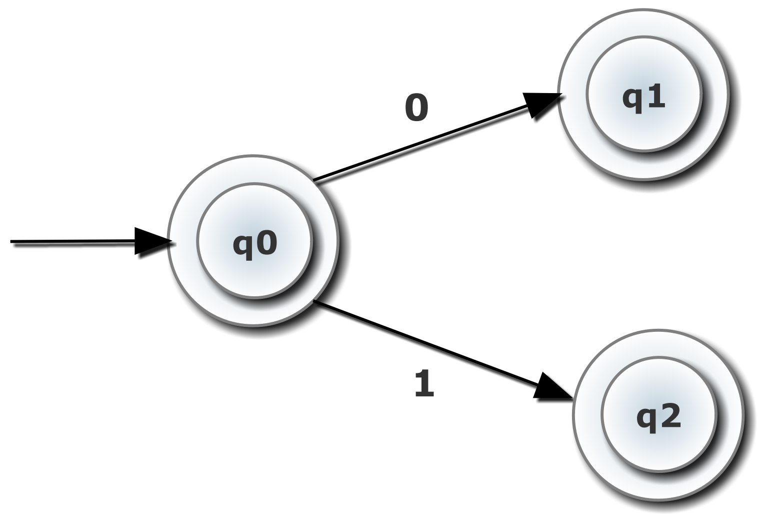




Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language 'L'.

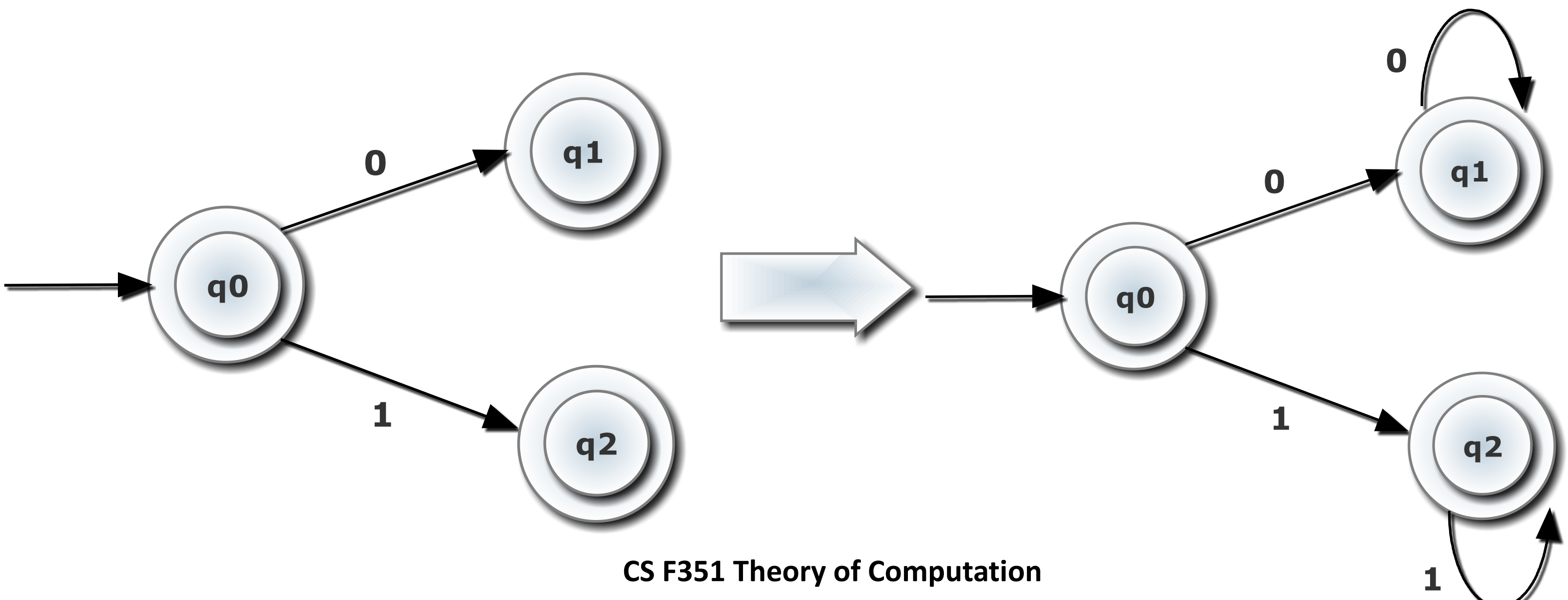
- L = {∈, 0, 1, 00, 11, 010, 101, 01010,----}



Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language 'L'.

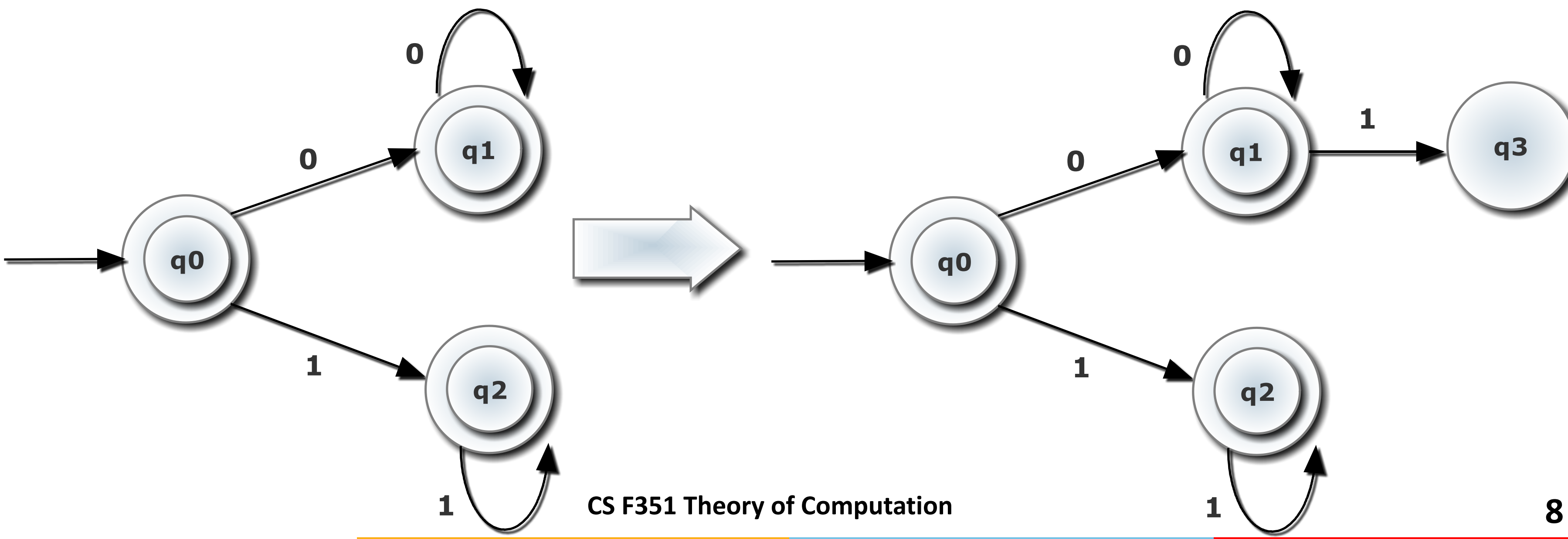
- L = {∈, 0, 1, 00, 11, 010, 101, 01010,----}



Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language 'L'.

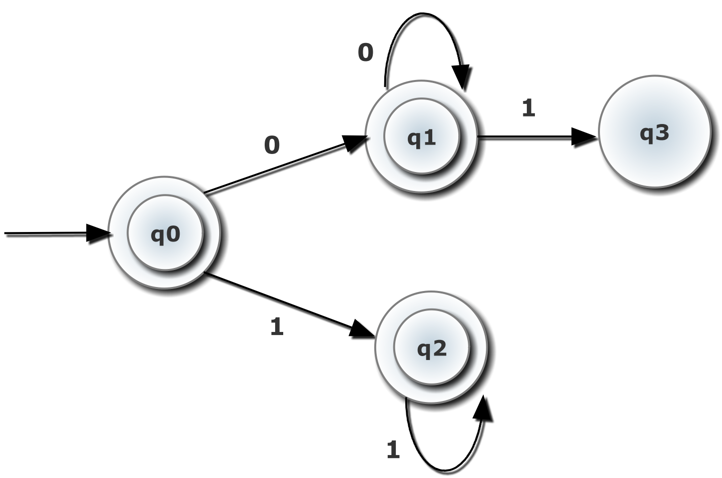
- L = {∈, 0, 1, 00, 11, 010, 101, 01010,----}

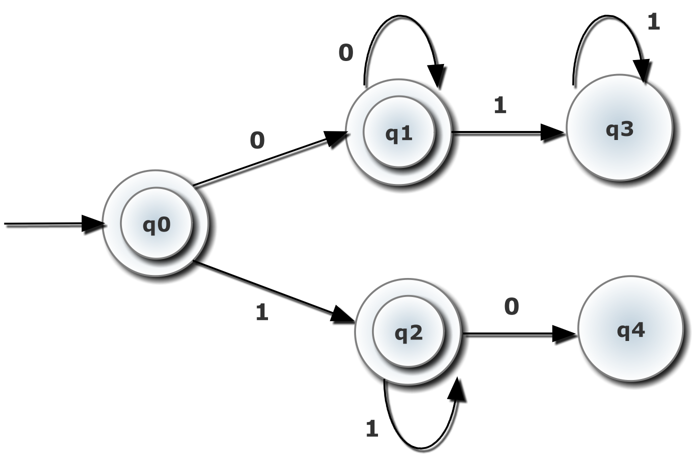


Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

- L = {∈, 0, 1, 00, 11, 010, 101, 01010,----}

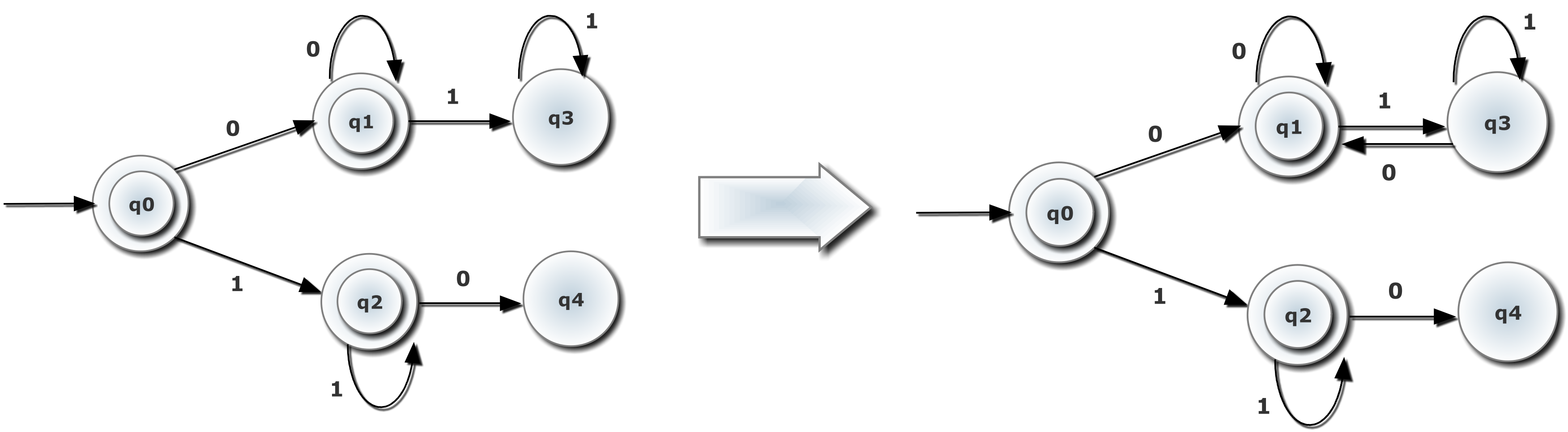




Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

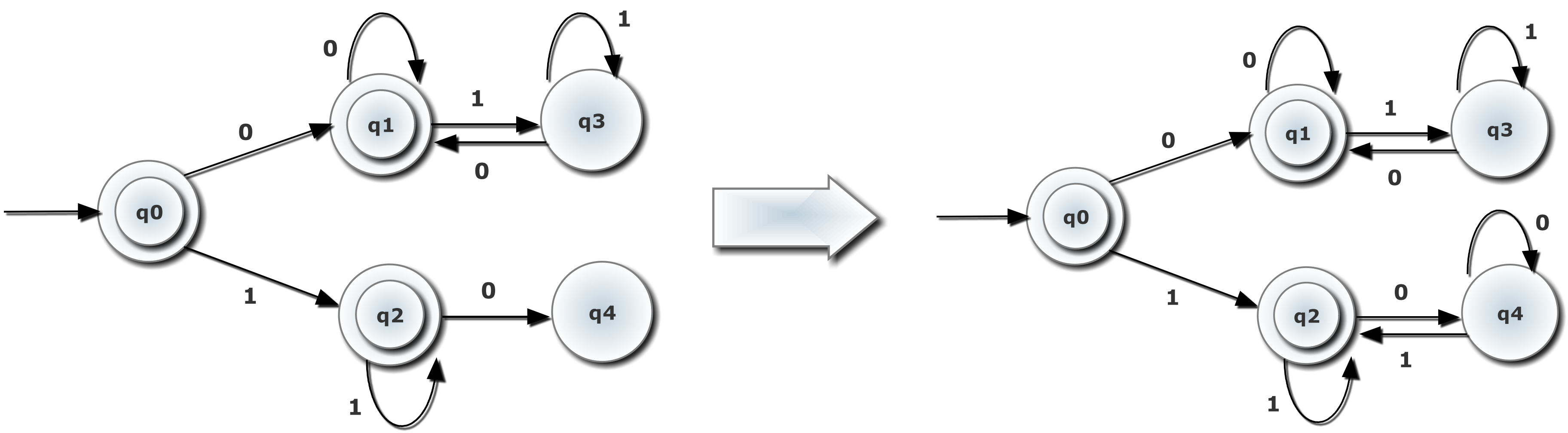
- L = {∊, 0, 1, 00, 11, 010, 101, 01010,----}



Example (Continued…..)

Step 3: Finally, construct a DFA for all the strings specified in the Language ‘L’.

- L = {∈, 0, 1, 00, 11, 010, 101, 01010,----}



Observations regarding DFA

DFA

- For a language, there can be many DFAs.
- Infact, there can be infinitely many DFAs.
- However, every DFA accepts exactly one unique language.

Important Points about DFA

The size of an automaton is independent of the length of the input string given to it. Strings can be arbitrarily long. -

An automaton reads one character at a time.

Deterministic Property of DFA

$$\delta(q,a) \Rightarrow q^{'}$$

- **Here, deterministic property states that given a (state, input alphabet) pair, the computational model will deterministically goes to a unique state.**
- **This means that once a string is fixed, the automaton has a unique walk starting from the start state. If the last state is an accept state then the string is accepted, otherwise not.**

Finite Property of DFA

Set of States is Finite.

- When a FA reads any input string, FA remembers only finite amount of information.

FA has only information about the current state and its future actions will be taken based on the current state.

- When FA is reading any input string and at any point of computation, FA will be at one of the state and from that point onwards, whatever happens to the remaining part of the string is completely independent to whatever FA has read earlier.
- There is no way to store the information for FA.

Formal Computation of DFA

Let $M = \{Q, \sum, \delta, q0, F\}$and let $w = a_1a_2a_3$---$a_n$where $a_i \in w$

We can say that M accepts w if $\exists$ exists a sequence of states r_0, r_1, r_2,----,r_n s.t.

- $r_0 = q0$ (Initial Condition)
- $r_i = \delta(r_{i-1}, a_i)\ \forall\ i = 1,2,$---n (Transition Condition)
- $r_n = F$ (Acceptance Condition)

Acceptance Mechanism of DFA

The DFA M accepts a language $L \subseteq \Sigma^*$

- If every string in L is accepted by M and no more.

This is also referred as L(M).

- The set of strings in the language L accepted by M.
- The language accepted by M, (called L(M)) is defined as
- $L(M) = \{x \,\varepsilon\, \Sigma^* \mid \delta(q_0, x) \,\varepsilon\, F\}$

Language Acceptance Mechanism of DFA

A language L1 is called regular if there is a DFA 'M' s.t. L(M) = L1

- Regular languages are accepted by DFA.
- Also known as Type 3 languages.

However, not every language is regular.

- $L = a^n b^n c^n \mid n \geq 1$ is not regular language.
- Hence, no DFA exists for this language L.

Complement of Language

Suppose L $\subseteq \Sigma^*$ then Complement of L is as follows:

$$\bar{L} = \{x \in \sum^* \mid x \notin L\} = \sum^* - L$$

Then a DFA that accepts the complement of L, i.e. Σ^* - L, can be obtained by swapping its accepting states with its non-accepting states,

- $M_c = < Q, \Sigma, q_0, \delta, Q - F >$ is a DFA that accepts Σ^* - L.
- Also known as Complemented FA.

Home work Assignment

Construct a DFA which accepts all strings of 0's and 1's where each string does not start with symbol '0' or does not end with the symbol '1'.

Example

Construct a DFA which accepts all strings of 0’s and 1’s where the length of each string is not exactly divisible by 3

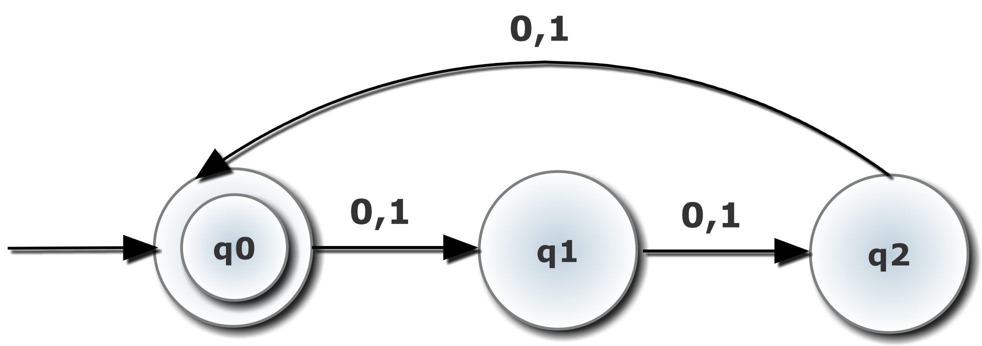
DFA = ?

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

Example (Continued…..)

DFA for all strings of 0's and 1's where the length of each string is exactly divisible by 3

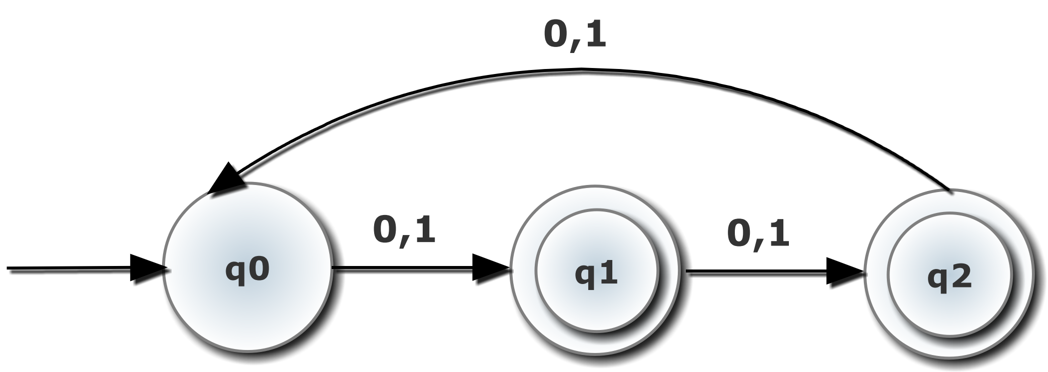
- L = {∈,000, 111, 010110, 101111,---- }



Example (Continued…..)

DFA for all strings of 0's and 1's where the length of each string is not exactly divisible by 3

- L′ = {0,1, 00, 11, 1101, 00110,---}



Complemented DFA