CSE211 – Formal Languages and Automata Theory

U1L8 – Constructing DFA

Dr. P. Saravanan
School of Computing
SASTRA Deemed University

Agenda

- Recap of previous class
  - Types of Automata
  - Pictorial representation of Automaton
  - Introduction to Finite Automata
  - Deterministic Finite Automata(DFA)
  - Definition of DFA
- Notations of DFA
- Extended transition function of DFA
- Designing DFA-Examples

Deterministic Finite Automata

- A deterministic finite automaton (DFA) is a 5-tuple notation:

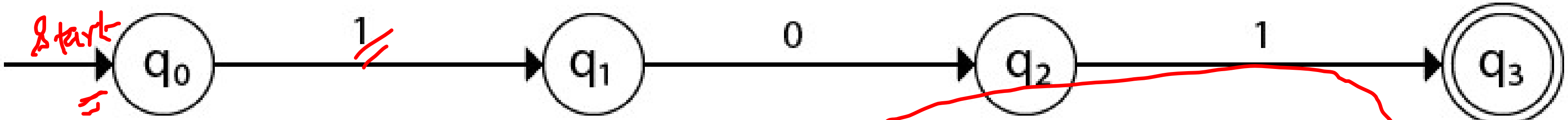
  A= $(Q, \Sigma, \delta, q_0, F)$ where

  - Q is a finite set of states
  - Σ is an alphabet
  - $\delta: Q \times \Sigma \rightarrow Q$ is a transition function
  - $q_0 \in Q$ is the initial state
  - $F \subseteq Q$ is a set of accepting states (or final states).
- In transition diagrams, the accepting states will be denoted by double loops

Designing DFA Example-1

- Design a FA with ∑ = {0, 1} accepts the only input 101

Solution:



$(q_0, 101) \rightarrow q_3$

Q, Σ

$\delta(q_0, 1) \rightarrow q_1$

$\delta(q_1, 0) \rightarrow q_2$

$\delta(q_2, 1) \rightarrow q_3 \in F$

transit

$\Rightarrow (q_1, 01)$

$\Rightarrow (q_2, 1) \Rightarrow (q_2, 1) \Rightarrow q_3 \in F$

Dr.PS

How a DFA processes strings

- Given an input string $x = a_1 a_2 \ldots a_n$, if

  $\delta(q_0, a_1) = q_1$,

  $\delta(q_1, a_2) = q_2$,

  $\ldots$,

  $\delta(q_{i-1}, a_i) = q_i$,

  $\ldots$,

  $\delta(q_{n-1}, a_n) = q_n$, and $q_n \in F$,

  then x is "accepted"; otherwise, "rejected."

- Every transition is deterministic.

101

$\delta(q_0, 101)$

$\delta(q_0, 101) = q_1$

$\delta(q_0, 101) = q_3$

Dr.PS

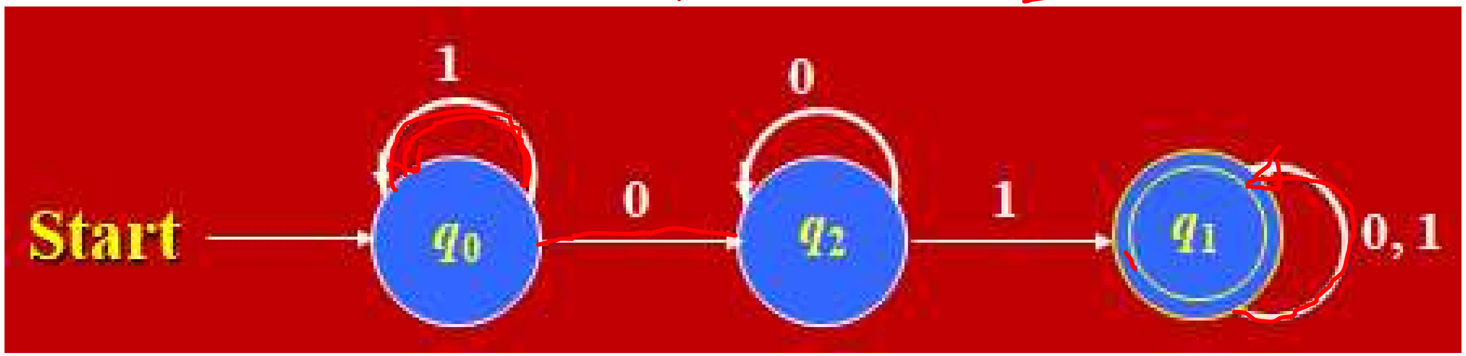
Designing DFA: Example 2

- Design an FA A to accept the language

$$L = \{x01y \mid x \text{ and } y \text{ are any strings of 0's and 1's}\}.$$

- Examples of strings in L={01, 11010, 100011...}

- Transition diagram



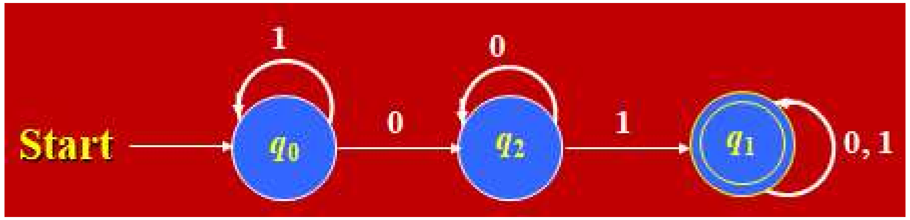
- Why is it deterministic?

$\delta(q_0, 1) = q_0$
$\delta(q_0, 0) = q_2$

Dr.PS

Simpler Notations for DFA's

- Transition diagram
- `



- Transition Table

$10 \to L = \{x01y\}$ states

input →

	0	1
$\rightarrow q_0$	q_2	q_0
$*q_1$	q_1	q_1
q_2	q_2	q_1

$\delta(q_0, 1) = q_0$

$\delta(q_0, 0) = q_2$

~~$\delta(q_2,$~~

$q_2 \in F$

101

$\delta(q_0, 0) = q_2$

$\delta(q_2, 1) = q_1$

~~$\delta(q_1,$~~

→: initial state; *: final state

Dr.PS

Extending transition function to strings

- Given an input string $x = a_1a_2 \ldots a_n$, if

$\delta(p, a_1) = q_1$, $\delta(q_1, a_2) = q_2$,

…,

$\delta(q_{i-1}, a_i) = q_i$,

…,

$\delta(q_{n-1}, a_n) = q_n$, and $q_n \in F$,

then *the extended transition* $\hat{\delta}$ *is defined as*

$$\hat{\delta}(p, x) = q$$

Dr.PS

Extended transition function for Strings

$w = 101$

- Also may be defined recursively
- Recursive definition for $\hat{\delta}$

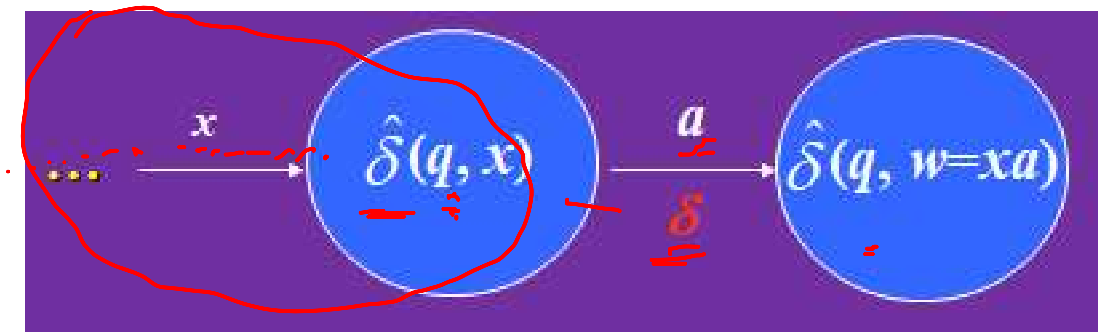
x: 10, a: 1

Basis: $\hat{\delta}(q, \varepsilon) = q$.

Induction: if $w = xa$ (a is the last symbol of w), then

$$\hat{\delta}(q, w) = \delta(\hat{\delta}(q, x), a)$$

The graphical representation is



The Language of a DFA

- The language of a DFA A is defined as

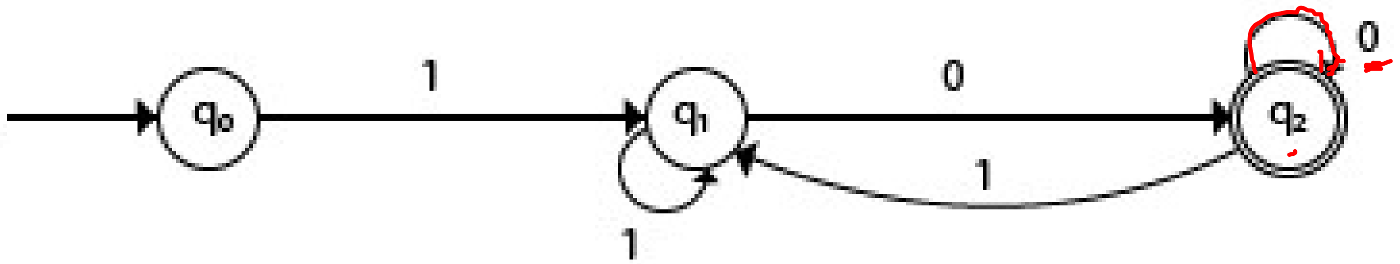
$$L(A) = \{w \mid \hat{\delta}(q_0, w) \text{ is in } F\}.$$

- If L is $L(A)$ for some DFA A, then we say L is a regular language

Designing DFA Example-3

- Design a FA with ∑ = {0, 1} accepts those string which starts with 1 and ends with 0.

110110 10



Dr.PS

Designing DFA Example-4

- Design FA with ∑ = {0, 1} accepts even number of 0's and even number of 1's

Summary

- Types of Automata
- Pictorial representation of Automaton
- Introduction to Finite Automata
- Deterministic Finite Automata(DFA)
- Definition of DFA
- Notations of DFA
- Extended transition function of DFA
- Designing DFA-Examples

References

- John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, *Introduction to Automata Theory,* Languages, and Computation, Pearson, 3rd Edition, 2011.
- Peter Linz, An Introduction to Formal Languages and Automata, Jones and Bartle Learning International, United Kingdom, 6th Edition, 2016.

Next Class:

Non-deterministic Finite Automata(NFA)

THANK YOU.