Afternoon

手机关机或静音放包/兜里

Deterministic Finite Automata

- ◆ Definition
- ◆ Notation
- ◆ Construction
- ◆ Regular Language

Formal Definition

Deterministic finite automaton is a five-tuple ,

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

Where Q is a finite set of *states* ,

Σ is a finite set of *input symbols* ,

q_0 is a *start state* ,

F is a *set of final state* ,

δ is *transition function* , which is a mapping
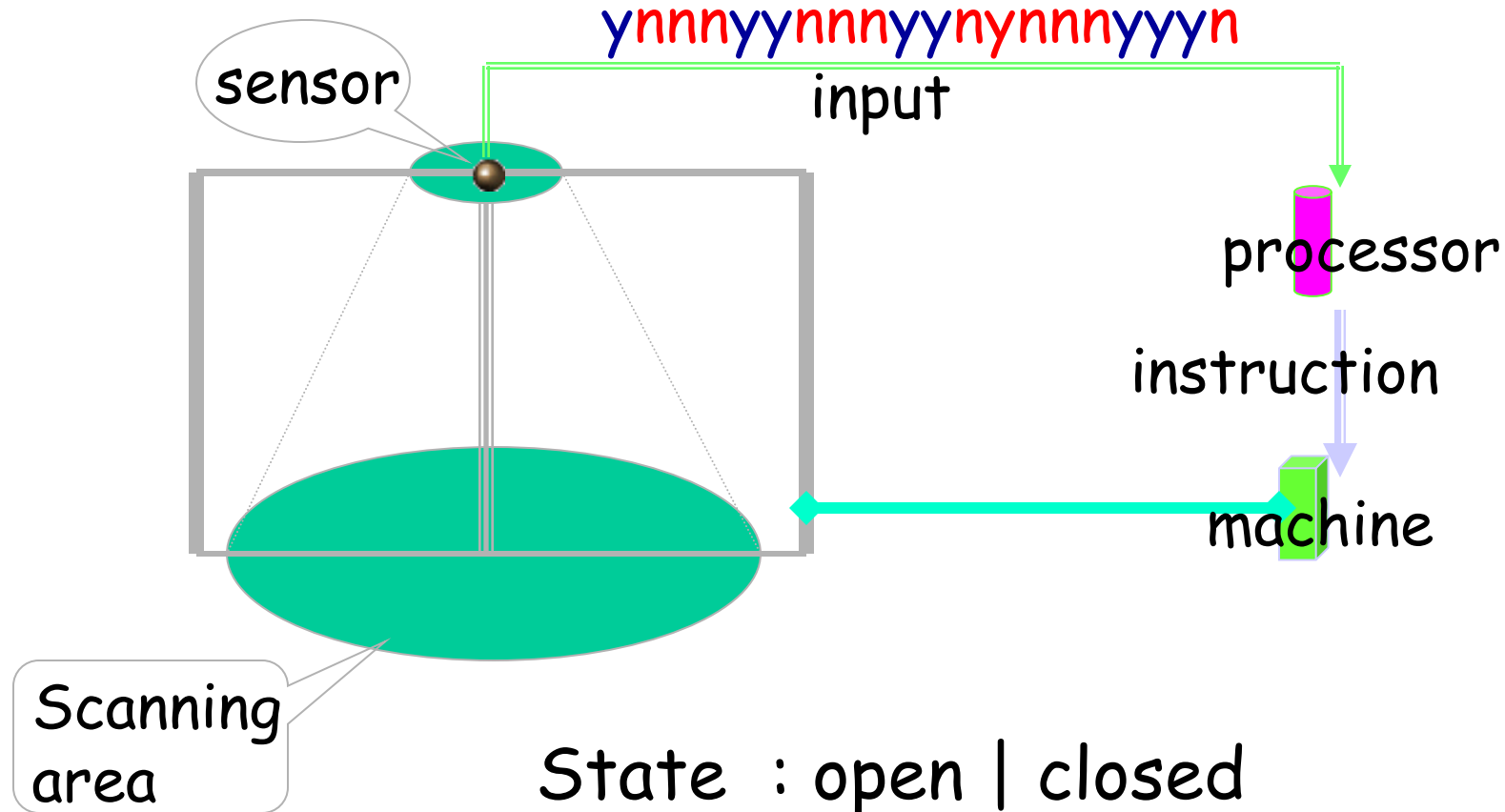
from $Q \times \Sigma$ to Q .

Auto-Door

Auto-Door

Auto-Door

ynnnyynnnyynynnnyyyn

sensor

input

processor

instruction

machine

Scanning
area

State : open | closed

Action : open | close

input   : y | n

Auto-Door

Input symbols : { 0 , 1 } State : { Closed , Open }

State transition :

$$(\text{Closed}, \ 0 \ ) \Rightarrow \text{Closed}$$
$$(\text{Closed}, \ 1 \ ) \Rightarrow \text{Open}$$
$$(\text{Open} , \ 1 \ ) \Rightarrow \text{Open}$$
$$(\text{Open} , \ 0 \ ) \Rightarrow \text{Closed}$$

Start state : Closed

Final state : Closed

Auto-Door

Input symbols : { 0 , 1 } State : { q , p }

State transition function:

$$\delta (q, 0) = q$$
$$\delta (q, 1) = p$$
$$\delta (p , 1) = p$$
$$\delta (p , 0) = q$$

Start state : q

Final state : q

Automaton

DFA for Auto-Door

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

$Q = \{closed, open\}, \quad \Sigma = \{n, y\}$

$q_0 = closed, \quad F = \{closed\}$

$\delta:$

$\delta(closed, n) = closed$

$\delta(closed, y) = open$

$\delta(open, n) = closed$

$\delta(open, y) = open$

DFA for Auto-Door

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

$Q = \{p, q\}, \quad \Sigma = \{0, 1\}$

$q_0 = q, \quad F = \{q\}$

δ:

$\delta(q, 0) = q$

$\delta(q, 1) = p$

$\delta(p, 0) = q$

$\delta(p, 1) = p$

Can you see the Door ?
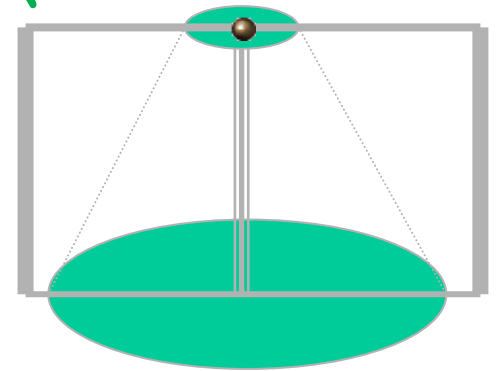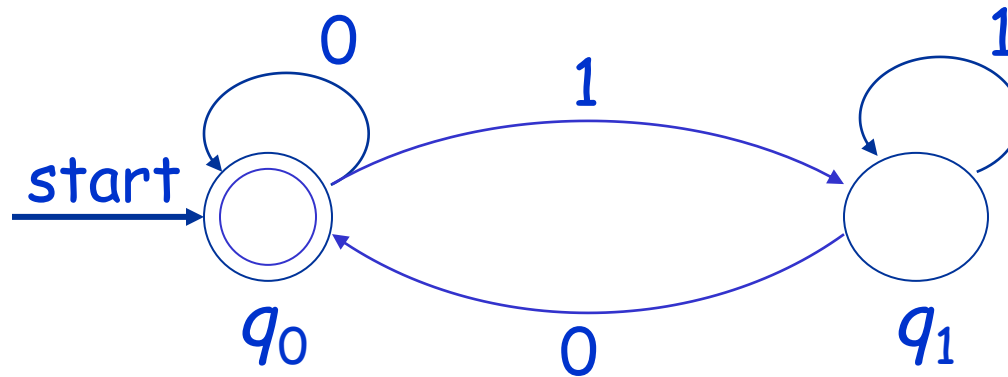
Diagram Notation

$M = (\{q_0, q_1\}, \{ 0,1 \}, \delta , q_0, \{ q_0 \})$

$\delta :$

$$\delta (q_0 , 0) = q_0 , \quad \delta (q_0 , 1) = q_1$$
$$\delta (q_1 , 0) = q_0 , \quad \delta (q_1 , 1) = q_1$$

Table Notation

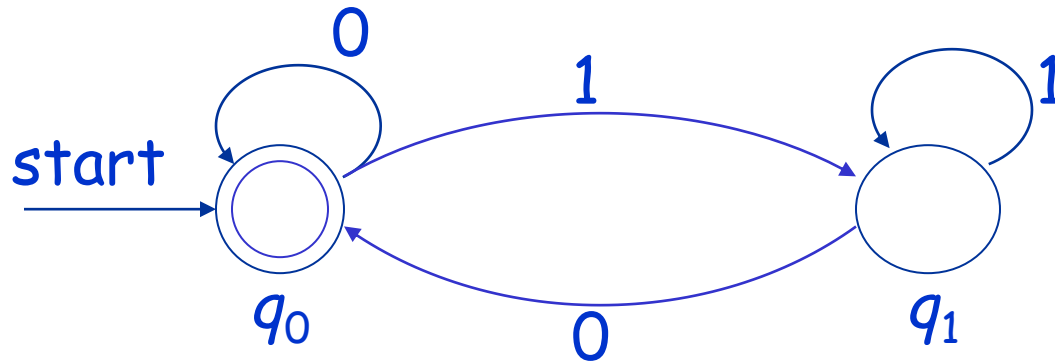|	0	1
$\rightarrow *q_0$	q_0	q_1
q_1	q_0	q_1

$M = (\{q_0, q_1\}, \{ 0,1 \}, \delta, q_0, \{ q_0 \})$

δ :

$$\delta(q_0, 0) = q_0, \quad \delta(q_0, 1) = q_1$$
$$\delta(q_1, 0) = q_0, \quad \delta(q_1, 1) = q_1$$

Partition Strings

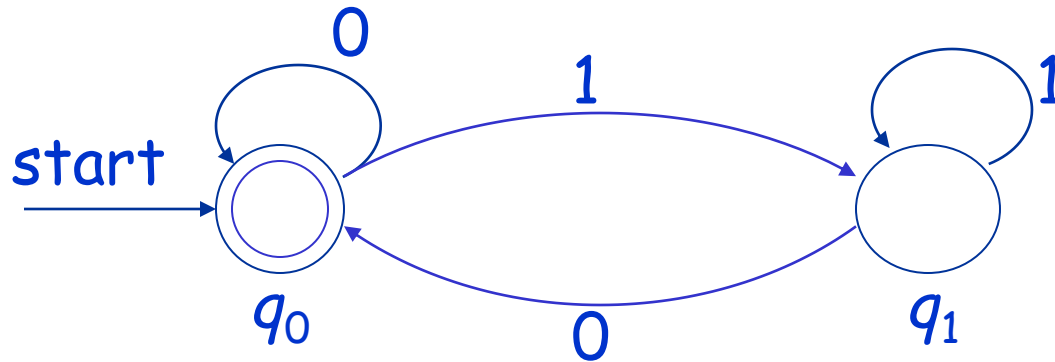

M partitions all strings into two groups:

$L_1 = \{ w \in \{0,1\}^* \mid w \text{ end with } 0 \} \cup \{ \varepsilon \}$

$L_2 = \{ w \in \{0,1\}^* \mid w \text{ end with } 1 \}$

DFA as a recognizer of language

M "recognize" the following language :

$$L = \{\, w \in \{0,1\}^* \mid w \text{ end with } 0 \,\} \cup \{\, \varepsilon \,\}$$

With the language L, and a string $w \in \{0,1\}^*$

M tell us whether w belongs to L, or not

Decision problem

Given a language L, and a string w

Is w belong to L ?

Example 1

$L = \{w \in \{0,1\}^* \mid w$ has both an even number of 0's
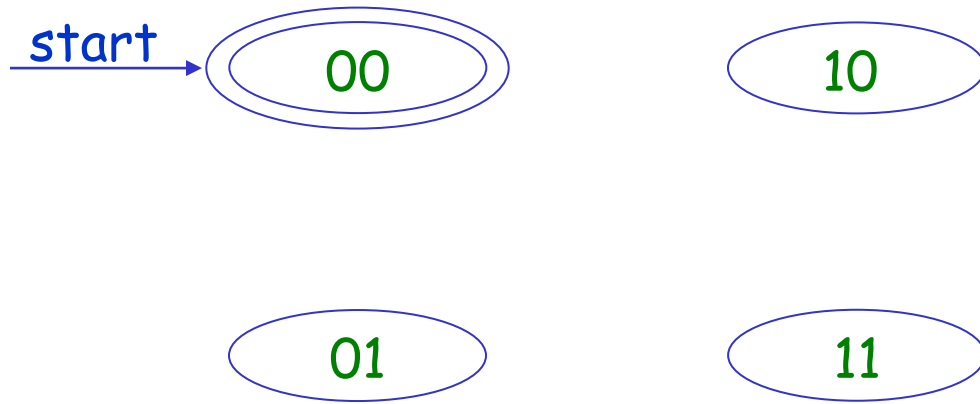
and an even number of 1's $\}$

◆ Partition strings into four groups

- 00 : even 0 and even 1

- 01 : even 0 and odd 1

- 10 : odd 0 and even 1

- 11 : odd 0 and odd 1

Example 1

$L = \{w \in \{0,1\}^* \mid w$ has both an even number of 0's
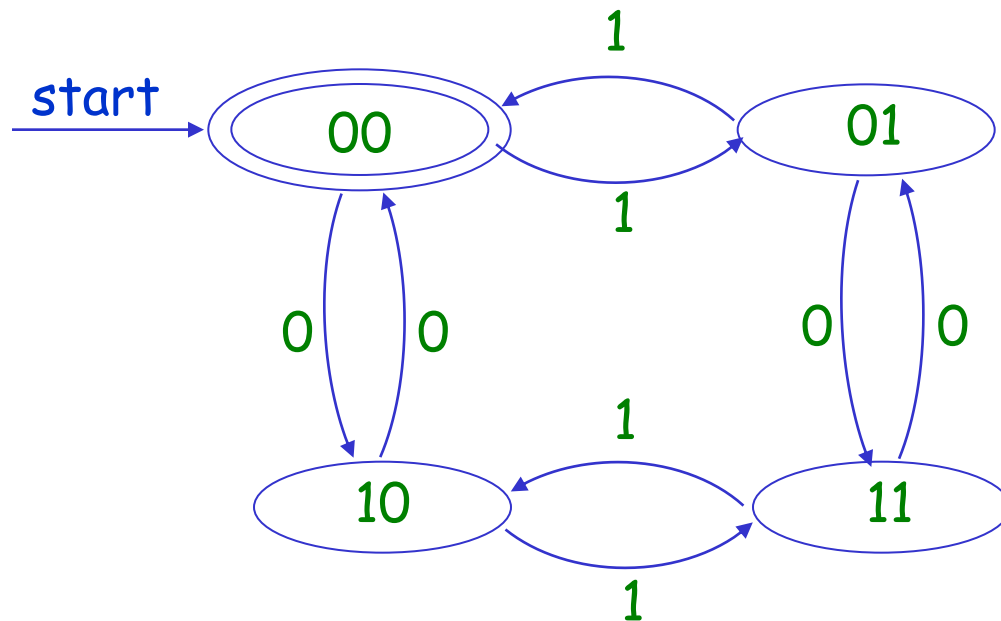
and an even number of 1's $\}$

➢ Set states corresponding to partitions

start → ((00))   (10)

(01)   (11)

Example 1

$L = \{w \mid w$ has both an even number of 0's

and an even number of 1's $\}$

➢ Put transition arcs between states

Example 2

L = {w | w consists of 0's and 1's , and contains sub-string 01}

or   {x01y | x and y are consists of any number of 0's and 1's }

Problem :

How to decide whether a given string w belongs to L ?

Example 2

L = {x01y | x and y are consists of any number

of 0's and 1's }

How to start our work ?

➢ What is the meaning " w belongs to L "

➢ Partition strings by properties of L

➢ Set states which correspond to the partitions

➢ Put transition arcs between states

Example 2

$L = \{x01y \mid x \text{ and } y \text{ are consists of any number}$

$\text{of 0's and 1's}\}$

➢ Partition strings by properties of L

$\{ \varepsilon,0,1,00,01,10,11,000,001,010,100,011,101,110,111 \}$

$\{ 0000,0001,0010,0100,1000,0011,0101,1001,\ldots\ldots \}$

$\delta (q, \; 1 \;) = p$ 1 1 1 0 0 0 1 1 0 0 1
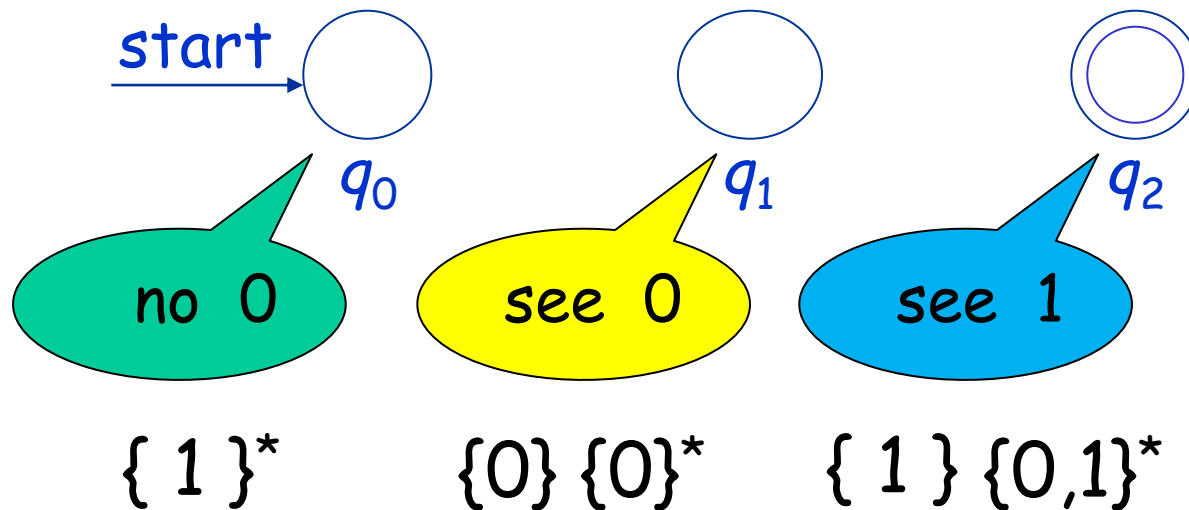
1 1 1 0 0 0 1 1 0 0 1

Example 2

$L = \{x01y \mid x$ and y are consists of any number

of 0's and 1's $\}$

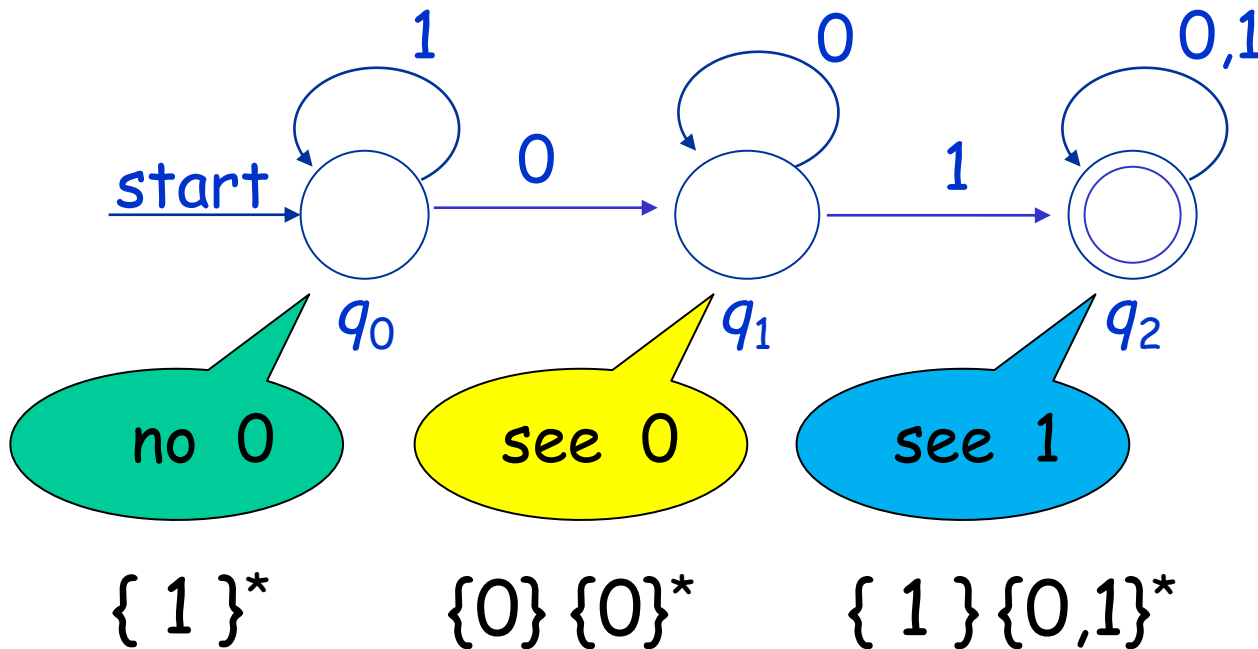➢ Set states which correspond to the partitions

start → ◯ q_0 ◯ q_1 ◎ q_2

no 0   see 0   see 1

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

Example 2

L = {x01y | x and y are consists of any number

of 0's and 1's }

➢ Put transition arcs between states

q_0 no 0 { 1 }*

q_1 see 0 {0} {0}*

q_2 see 1 { 1 } {0,1}*

Extending δ to string

BASIS

$$\hat{\delta}(q, \varepsilon) = q.$$
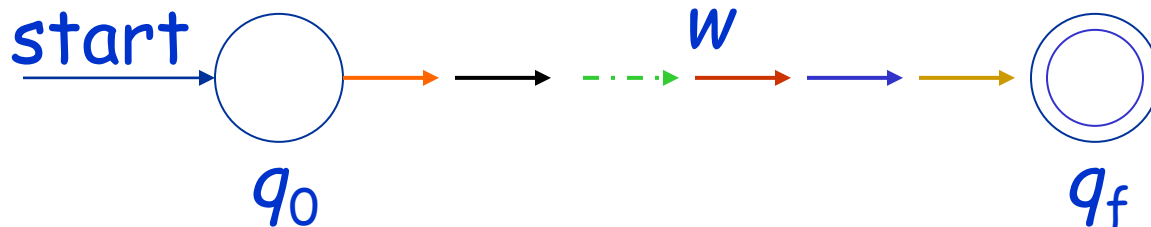
$$\delta(\ q,\ a\ )\ =\ p$$

INDUCTION

Suppose *w* is a string of the form *xa* , that is , *a* is the last symbol of *w*, and *x* is the string consisting of all but the last symbol. Then

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

Language of a DFA

Definition The *language* of a DFA A is denoted $L(A)$, and defined as

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

start → q_0 —→ · · · —→ w —→ q_f

Regular language

Definition

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

　　　　RegL = { *L* | There is a DFA accepting *L* }

Note : a kind of languages accepted by DFA's

Exersizes

Construct DFA for following languages :

    a) $\{ 0 \}^*$

    b) $\{w \mid w \in \{0,1\}^*$ and begin with 0 $\}$

    c) $\{w \mid w$ consists of any number of 0's followed

                        by any number of 1's $\}$

    d) $\{ \varepsilon \}$

    e) ϕ

FIND THE MISSING NUMBER

Good good study
day day up !