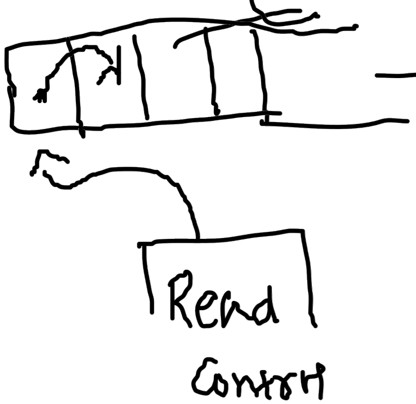
Theory of Computation

Finite State Automaton [FSA]



No Jump

No Write

No left Move

"States" ?

"Finite States"

CS

DS, AI, DL ... — Solvable — ∃ Algo

unsol. — ∄ Algo

"Computing Models"

Cal

↓

Sci

↓

⋮

HPC

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

symbol

$L = \{ x \mid x \in \{0,1\}^* \text{ s.t. } x \text{ ends with '01'} \}$

Ex: 01, 001, 101, 11001, ...

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

Set of states

I/p α-bet

Transition Fn

Start state

Final states

5-tuple M/c

δ : Domain $\qquad$ Co domain

$(Q \times \Sigma) \longrightarrow Q$

$L = \{ x \mid x \text{ ends with } 01 \}$

Subset

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

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

$\{0,1\}^*$ — The set of all strings over $\underline{\{0,1\}}$

$\{ \epsilon, 0, 1, \underline{00, 01, 10, 11}, 000, 0000$

ϵ ↑ Empty string

001 ... 111, ... 1111

① Can we design a FSA to Accept Σ^*

② " " $S \subseteq \Sigma^*$

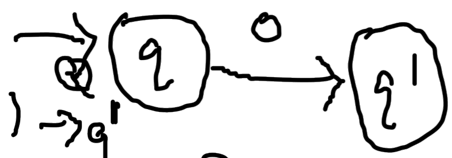
③ This is for you ______

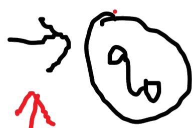
$\boxed{\Sigma^* - \text{Ca}}$

$x \; . \; . \; . \; . \; . \; 01$

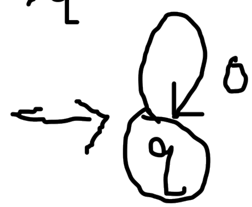
$\delta : Q \times \Sigma \to$

$(q, 0) \to q'$



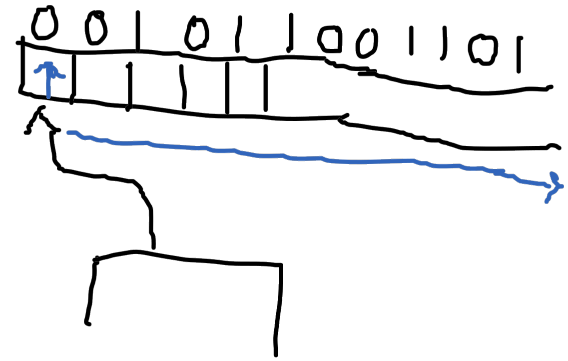


Start State



$Q \times \Sigma \to Q$

$(q, 0) \to q$



q_0 ⋯ F

1101

10111100001

0 - - - - - 01

1 - - - - - 101

q_0 ⋯ NON final

0 - - - - - 0

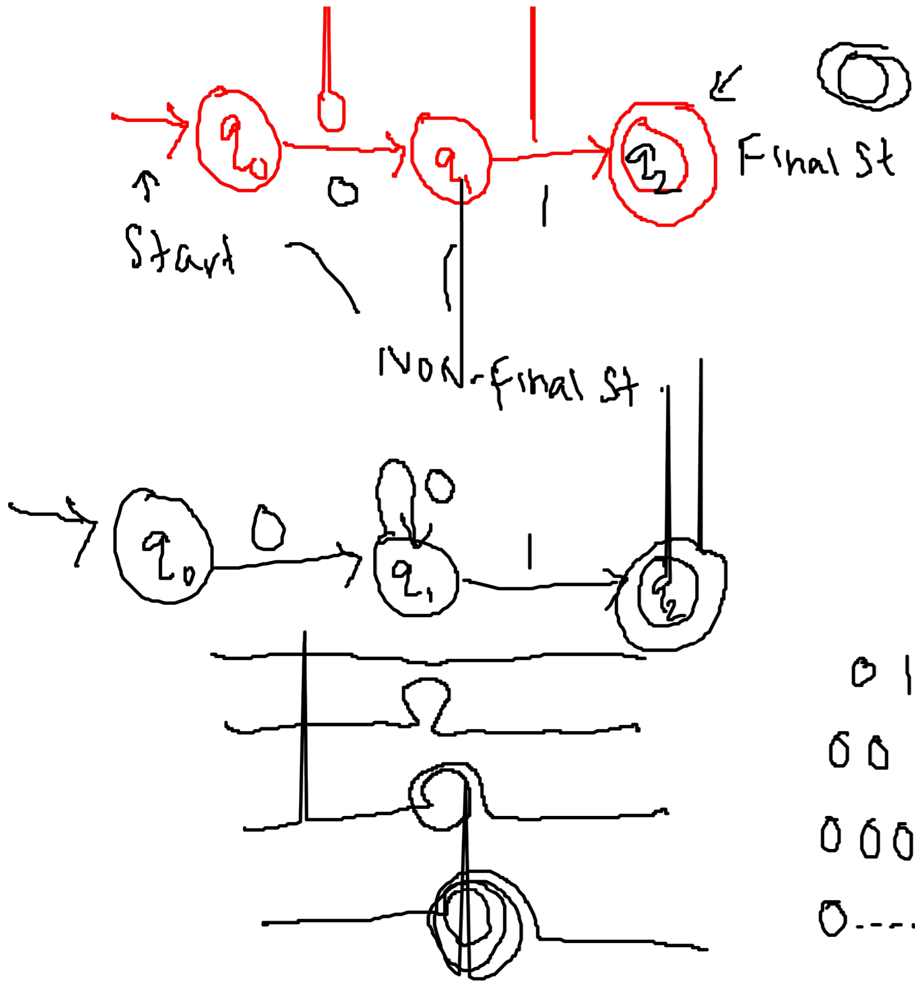
1 - - - - - 1

Step 1: FSA $\underline{M}$ that accepts just 01

2: Modify M to ACC strings of leng 3 ending with 01

3 4 "

4 - - - to reject strings NOT ending with 01



0 1

0 0 1

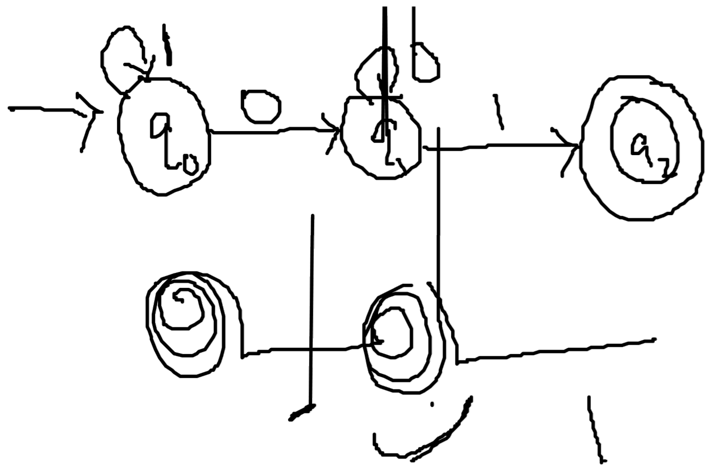
1 0 1

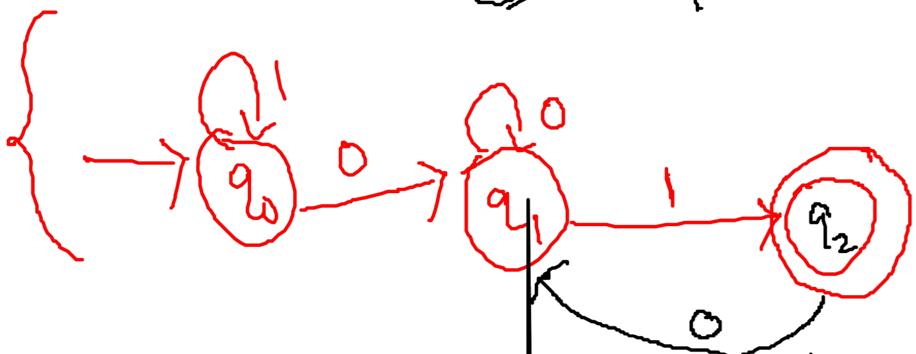
0 1

0 0 1

0 0 0 1

0 - - - - 0 1





FSA – MUST – Accept ALL Valid Strings (Ending with 01)

– Reject ALL Invalid Strings (NOT ending with 01)

01

101

11101

1....101

1....10....01

$\delta : Q \times \Sigma \to Q$

$\{q_0, q_1, q_2\} \times \{0, 1\} \to \{q_0, q_1, q_2\}$

$(q_0, 0) \to q_1$

$(q_0, 1) \to q_0$

$(q_1, 0) \to q_1$

$(q_1, 1) \to q_2$

$(q_2, 0) \to$

$(q_2, 1) \to$

$1\cdots1\ 0\cdots0\ \underline{001101}$

$0\ 1\ 1110\cdots01$

$1\cdots1\ 0\cdots01$

$\delta(q_2, 0) = q_0$

0101 → valid string

$q_0 \xrightarrow{0} q_1 \xrightarrow{1} q_2 \xrightarrow{0} q_0 \xrightarrow{1} q_0$

01001

0101

$\delta(q_2, 0) = q_1$ ✓

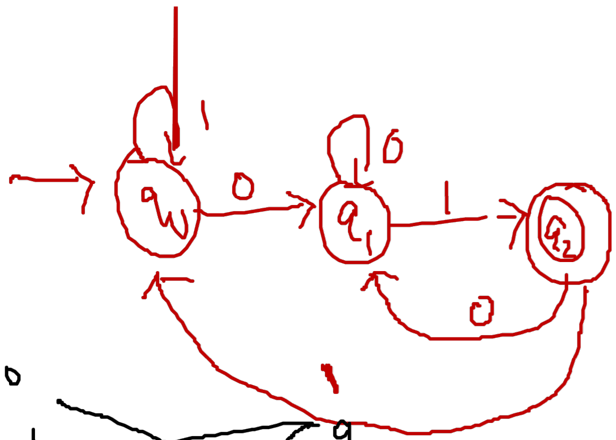
→ q_0

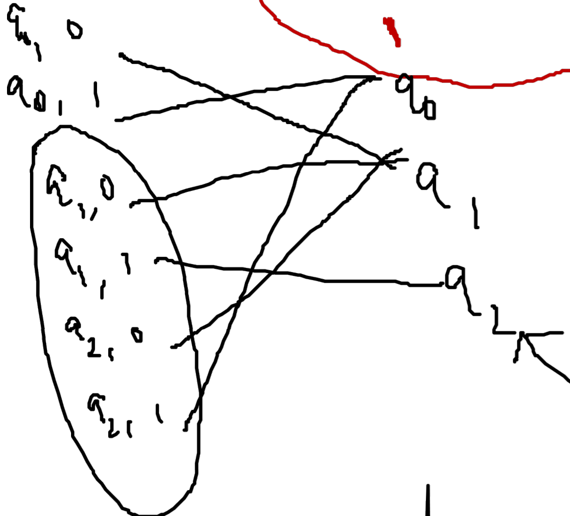
→ q_2

→ q_3

X

= FSA Rejects 0101 (valid st)





$\delta(q_2, 1) =$ ___ q_2 0101 ✓

0 0 ... 0 1

0 0 1

0 1 01

$\delta(q_2, 1)$ — q_1 0111 [Invalid but Accepted FSA]

q_2

0111...11111

FSA $M = \{Q = \{q_0, q_1, q_2\}, \Sigma = \{0, 1\}, \delta, q_0, F = \{q_2\}\}$

011 [Reject but Acc by FSA]

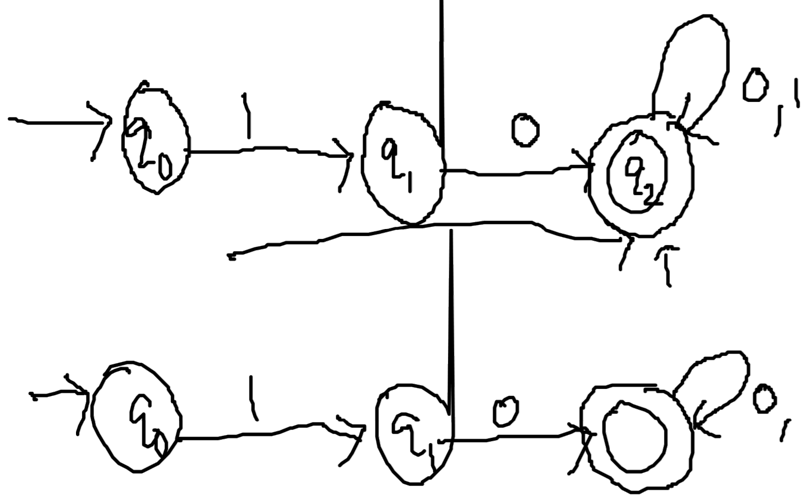
ACC All String ending with 01.

$\Sigma = \{0,1\}$ $L = \{x \mid x \text{ begins with } 10 \}.$

$10, \ 10\underline{1}, \ 10\underline{0}, \ 10\,\underline{0}\,\underline{0}, \ \ldots, \ 10$ Any string over $\{0,1\}$

Suffix

0	0
0	1
1	0
1	1

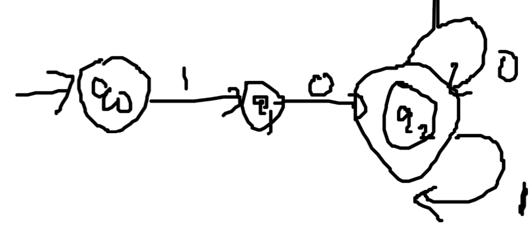


$10\underline{0}$

$10\underline{1}$

$\delta(q_2, 0) =$

$\delta(q_2, 1) =$



$q_0 \to q_1 \to q_2$

0, 1, 00, 01, 10, 11, 000, 0011, 0...0, 1...1, 010101, 1010...

Any suffix over {0,1}.

$M = \{ Q = \{q_0, q_1, q_2, q_3\},\ \Sigma = \{0,1\},\ \delta$

Invalid

All strings {0,1} beginning 0

beg with 11

$Q \times \Sigma \to Q$

$(q_0, 0) \to q_3$

$(q_0, 1) \to q_1$

$(q_1, 0) \to q_2$

$(q_1, 1) \to q_3$

$(q_2, 0) \to q_2$

$(q_2, 1) \to q_2$

$(q_3, 0) \to q_3$

$(q_3, 1) \to q_3$

00 011110 110

1101

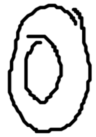
00

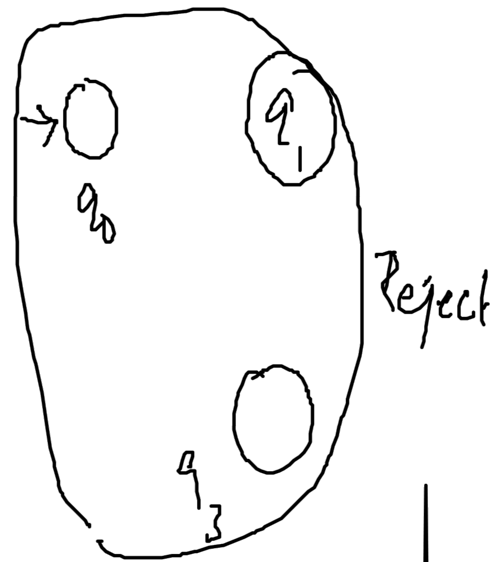
0

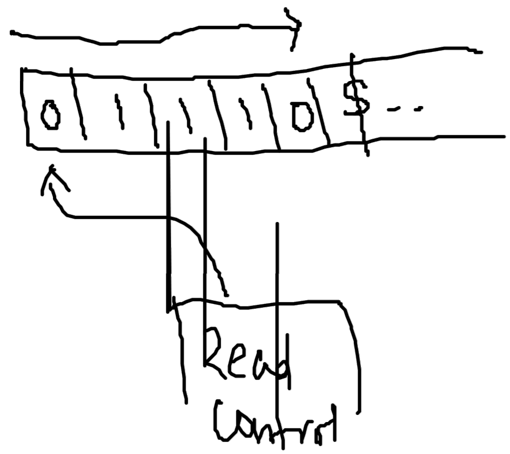
10 Accept

Rej

1 1

Accept 

Reject 



When I/p Tape is Exhausted
[I/p is Read Completely by FSA]

→ Control is at q_2 - Accept

→ Control is at $q_0 | q_1 | q_3$ - Rej

$L = \{ x \mid x \text{ begins } 10 \}$ — $\underline{4}$-State FSA [IS this a MIN State FSA]

Does $\exists$ 3-State FSA

1-State FSA

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

$Q \times \Sigma \to Q$

$\delta(q_0, 0)$

$\delta(q_0, 1)$

$M = \{ Q, \Sigma, \delta, q_0, F \}$

$\{q_0\}$ $\{0,1\}$ δ $q_0 \in Q$ $\subseteq Q$

"Modelling the Comp"

" Program | Algorithm | Procedure"

$\{q_0\} \times \{0,1\} \to \{q_0\}$

$\boxed{\text{Unique}}$ Start state

$F = \emptyset$ or $\{q_0\}$

~~No~~ $F \subseteq Q$

$q \in Q$ $\quad$ q : unique

$\in$ → belongs to | an element of Q

$F \subseteq Q$ $\quad$ F : Can contain multiple states

→ ϕ Empty set [state]

$Q = \{q_0, q_1, q_2\}$

→ $\{q_0\}$ $\quad$ $\{q_1, q_2\}$

→ $\{q_1\}$

→ $\{q_2\}$

$\{q_0, q_1\}$ $\quad$ $\{q_0, q_1, q_2\}$.

$Q = \{q_0, q_1, q_2\}$

L_1 $\quad$ $F = \phi$

L_2 $\quad$ $F = \{q_0\}$

L_3 $\quad$ $\{q_1\}$

L_4 $\quad$ $\{q_2\}$

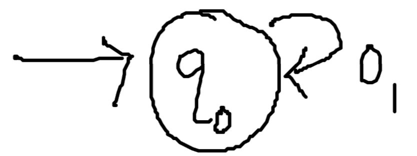
$\vdots$

L

$2^3 = 8$ different languages.

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

$Q = \{q_0\}$



$F = \emptyset$ ← Empty State

$L = \emptyset$

↳ FSA Accepts Nothing.

$Q \times \Sigma \to Q$

$F \subseteq Q$

$L \subseteq \{\epsilon, 0, 1, 00, \ldots\}$

↑ Empty String

[Strings of length 0]

Accepts Everything

$\{\epsilon, 0, 1, 00, \ldots\}$

? $L \in \mathcal{P}(\Sigma^*)$