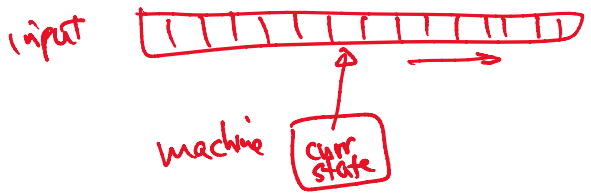
January 25, 2022 10:36 AM

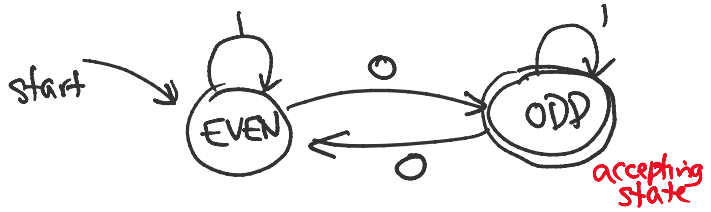
Deterministic Finite Automata (DFA)

intuitively, machine/program that uses const amount of memory & reads input in pass (from left to right)

Q ← const # of "states"



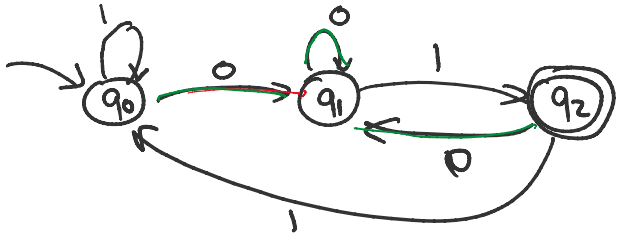
Ex0 all strings over {0,1} with odd # of 0's



$Q = \{q_0, q_1, q_2\}$
$\Sigma = \{0,1\}$
$s = q_0$
$A = \{q_2\}$

q	δ(q,0)	δ(q,1)
q_0	q_1	q_0
q_1	q_1	q_2
q_2	q_1	q_0

Ex1 all strings ending with 01



States:
- q_1 : just seen 0
- q_2 : just seen 01
- q_0 : none of above

program:

```
state = q0
while (not end of input) {
    c = next input symbol;
    if (state == q0 && c == 0)
        state = q1;
    else if (state == q0 && c == 1)
        state = q0;
    ⋮
}
if (state == q2)
    output yes
else  ... no
```

← O(n) time & O(1) space

e.g. input: 00101

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

$$\begin{aligned}\delta^*(q_0, 001) &= \delta^*(\delta(q_0,0), 01) \\ &= \delta^*(q_1, 01) \\ &= \delta^*(\delta(q_1,0), 1) \\ &= \delta^*(q_1, 1) \\ &= \delta^*(\delta(q_1,1), \varepsilon) = \delta^*(q_2, \varepsilon) = q_2\end{aligned}$$

Formal Defn A DFA is specified by 5 things:

Formal Defn A DFA is specified by 5 things:

$M = (Q, \Sigma, s, \delta, A)$ where

Q is a finite set of states

Σ is finite alphabet

$s \in Q$ is the start state

$A \subseteq Q$ is the set of accepting states

$\delta : Q \times \Sigma \to Q$ is the transition fn

(Q: curr state; Σ: curr symbol; Q: next state)

Def Given transition fn δ, define its extended transition fn $\delta^* : Q \times \Sigma^* \to Q$ inductively:

(i) $\delta^*(q, \varepsilon) = q$

(ii) $\delta^*(q, x) = \delta^*(\delta(q,a), y)$ if $x = ay$ with $a \in \Sigma$, $y \in \Sigma^*$

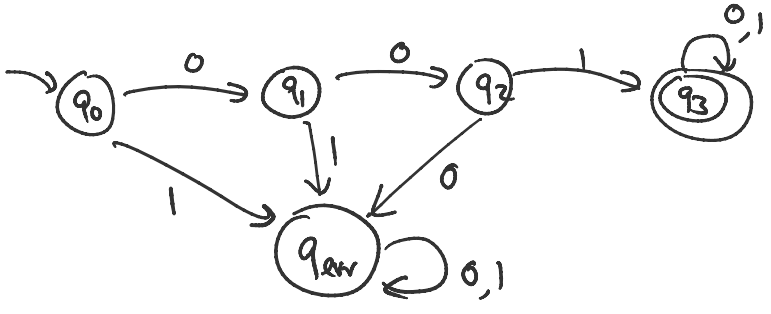
Def M accepts x iff $\delta^*(s, x) \in A$

Define $L(M) = \{ x \in \Sigma^* : M \text{ accepts } x \}$

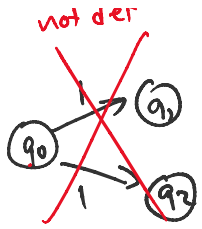
($L(M)$: lang. accepted by M)

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

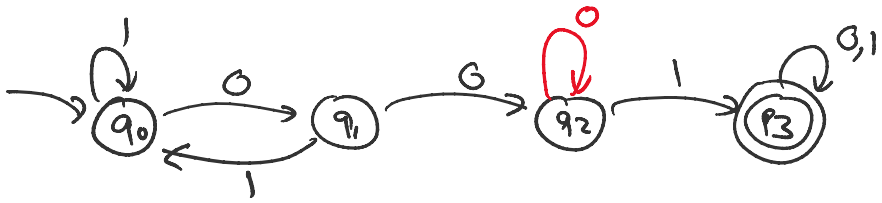
a) all strings beginning with 001



not det



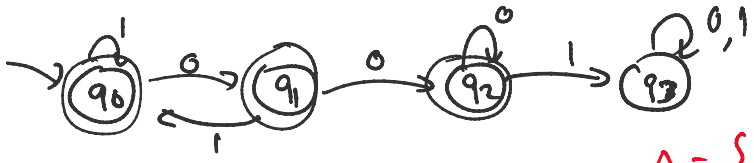
b) all strings containing 001 as a substring



q_3: found 001
q_2: just seen 00 but not found 001 yet
q_1: just seen 0 but not in q_2, q_3
q_0: none of above

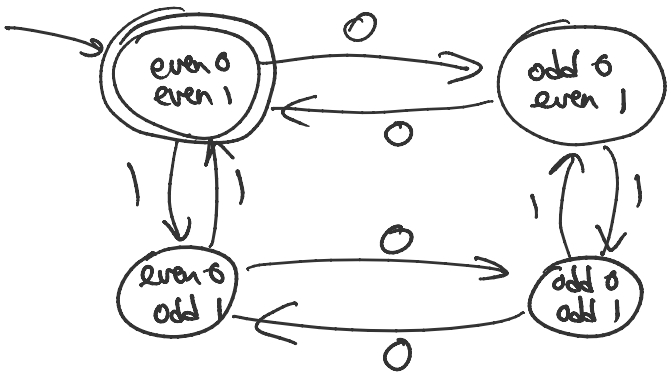
$\Rightarrow$ generalizes to other pattern strings
$\Rightarrow$ pattern matching alg'm with $O(n)$ time

c) all strings not containing 001



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

d) all strings with even # of 0's
and even # of 1's



e) strings with length divisible by 5