Feb 5th, 2019 — CSE 396 Week 2 Lecture 1 Spr19 ①

Def^n: A <u>deterministic finite automaton</u> (DFA) is a 5-tuple $M = (Q, \Sigma, \delta, s, F)$, where

Q is a finite set of <u>states</u>.

Σ is an alphabet — that is, a finite set of <u>chars</u>.

s is a member of Q, the <u>start</u> state (q_0 in text)

F, is a subset of Q, the set of <u>final states</u> (accepting).

δ is the <u>transition function</u>: $\delta : \underline{Q \times \Sigma} \longrightarrow Q$

$$(q, c) \longrightarrow q'$$

```
class DFA {
   set<State> Q;
   set<char> Σ;
   State s;
   set<state> F;
   State delta(state p, char c); }.
```

KWR prefers: ~~det~~ delta

```
set<Triple<State, char, State>> delta;
```

~~Visula~~ <u>Visualization</u>: $\delta \subseteq (Q \times \Sigma) \times Q$ $\quad p, q \in Q,\ c \in \Sigma$

$$((p, c), q)$$

Q is a set of nodes

δ is a set of edges, with <u>labels</u> from Σ.

$$p \xrightarrow{c} q$$

计 start — new state

$x = cc$

$x' = c$

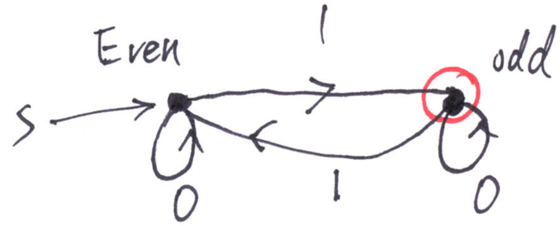
Example: Tell whether a given string $\underline{x}$ over $\Sigma = \{0, 1\}$ has an odd number of 1s. ($\#1(x)$ = number of 1's.)

if $\#1(x)$ ~~=~~ is odd or not??

$Q = \{\text{even}, \text{odd}\}$.

s = even, since we have seen zero of 1's.

$\delta = \{(s, 0, s), (s, 1, \text{odd}), (\text{odd}, 0, \text{odd}), (\text{odd}, 1, \text{even})\}$.



$x = 1011$

$s \xrightarrow[1]{} \text{odd} \xrightarrow[0]{} \text{odd} \xrightarrow[1]{} \text{even} \xrightarrow[1]{} \text{odd}$

M accepts $\underline{\underline{x}}$.

The language $L(M)$ of this DFA M equals $\{x \in \{0,1\}^* : \#1(x) \text{ is odd}\}$

zero or more

Def^n: A computation by a DFA $M = (Q, \Sigma, \delta, s, F)$ is a sequences

$\vec{c} = (q_0, x_1, q_1, x_2, \ldots, x_{n-1}, q_{n-1}, x_n, q_n)$ where:

$n = |x|$ (the length of x)

$x = x_1 \cdots x_n$ where x_i is i-th bit.

$q_0 = s$, each $q_i \in Q$ and

$\boxed{\text{For all } j,\ 1 \le j \le n,\ (q_{j-1}, x_j, q_j) \in \delta}$

$\vec{c}$ is accepting if also $q_n \in F$.

Def^n: $L(M) = \{x \in \Sigma^* : M \text{ has an accepting computation on input } x\}$.

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

$M_0 =$ 

$Q = \{s\}$, $F = \emptyset$

$L(M_0) = \emptyset$

ε empty string $\emptyset$ empty set.

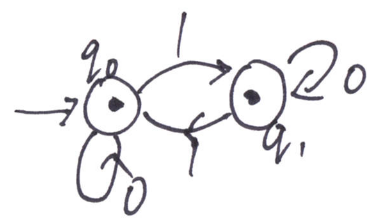
$M_{all} =$ 

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

$F = \{s\}$

$L(M_{all}) = \Sigma^* = \{0, 1\}^*$

other DFAs M s.t. $L(M) = \Sigma^*$:



$Q = \{q_0, q_1\}$

$F = \{q_0, q_1\}$.

Hence a DFA need not be "in lowest terms".

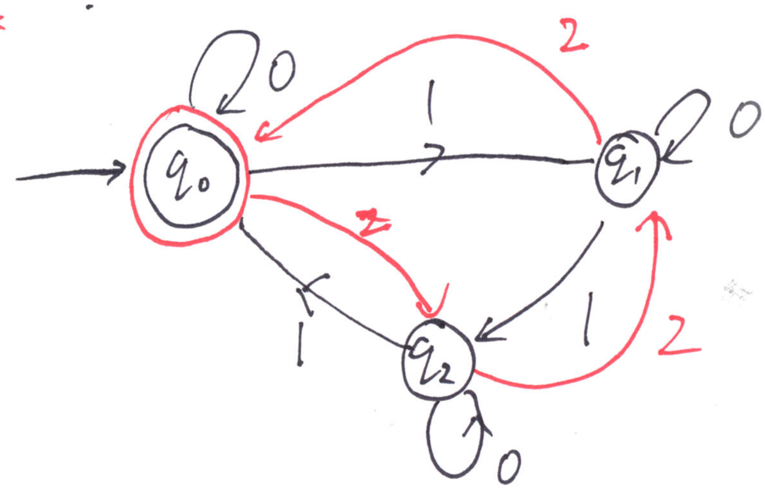
One More Example:

Tell whether a given string X has the property that

$\cancel{\#(1)}$ $\#1(X) \equiv 0$ ~~mod~~ mod 3 ?

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

$\uparrow$ $\equiv 0$ $\uparrow$ $\equiv 1$ $\uparrow$ $\equiv 2$



$s = q_0$

$F = \{q_0\}$.

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

change $\Sigma = \{0, 1, 2\}$. ?? if the sum of digits in X is a multiple of 3

$\equiv 0 \mod 3$

$\equiv 1 \mod 3$.

set operation:

union $A \cup B = \{x : x \in A \text{ or } x \in B\}$

intersection $A \cap B = \{x : x \in A \text{ and } x \in B\}$.

Difference of sets: $A \setminus B = \{x : x \in A \text{ but } x \notin B\}$

(in the Text, '$-$')

symmetric difference: $A \triangle B = \{x : x \in A \text{ XOR } x \in B\}$

$$= (A \setminus B) \cup (B \setminus A).$$