CS 205

Lecture # 3

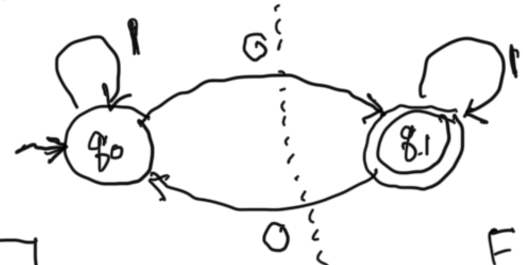
11/01/21

Deterministic Finite Automata : DFA

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

- Q: finite set of states
- Σ: alphabet
- $\delta : Q \times \Sigma \to Q$
- $F \subseteq Q$: final / accepting states

Graphical Notation



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

$F = \{q_1\}$

$\delta(q_0, 0) = q_1$

$\delta(q_0, 1) = q_0$

$\hat{\delta}(q_0, 001) = q_0$

$$\hat{\delta}(q_0, 001) = \delta(\hat{\delta}(q_0, 00), 1)$$

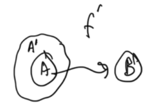
$$= \delta(\delta(\hat{\delta}(q_0, 0), 0), 1)$$

$$= \delta(\delta(\delta(\hat{\delta}(q_0, \epsilon), 0), 0), 1) = q_0$$

Extend transition fn. from symbols to strings

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

extended to



$$\hat{\delta} : Q \times \Sigma^* \to Q$$

Define $\hat{\delta}$ by induction as follows:

1. $\hat{\delta}(q, \epsilon) = q$ for all $q \in Q$
2. $\hat{\delta}(q, ua) = \delta(\hat{\delta}(q, u), a)$ for all $q \in Q$, $u \in \Sigma^*$, $a \in \Sigma$

Convention: Write δ instead of $\hat{\delta}$

Def (1) The DFA $M = (Q, \Sigma, \delta, q_0, F)$ accepts the string $\omega \in \Sigma^*$ iff $\hat{\delta}(q_0, \omega) \in F$.

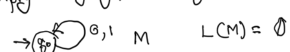
(2) The language accepted by M denoted $L(M) = \{\omega \in \Sigma^* \mid \hat{\delta}(q_0, \omega) \in F\}$

---

(3) A language $L \subseteq \Sigma^*$ is called regular if it is accepted by a DFA M, i.e., $L = L(M)$ for some DFA M.

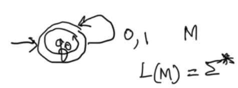
Examples ($\Sigma = \{0,1\}$)

(1) $\emptyset$ (the empty language) is regular



(2) Σ^* is regular

$\hat{\delta}(q_0, x) = q_0 \in F$ for all $x \in \Sigma^*$



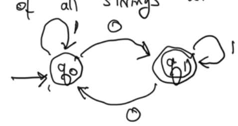
(3) $\{\epsilon\}$ is regular

$\hat{\delta}(q_0, \epsilon) = q_0 \in F$
$\hat{\delta}(q_0, x) = q_1 \notin F$ for all $x \neq \epsilon$.

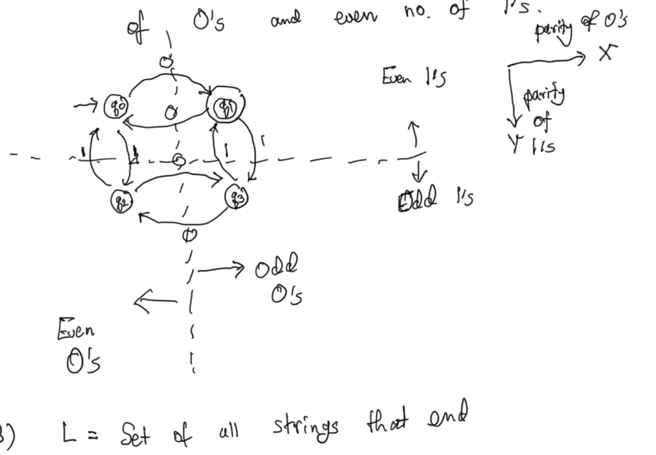
"δ is a function."

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

(1) L = Set of all strings with an odd no. of 0's.



(2) L = Set of all strings with an odd no. of 0's and even no. of 1's.



(3) L = Set of all strings that end with a 1.