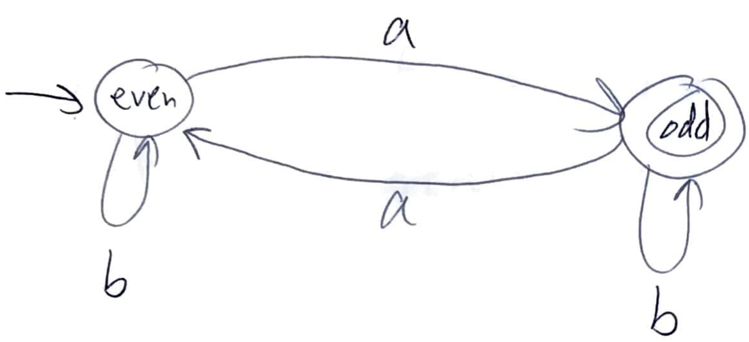
CSCE 355
1/12/2022

Recall:



transition diagram

abbab

Def: A deterministic finite automaton (DFA) is a 5-tuple $\langle Q, \Sigma, \delta, q_0, F \rangle$ where

- Q is a finite set (state set; elements called states)
- Σ is an alphabet
- $q_0 \in Q$ (the start state)
- $F \subseteq Q$ (the set of accepting states) (elements in $Q \setminus F$ are called rejecting states)

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

$\Sigma = \{a, b\}$

$q_0 = \text{even}$

$F = \{\text{odd}\}$

② $\delta : Q \times \Sigma \to Q$ (transition function)

(δ is a function that takes a state and an alphabet symbol and maps it to a state)

$q \xrightarrow{a} r$

means

$\delta(q, a) = r$

	a	b
(start state) → even	odd	even
* odd	even	odd

tabular form

e.g., $\delta(\text{even}, a) = \text{odd}$

→ means start state

* means accepting state

Given a DFA A and a string w over A's alphabet, A accepts w if the final state of A after reading w is an accepting state ($\in F$) Otherwise, say that A rejects w.

$\in$ — membership ("is an element of")

$x \in F$

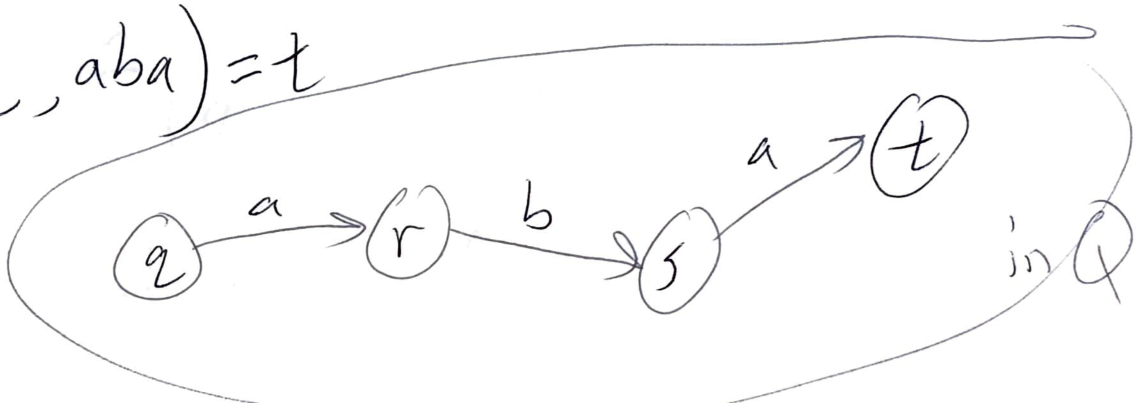
$F \subseteq Q$ F is a subset of Q

(3) Extended transition function

$A = \langle Q, \Sigma, \delta, q_0, F \rangle$ a DFA.

We define the extended transition function $\hat{\delta} : Q \times \Sigma^* \to Q$ so that

$\hat{\delta}(q, aba) = t$



Ex: $\hat{\delta}(\text{odd}, aaba) = \text{even}$

Inductive def of $\hat{\delta}$:

Base case 1) $\hat{\delta}(q, \varepsilon) = q$

Inductive case 2) For any string $w \neq \varepsilon$, can write $w = xa$ for some unique $x \in \Sigma^*$ and $a \in \Sigma$ (a is the last symbol of w, x is the principal prefix of w

4 ~~Assume $\hat{\delta}$ has been defi~~

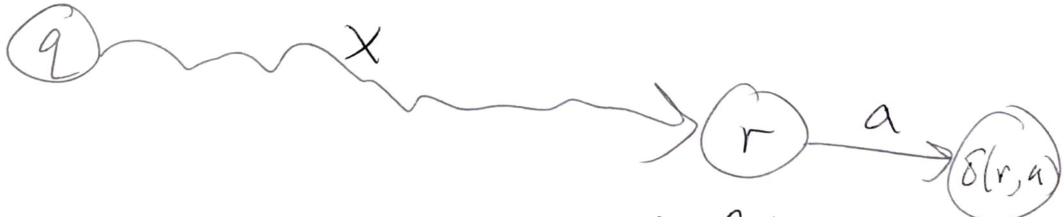
(and notice $|x| = |w| - 1$)

Assume $\hat{\delta}(s, x)$ has been defined for ~~any~~ all $s \in Q$.

Then define, for all $q \in Q$,

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

where $r = \underline{\hat{\delta}(q, x)}$



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

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

<u>Ex</u>:

$$\hat{\delta}(\text{even}, \underline{aaba}) = \delta(\underline{\hat{\delta}(\text{even}, aab)}, a)$$

$$= \delta(\underline{\delta(\hat{\delta}(\text{even}, aa), b)}, a)$$

$$\cancel{= \delta(\delta(\hat{\delta}(\text{ev}}$$

(5) $= \delta(\delta(\delta(\hat{\delta}(\text{even}, a), a), b), a)$

$\cancel{= \delta(\delta(\delta(\delta(\text{even}, \varepsilon), a), a), b), a)}$

$= \delta(\delta(\delta(\delta(\underline{\hat{\delta}(\text{even}, \varepsilon)}, a), a), b), a)$

$= \delta(\delta(\delta(\underbrace{\delta(\text{even}, a)}_{\text{odd}}, a), b), a)$

$= \delta(\delta(\underbrace{\delta(\text{odd}, a)}_{\text{even}}, b), a)$

$= \delta(\underbrace{\delta(\text{even}, b)}_{\text{even}}, a)$

$= \delta(\text{even}, a)$

$= \text{odd}$

Alternat def: For any $q \in Q$ and $w = w_1 \cdots w_n \in \Sigma^*$ $(n \geq 0)$, and any state $r \in Q$, $\hat{\delta}(q, w) = r$ iff there exist a sequence of states $s_0, s_1, \ldots, s_n$ such that

(6) a) $s_0 = q$

b) $s_n = r$

c) for all $1 \leq i \leq n$,

$$s_i = \delta(s_{i-1}, w_i)$$

$$\begin{array}{cccccc} w = & w_1 & w_2 & \cdots & w_n \\ & \uparrow & \uparrow & & \uparrow \\ q = s_0 & s_1 & s_2 & \cdots & s_n \end{array}$$

$$q = s_0 \xrightarrow{w_1} s_1 \xrightarrow{w_2} s_2 \xrightarrow{w_3} \cdots \xrightarrow{w_n} s_n = r$$

<u>Thm</u>: These two defs agree.

~~Prob~~ <u>Prop</u>: $\hat{\delta}(q, a) = \delta(q, a)$ for $q \in Q$

and $a \in \Sigma$ (string of length 1)

<u>Def</u>: Let $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ be a DFA and w a string over A's alphabet. Say that A <u>accepts</u> w iff $\hat{\delta}(q_0, w) \in F$

Otherwise, say that A <u>rejects</u> w.

7 **Def**: Let Σ be an alphabet.

A language over Σ is any subset of Σ^* (any set of strings over Σ).

(L is a language over Σ iff $L \subseteq \Sigma^*$)

Def: Let A be a DFA with alphabet Σ.

The language recognized by A (the language of A) is the set of all strings over Σ that A accepts. ~~Denot~~ Denote this by $L(A)$

$$L(A) = \{ w \in \Sigma^* : A \text{ accepts } w \}$$

A recognizes $L(A)$.

Def: A language L is regular if there is some DFA that recognizes it.

Ex: These are all regular:

$\{ w \in \{a,b\}^* : w \text{ has an odd number of } a\text{'s} \}$

$\{ w \in \{a,b\}^* : w \text{ ends in } b \}$

(8) $\{w \in \{a,b\}^* ; w \text{ starts with } a\}$

DFA recognizing whether a string over $\{a,b\}$ has ~~aaba~~ aaba as a substring.