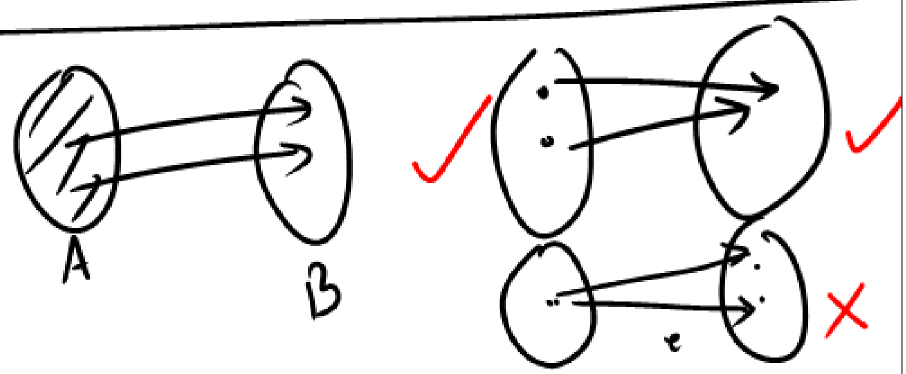
Lecture #3

- Deterministic Finite Automata
- Product Construction

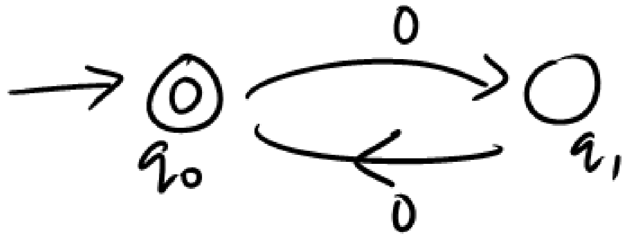
What is a function:

$$f : A \to B$$

f associates <u>every</u> element of A to <u>some</u> element of B



$\Sigma = \{0\}$



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

alphabet $\Sigma = \{0\}$

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

initial state $q_0 : q_0 \in Q$

Final states $F \subseteq Q$

Since δ is a function
- from every state, on every letter, there must be a new state
- this new state is unique

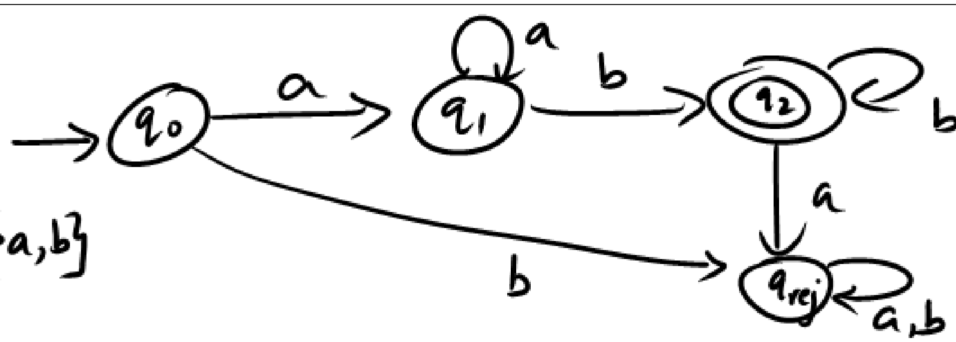
determinism

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

<u>where</u>

- Q is a nonempty finite set (called "states")
- Σ is a nonempty finite set (called the "alphabet"; elements of Σ are letters/characters)
- δ is a function from $Q \times \Sigma$ to Q
- $q_0 \in Q$ (called the "initial state")
- $F \subseteq Q$ (called the "final states")

A



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

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

where $Q = \{q_0, q_1, q_2, q_{rej}\}$

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

$q_0 = q_0$

$F = \{q_2\}$

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

δ	a	b
q_0	q_1	q_{rej}
q_1	q_1	q_2
q_2	q_{rej}	q_2
q_{rej}	q_{rej}	q_{rej}

Formal notion of acceptance

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

$w = a_1 a_2 \ldots a_k \in \Sigma^*$.

A accepts w if (and only if) there exists a sequence of states (in Q) $r_0, r_1, r_2 \ldots r_k$ such that

- $r_0 = q_0$
- $r_k \in F$
- $r_{i+1} = \delta(r_i, a_{i+1}) \quad i = 0, \ldots k-1$

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

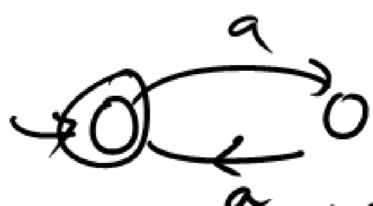
$Q = \{ \text{John}, \{r\} \}$

$\Sigma = \{a\}$

$q_0 = \text{John}$

$F = \{\text{John}\}$

δ	a
John	$\{r\}$
$\{r\}$	John



aa is accepted by A

since $\text{John} \xrightarrow{a} \{r\} \xrightarrow{a} \text{John}$ is

a sequence s.t.

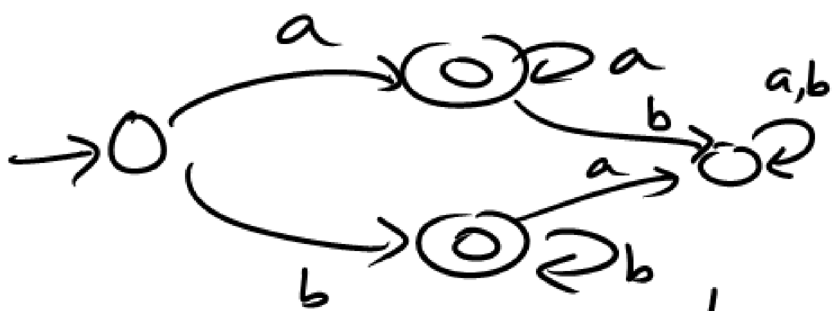
q_0

The language of A is the set of strings accepted by A.

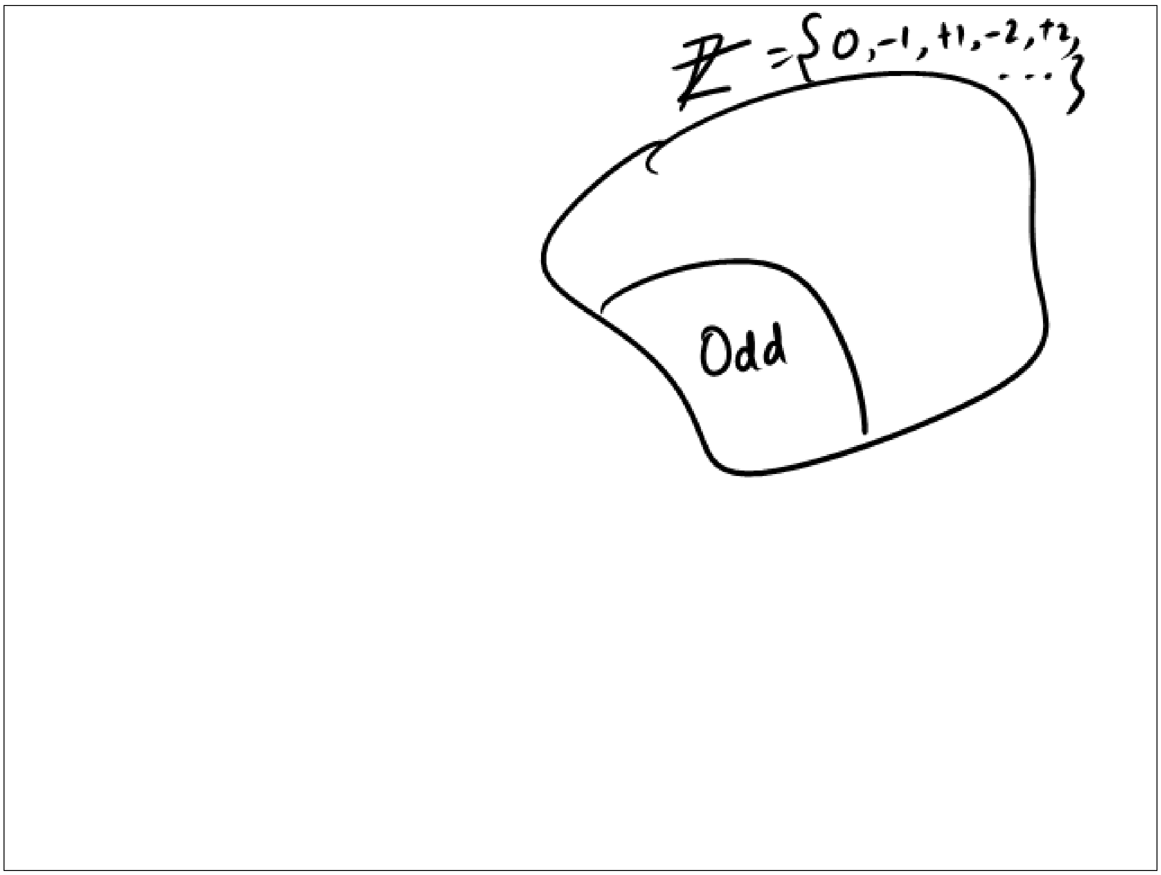
$$\mathcal{L}(A) = \{ w \in \Sigma^* \mid A \text{ accepts } w \}$$

Why multiple finite states

$$L = \left\{ \begin{array}{l} a, aa, aaa, \ldots \\ b, bb, bbb, \ldots \end{array} \right\}$$



Needs multiple final states!

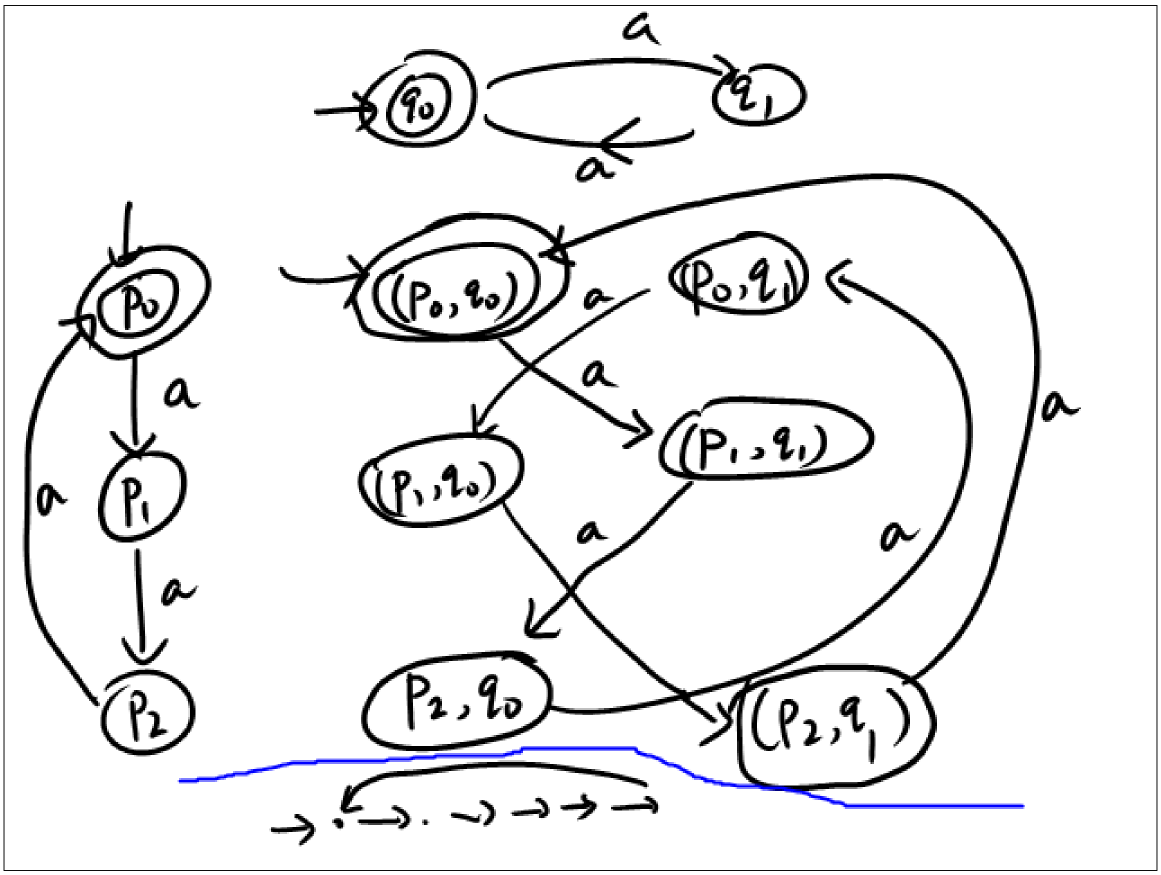

$\mathbb{Z} = \{0, -1, +1, -2, +2, \dots\}$
Odd

Operations on languages Σ

$L_1 \cup L_2 = \{ w \in \Sigma^* \mid w \in L_1 \text{ or } w \in L_2 \}$

$L_1 \cap L_2 = \{ w \in \Sigma^* \mid w \in L_1 \underline{\text{and}}\ w \in L_2 \}$

$\overline{L} = \{ w \in \Sigma^* \mid w \notin L \}$



a
q_0
q_1
a
p_0
a
p_1
a
p_2
a
(p_0, q_0)
(p_0, q_1)
a
a
(p_1, q_1)
(p_1, q_0)
a
a
a
p_2, q_0
(p_2, q_1)