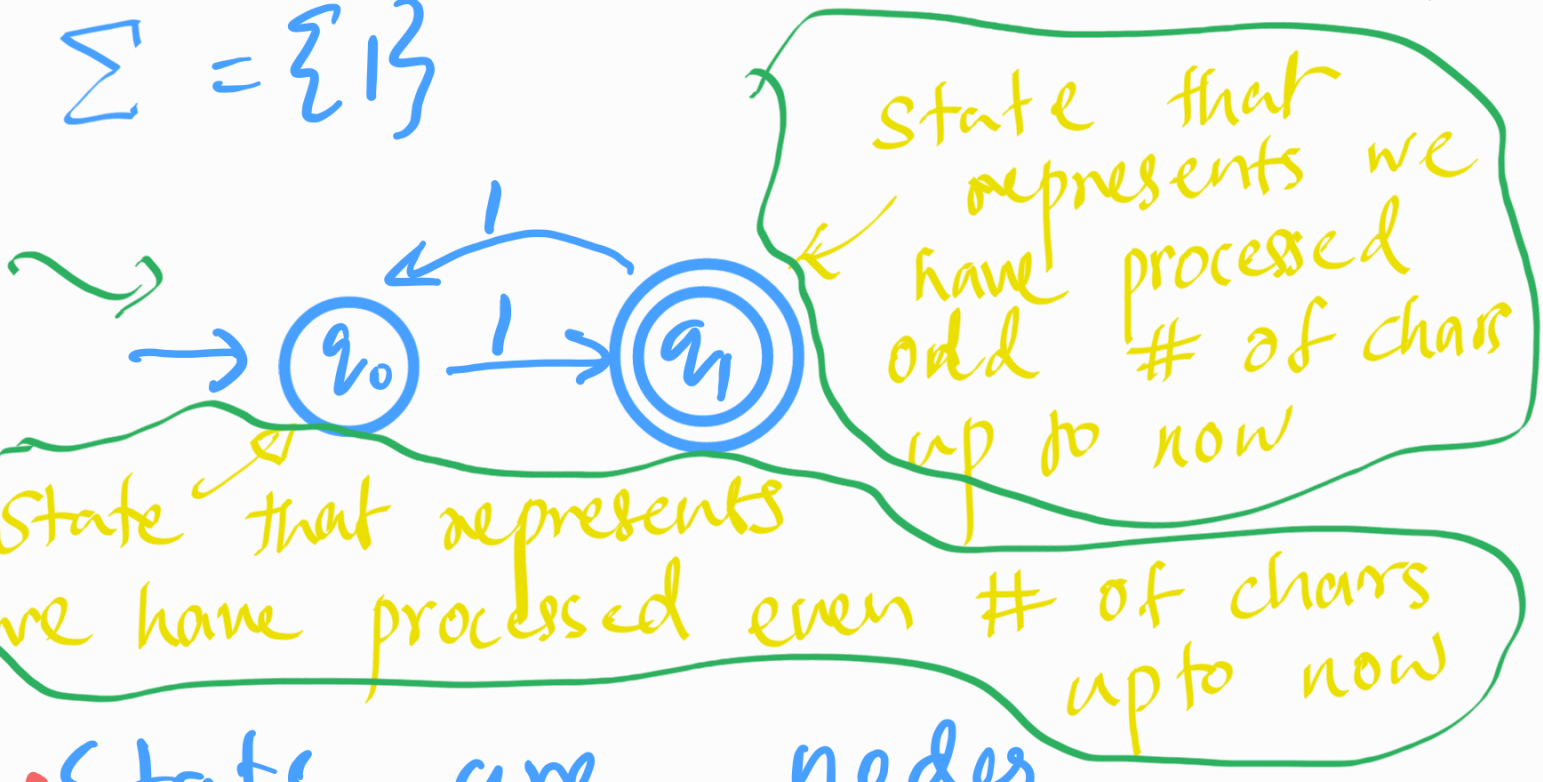
Ex: Design a finite state automaton which accepts the following language

$$A = \{w : \text{the length of } w \text{ is odd}\}$$

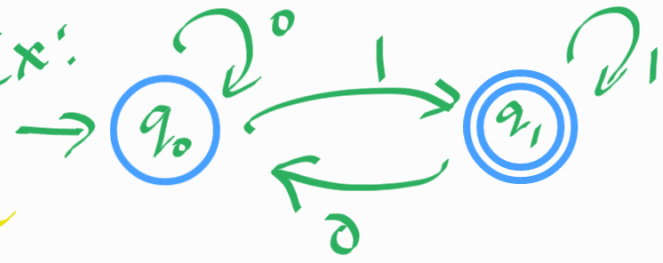
$$\Sigma = \{1\}$$



- State are nodes
  Some states are accepting states
- arrows represent transitions
- inputs are symbols available in the alphabet.

01/28/2025

Ex:



M

δ

Definition

- A finite state automaton is 5-tuple: $(Q, \Sigma, \delta, q_0, F)$

- Q is the set of states

$Q = \{q_0, q_1\}$

- Σ is the finite set called alphabet.

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

- $\delta : Q \times \Sigma \rightarrow Q$

$\langle q_0, 0\rangle, \langle q_0, 1\rangle \ldots$

- $q_0 \in Q$ is the start state.

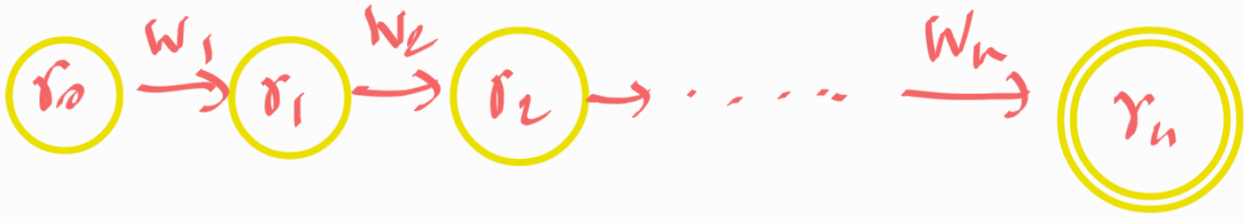
- $F \subseteq Q$ is the set of accepting states.

$F = \{q_1\}$

If A is the set of strings that is accepted by the machine M, then

$L(M) = A$

- Formally, given $M = (Q, \Sigma, \delta, q_0, F)$ and $w = w_1 w_2 w_3 \ldots w_n$, M accepts w if there is $r_0, r_1, r_2, \ldots, r_n$ such that, $r_0 = q_0$, $\delta(r_i, w_{i+1}) = r_{i+1}$, and $r_n \in F$



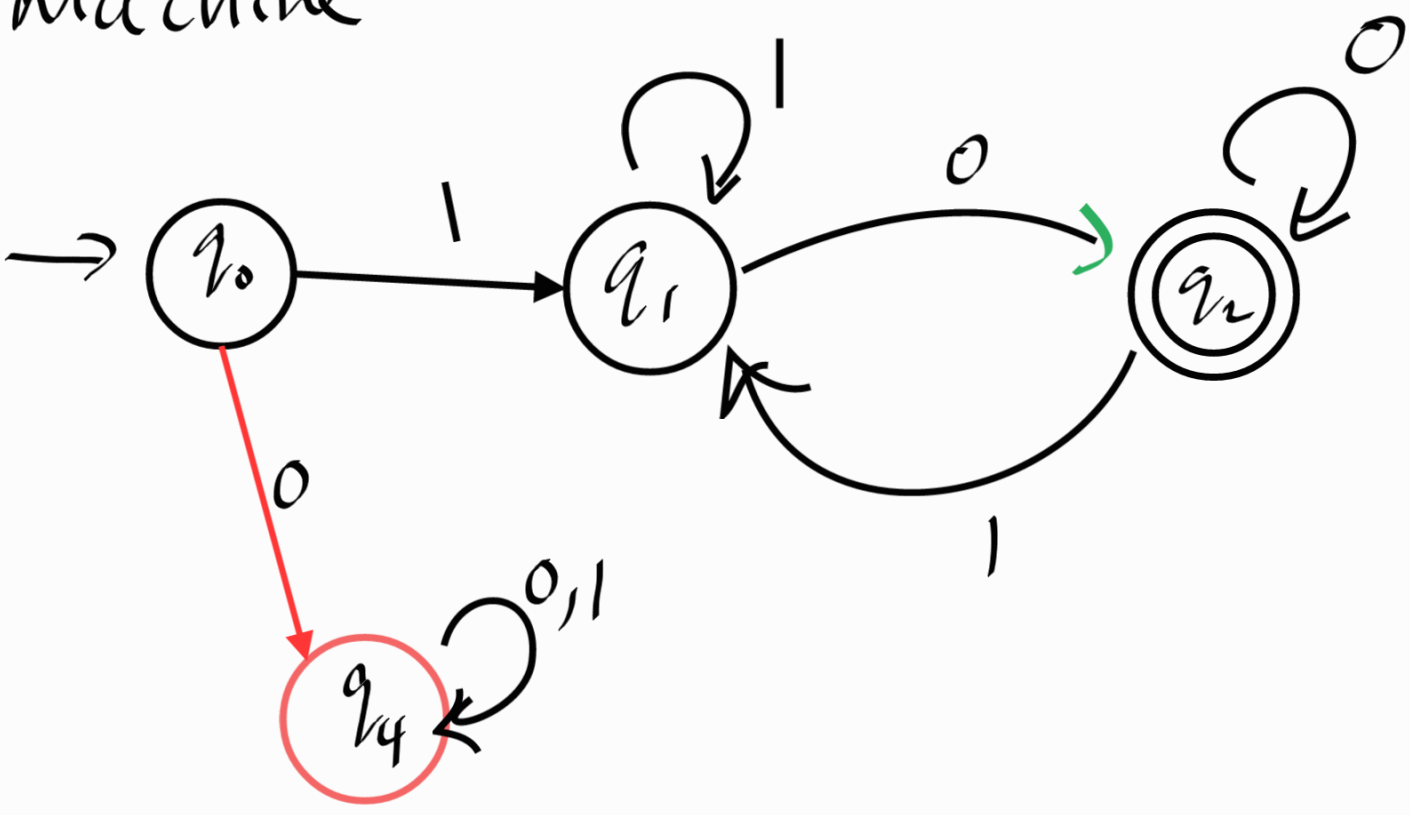
0 0 1 1 ✓

$q_0 \xrightarrow{0} q_0 \xrightarrow{0} q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_1$ ✓

0 1 0 ✗

$q_0 \xrightarrow{0} q_0 \xrightarrow{1} q_1 \rightarrow q_0$

Suppose you have following machine



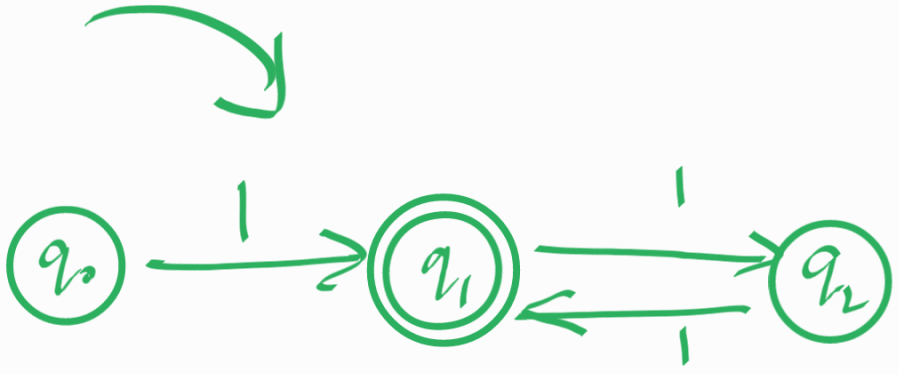
Question: Is q_4 needed?

Yes! we need this formally

but since this transition money to a state that will not lead to a accepting state we could drop it.

Consider the previous example where we created the machine that accepts odd length strings over alphabet $\Sigma = \{1\}$

How about following machine?

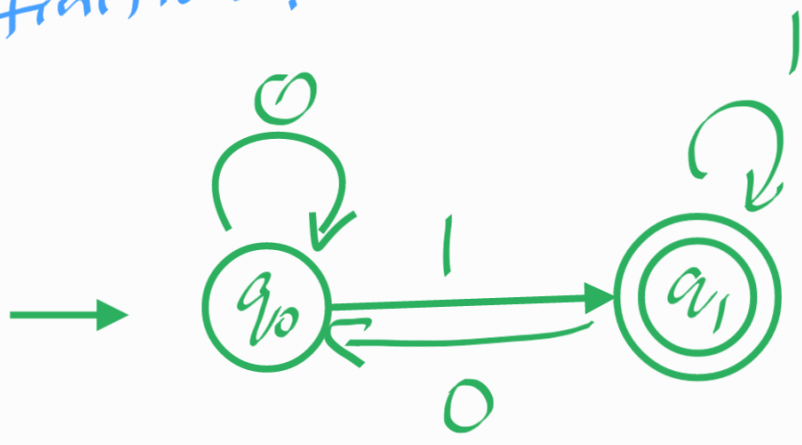


$\Sigma = \{1\}$

Do these machines accept the same language?

Yes

Let's look at an example finite automaton again and map it back to the original definition.



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

1. $Q = \{q_0, q_1\}$
2. $\Sigma = \{0, 1\}$
3. δ is described as follows:

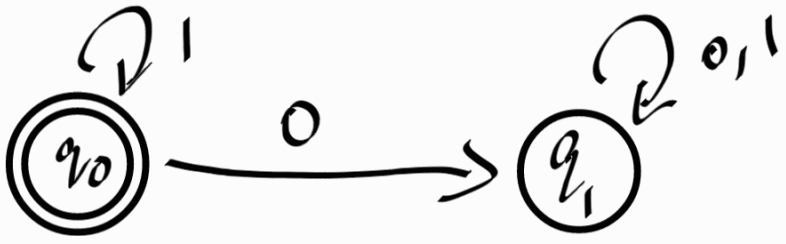
	0	1
$\underline{q_0}$	q_0	q_1
q_1	q_0	q_1

4. $\underline{q_0}$ is the start state
5. $F = \{q_1\}$

Let us try to identify languages accepted by following machines.

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

Ex1



ϵ

1

1^*

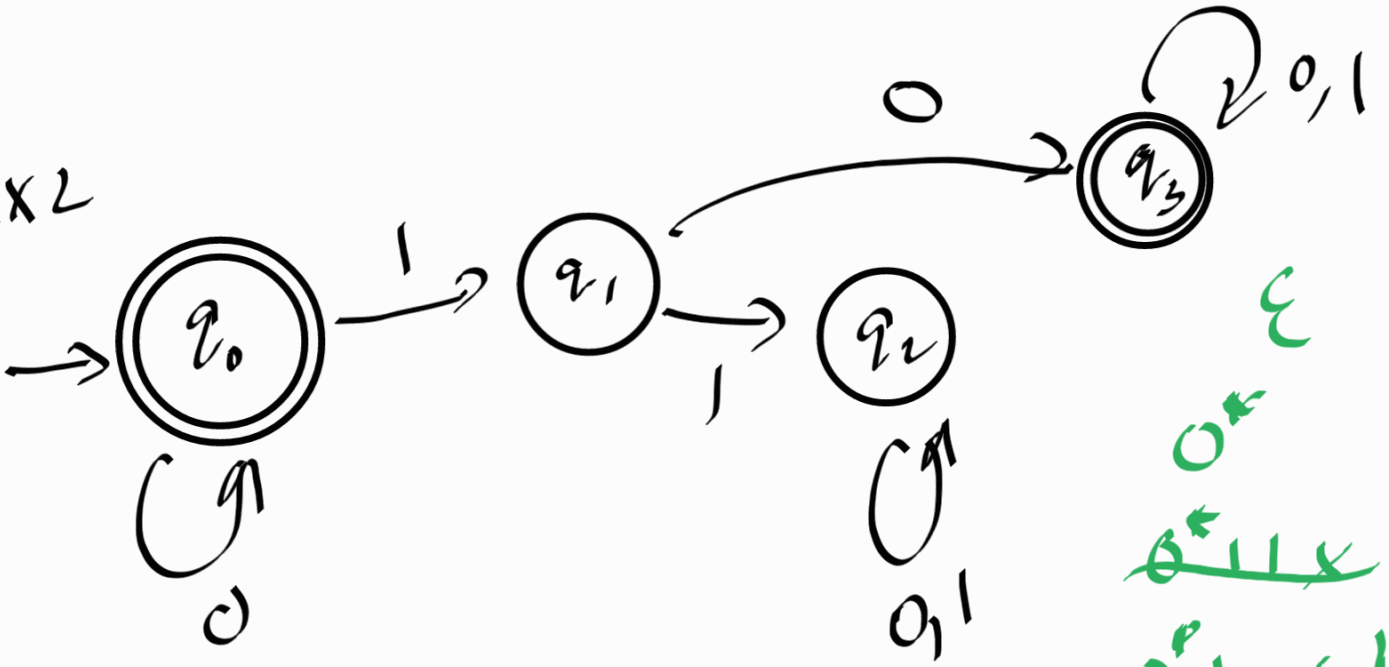
Kleene star

$\{1^n \mid n \geq 0\}$

$=$

$\{1^*\}$

Ex2



ϵ

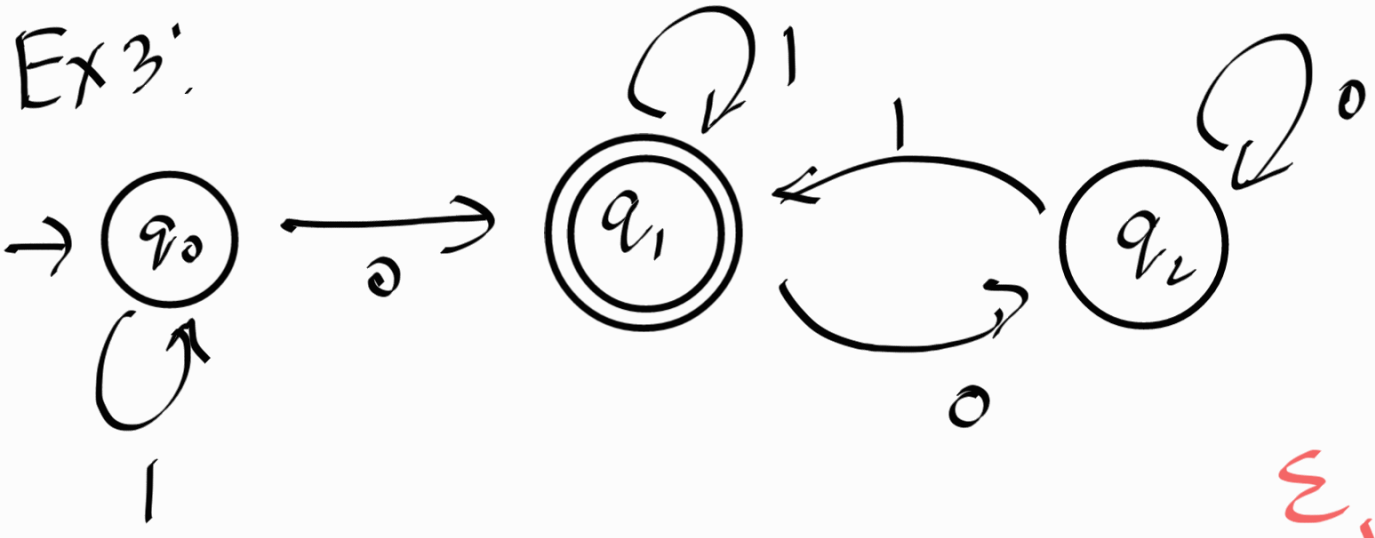
0^*

0^*11^*

$0^*10(0|1)^*$

$\{0^*, 0^*10(0|1)^*\}$

Ex 3:



$\{ 1^*01^*, 1^*01^*(01^+)^* \}$

$1^*01^*(00^*1)1^*$

$1^*01^*(01^+)^*$

ε, X

1^*01^* ✓

1^*01^*0 X

$1^*01^*00^*$ X

$1^*01^*00^*1$ ✓

$1^*01^*\underline{00}^*\underline{1}1^*$

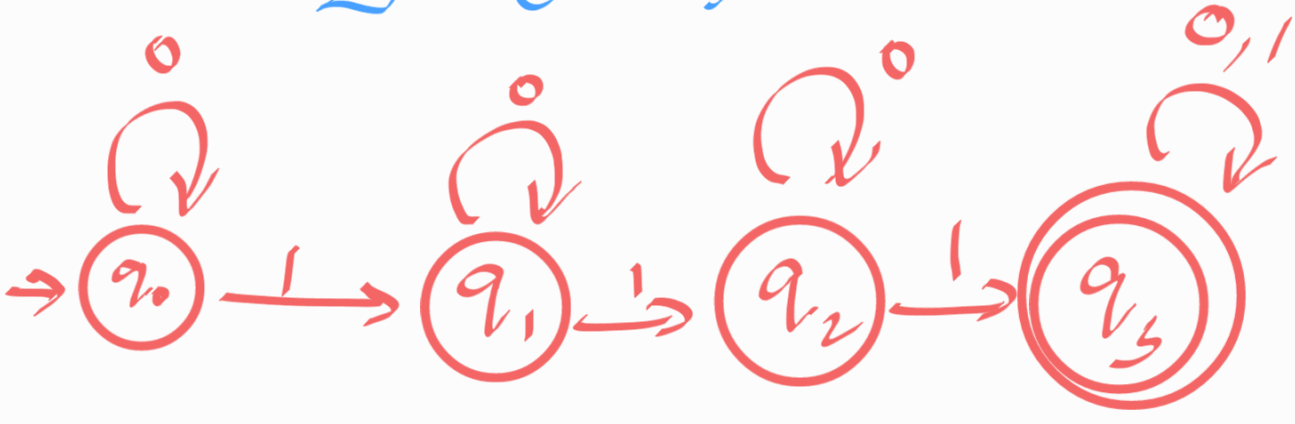
$1^*01^*0^+1^+$

A language is regular if some DFA accepts it.

Question: How do we design finite automaton?

Ex: Construct a DFA that accepts Language $A = \{w \mid w \text{ contains at least three 1's}\}$

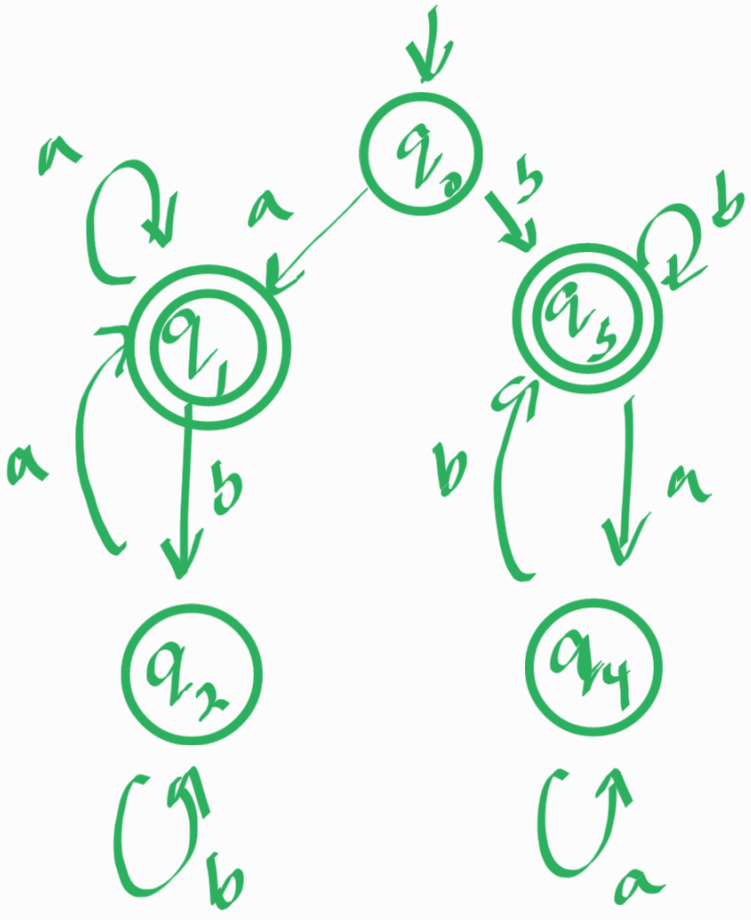
$$\Sigma = \{0, 1\}$$



$$0^*10^*10^*1(0|1)^*$$

Ex2: $A = \{ w \mid w$ starts and ends with the same symbol$\}$

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



ε ← not accepted

a ✓

ab ✗

aa ✓

bb ✓

aba ✓

baa ✗

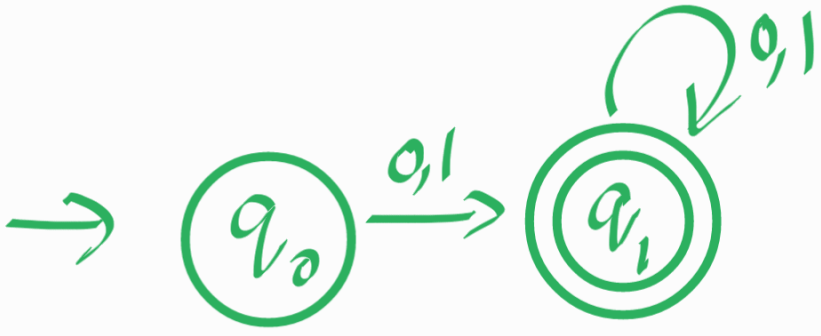
bab ✓

b...b ✓

a...a ✓

b ✓

Ex3: Everything but the empty string



Regular operations.

Let A, B be languages.

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

Concatenation: $A \circ B = \{xy \mid x \in A \wedge y \in B\}$

Star: $A^* = \{x_1 x_2 x_3 \ldots x_k \mid k \geq 0 \text{ and each } x_i \in A\}$

Let $\Sigma = \{a, b, c, d, \ldots, z\}$

If $A = \{good, bad\}$ $B = \{dog, cat\}$

Then,

$A \cup B = \{good, bad, dog, cat\}$

$A \circ B = \{gooddog, goodcat, baddog, badcat\}$

$A^* = \{\varepsilon, good, bad, goodgood, goodbad, badbad, badgood,$
$goodgoodgood, goodgoodbad, \ldots\ldots\}$

Side Note!

If $x \in \mathbb{N}$, $y \in \mathbb{N}$, $x + y \in \mathbb{N}$

we say natural numbers are closed under the addition operation

natural numbers are not closed under division operation

$7 \in \mathbb{N}$, $8 \in \mathbb{N}$, but $8/7 \notin \mathbb{N}$

Theorem 1.25

The class of regular languages are closed under the union operation,

In other words,

If A_1, A_2 are regular languages, so is $A_1 \cup A_2$

Proof ⇒

By definition let M_1 be a DFA recognizing A_1, i.e

$$M_1 = (Q_1, \Sigma_1, \delta_1, q_1, F_1)$$

Similarly A_2 with

$$M_2 = (Q_2, \Sigma_2, \delta_2, q_2, F_2)$$

We construct $M = (Q, \Sigma, \delta, q_0, F)$ as follows!

1. $Q = \{(r_1, r_2) \mid r_1 \in Q_1, r_2 \in Q_2\}$
2. $\Sigma = \Sigma_1 \cup \Sigma_2$
3. $\delta((r_1, r_2), a) = (\delta_1(r_1, a), \delta_2(r_2, a))$
4. $q_0 = (q_1, q_2)$
5. $F = \{(r_1, r_2) \mid r_1 \in F \text{ or } r_2 \in F\}$

Ex.