$\Sigma$: finite alphabet
$\Sigma^*$: all finite length strings over $\Sigma$ including $\varepsilon$ (zero length)

Language $L \subseteq \Sigma^*$
We are interested in membership queries for a given language $L$.

String $S = x_1, x_2, x_3, \ldots, x_n$

- We can store at most some "constant" $n_0$ of input symbols
- We can scan the input only once

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

$L_1 = \{ x \in \{0, 1\}^* \mid \# 0's \text{ is divisible by } 3 \}$
Idea: Keep track of the "mod class of # 0's mod 3". 0, 1, 2

Initially we are in mod class [0]

next symbol

\[ 0 \rightarrow [x] \rightarrow [(x+1) \mod 3] \]

\[ 1 \rightarrow \text{no change in mod class} \]

1. There are finite 'states' (correspond to mod 3 classes) does not depend on length of string

2. What happen when the input symbol is 0, 1
Finite State Automaton / Machine

\[ \Sigma: \text{alphabet} \]
\[ Q: \text{set of final states} \ 9_1, 9_2, 9_3 \]
\[ q_0: q_0 \in Q \quad \text{initial state} \]
\[ F: F \subseteq Q \quad \text{final states} \]
\[ \delta: Q \times \Sigma \rightarrow Q \quad \text{transition function} \]

\[ \delta \left( q_0, 0 \right) = 9_1, \quad \delta \left( q_0, 1 \right) = 9_3, \quad \delta \left( q_0, 2 \right) = 9_2 \]

\[ M = ( \Sigma, Q, q_0, F, \delta ) \]

M corresponds to some language \( L \)

\[ L(M) = \{ x \in \Sigma^* \mid \delta(q_0, x) \in F \} \]

\[ \delta^*(q, a) = \delta(q, a) \quad a \in \Sigma \]

\[ \delta^*(q, ax) = \delta^*(\delta(q, a), x) \quad x \in \Sigma^* \]

\[ \delta^*(q, \epsilon) = q \]
Given some language description $L$, design a FSA $M$ s.t.

$L(M) = L$ (Nothing more, nothing less)

Given a machine $M$, show that

$L(M) = \{ \}$
What things does this machine accept?

0 1 0 1 0 1

Claim: For all strings \( w \in \{0, 1\}^* \)

\[ w \in L(M) \text{ iff } w \text{ has even } 0's \text{ and even } 1's \]

(i) \( w \in L(M) \Rightarrow w \text{ has even } 0's \)

\[ g(9, w) = 9, \Rightarrow w \text{ has even } 1's \]

(ii) If \( w \text{ has even } 0's \) \( w \in L(M) \)

\( w \in L_{0011} \)
We will prove (i) by using Induction on length of string.

For all \( n \geq 0 \), statement (i) is true.

For all \( w \in \{0,1\}^* \), statement (i) is true for \( |w| > 0 \).

Base case:

\( |w| = 0 \) \( \in \) \( \in L(M) \) and in accepted \( \in \in L_{0011} \).

Suppose for all \( w \in \{0,1\}^* \) \( |w| = n \).

Statement (i) is true.

We want to show that (i) is true for \( |w| = n+1 \).

\( w : w'0 \) or \( w'1 \) \( |w'| = n \)

\( \delta(q, wa) = \delta(\delta(q, w), a) \)

\( w'0 \in L_{0011} \) \( w' \notin L(M) \)

\( w'1 \notin L_{0011} \) \( w' \in L(M) \)

(i) \( \delta(q_1, w) = \begin{cases} \delta(q_1, w) & \text{if } w \in L_{0011} \\ q_2 & \text{if } w \in L_{011} \\ q_3 & \text{if } w \in L_{01} \\ q_4 & \text{if } w \in L_{001} \end{cases} \)
By induction on length \( l \) of string \( w \), we want to show that if \( w \in L_{011} \), then \( \hat{S}(q, w) = 9_1 \).

If \( w \in L_{011}, L_{001}, L_01 \), then there is nothing to prove.

Inductive step: \( w = w'0 \cdot w'' \).

For \( w' \), we use induction hypothesis.

Assignment due by 12 noon on Fri.

\[ L' = \{ \omega \in \Sigma_0, \Sigma_1 | \text{the third last symbol in } 1 \text{ \text{is } } 0, 1 \} \]