# Minimization problem

Minimization problem (for fixed $\Sigma$)

Given: DFA $A$

Output: DFA $B$ s.t. $L(A) = L(B)$ and $B$ has the smallest number of states possible for recognizing $L(A)$

Algorithm

Let $Q = \{q_1, \ldots, q_n\}$.

1. For each $1 \leq i < j \leq n$, initialize $T(i,j) = --$
2. For each $1 \leq i < j \leq n$

   If $(q_i \in F \text{ AND } q_j \notin F)$ OR $(q_i \in F \text{ AND } q_j \notin F)$
   $T(i,j) \leftarrow \checkmark$
3. Repeat

   $\{$ For each $1 \leq i < j \leq n$
   If $\exists a \in \Sigma$, $T(\delta(q_i, a), \delta(q_j, a)) = \checkmark$
   then $T(i,j) \leftarrow \checkmark$
   $\}$

Untill $T$ stays unchanged.

# Module - II: Different models of computation

2DFA: Two-way deterministic finite state automata.

| $\# \ w_1 \ w_2 \ \dots \ \dots \ \dots \ \dots \ \dots \ \dots \ w_n \ \$$ |
|---|

Input head moves left/right on this tape.

It does not go to the left of $\#$.

It does not go to the right of \$.

Can potentially get stuck in an infinite loop!

# Formal definition of 2DFA

## Definition

A 2DFA $A = (Q, \Sigma \cup \{\#, \$\}, \delta, q_0, q_{acc}, q_{rej})$, where

| | | | |
|---|---|---|---|
| $Q$ : | set of states, | $\Sigma$: | input alphabet |
| $\#$: | left endmarker | $\$$: | right endmarker |
| $q_0$: | start state | | |
| $q_{acc}$: | accept state | $q_{rej}$: | reject state |

$\delta : Q \times (\Sigma \cup \{\#, \$\} \rightarrow Q \times \{L, R\}$

The following conditions are forced:

$\forall q \in Q$, $\exists q', q'' \in Q$ s.t. $\delta(q, \#) = (q', R)$ and $\delta(q, \$) = (q'', L)$.