Defn $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{acc}, q_{rej})$

Q: finite state set

Σ: finite input alphabet set ; $\sqcup \notin \Sigma$ "blank"

Γ: finite tape alphabet . $\Sigma \cup \{\sqcup\} \subseteq \Gamma$.

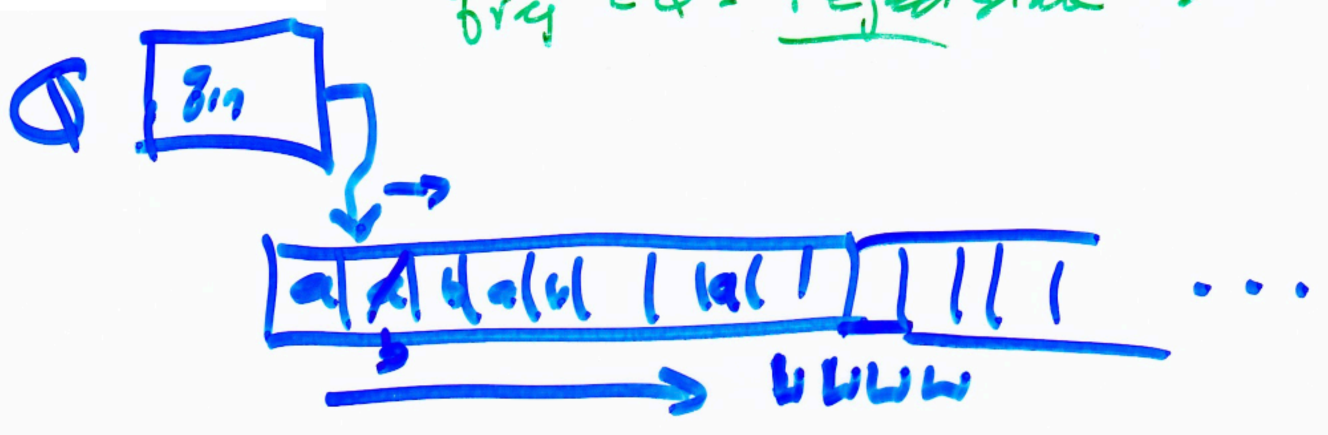
δ: $Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$ transition function

$q_0 \in Q$: start state

$q_{acc} \in Q$: accept state

$q_{rej} \in Q$: reject state

$q_{acc} \neq q_{rej}$



By definition, no transitions *out* of q_{acc}, q_{rej};

M *halts* if (and only if) it reaches either

M *loops* if it never halts ("loop" might suggest "simple", but non-halting computations may of course be arbitrarily complex)

M *accepts* if it reaches q_{acc},

M *rejects* by halting in q_{rej} or by looping

The language *recognized* by M:
$L(M) = \{ w \in \Sigma^* \mid M \text{ accepts } w \}$

L is *Turing recognizable* if ∃TM M s.t. L=L(M)

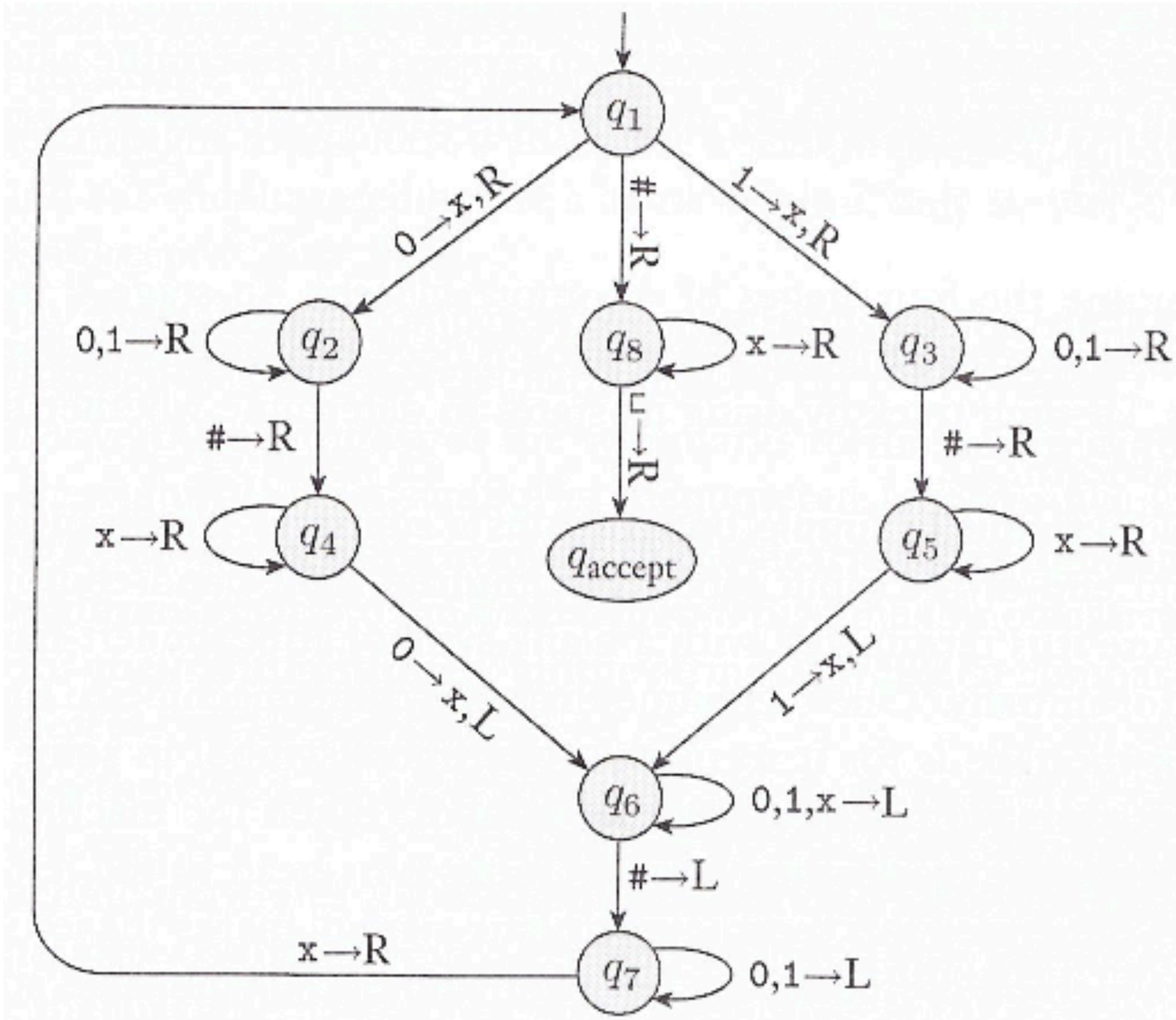
L is *Turing decidable* if, furthermore, that M halts on all inputs

A key distinction!

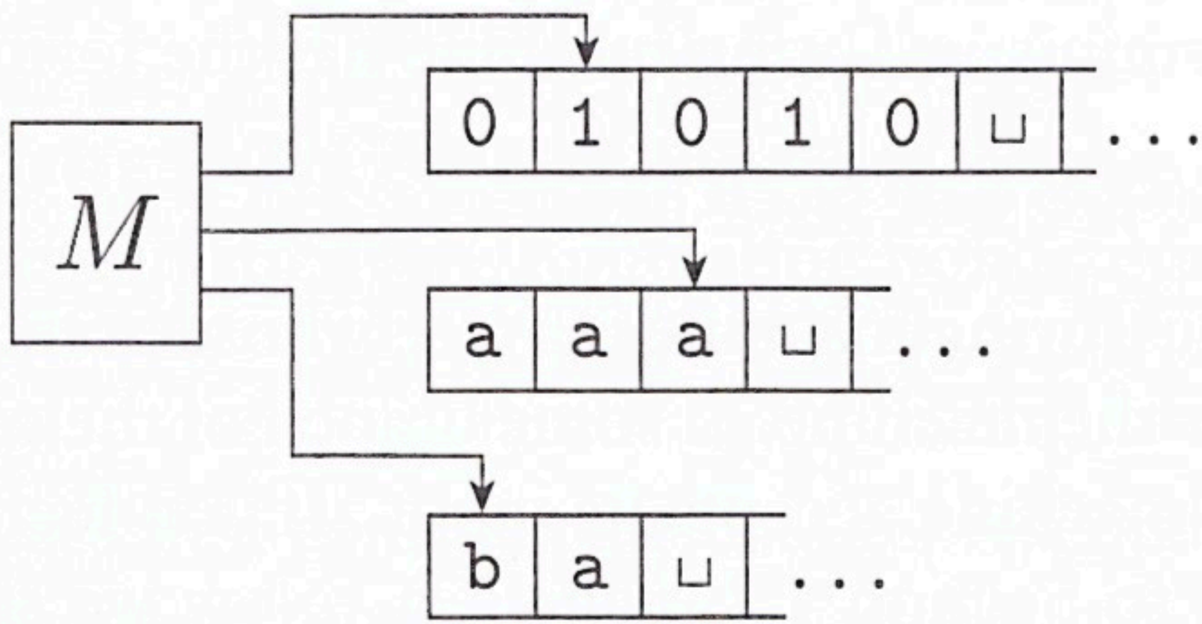
Example

$$L = \{ w\#w \mid w \in \{0,1\}^* \}$$

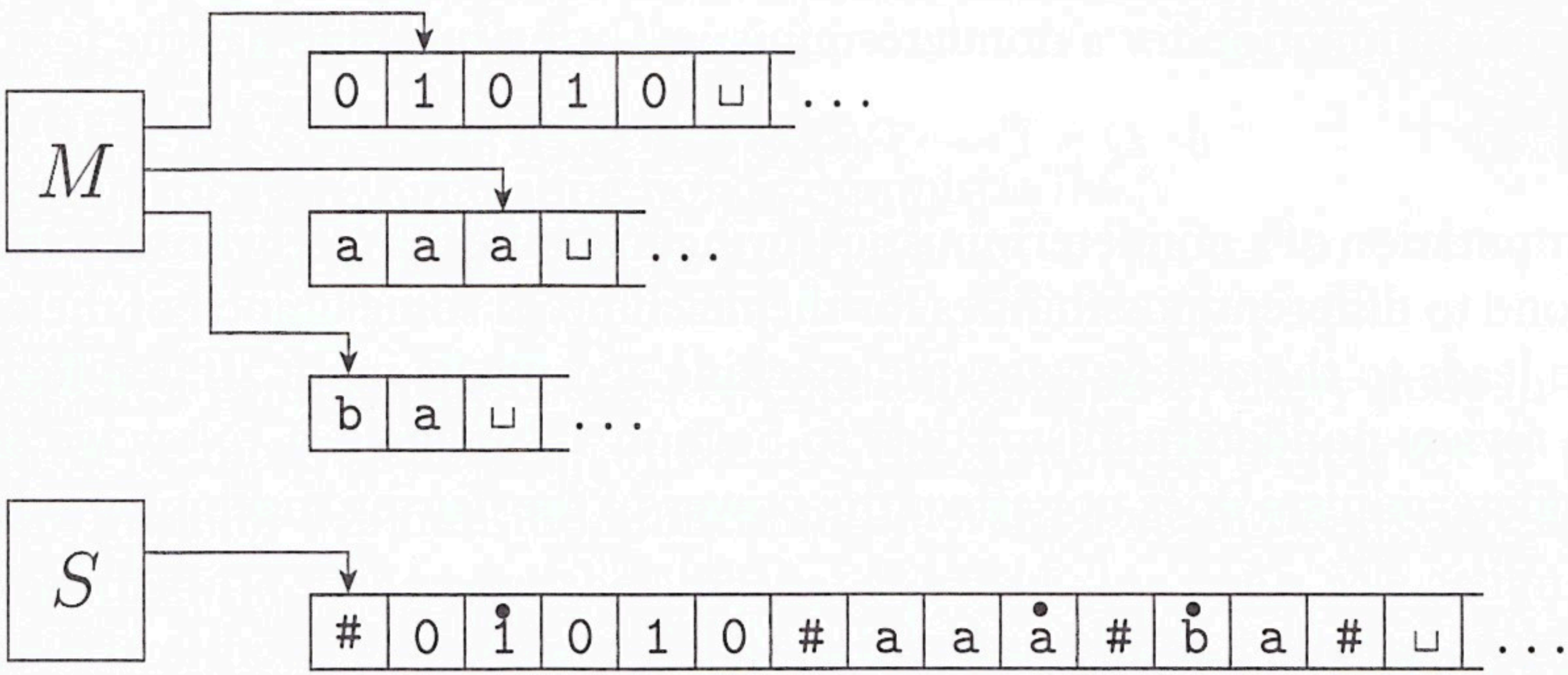
1. check that there's a single #
2. read, remember & cross off leftmost uncrossed letter
3. scan to # & compare next uncrossed letter
4. If OK, cross it off
5. repeat



q_1
$0 \rightarrow x,R$
$\# \rightarrow R$
$1 \rightarrow x,R$
$0,1 \rightarrow R$
q_2
q_8
$x \rightarrow R$
q_3
$0,1 \rightarrow R$
$\# \rightarrow R$
$\sqcup \rightarrow R$
$\# \rightarrow R$
$x \rightarrow R$
q_4
q_{accept}
q_5
$x \rightarrow R$
$0 \rightarrow x,L$
$1 \rightarrow x,L$
q_6
$0,1,x \rightarrow L$
$\# \rightarrow L$
$x \rightarrow R$
q_7
$0,1 \rightarrow L$

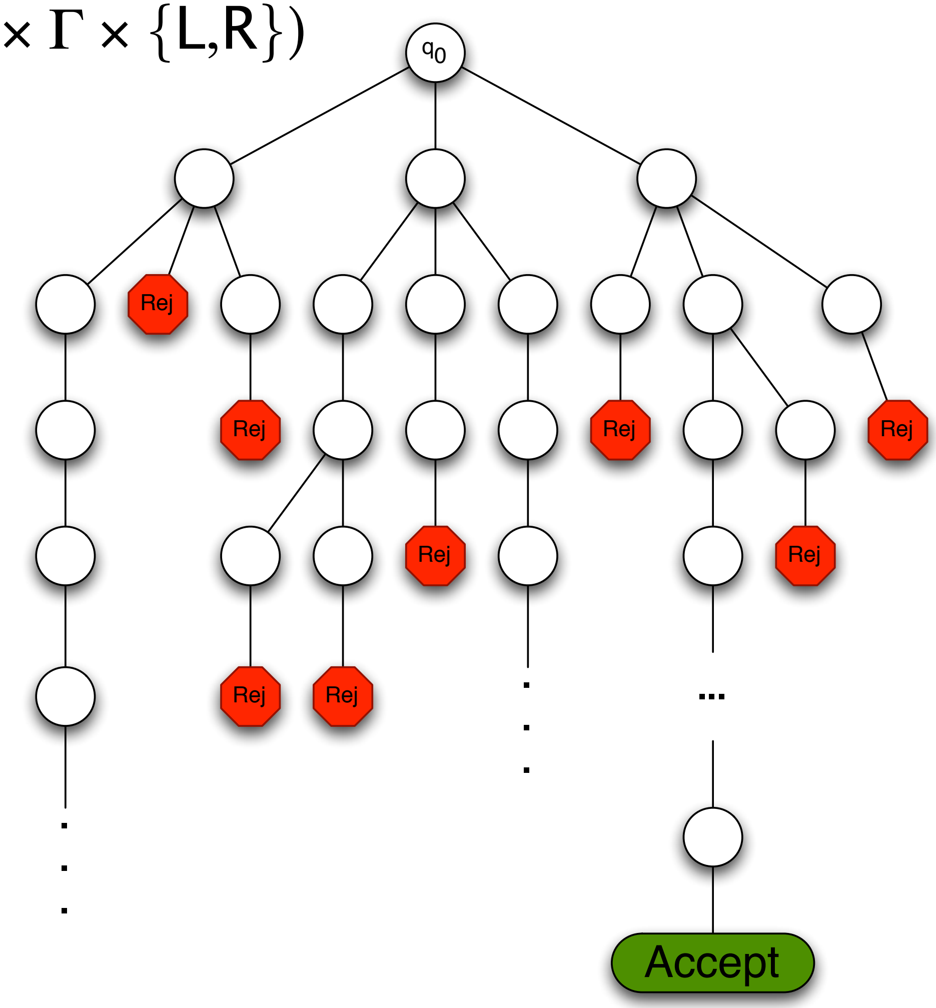


$$\delta \colon Q \times \Gamma^k \longrightarrow Q \times \Gamma^k \times \{\mathrm{L}, \mathrm{R}, \mathrm{S}\}^k$$


M
0 1 0 1 0 ⊔ ...
a a a ⊔ ...
b a ⊔ ...
S
0 1 0 1 0 # a a a # b a # ⊔ ...

Nondeterministic Turing machines:

$\delta: Q \times \Gamma \to \mathcal{P}(Q \times \Gamma \times \{L,R\})$



Nondeterministic Turing machines:

$$\delta: Q \times \Gamma \rightarrow \mathcal{P}(Q \times \Gamma \times \{L,R\})$$

Accept if *any* path leads to q_{accept}

Reject if all (halting) paths lead to q_{reject}