DEFN ("is in state q")

M ends in state q after reading $w \in \Sigma^*$ if

(1) $w = w_1 w_2 \cdots w_n$
where $w_i \in \Sigma$

(2) $\exists$ state $r_0, r_1, r_2 \cdots r_n \in Q$
s.t. (a) $r_0 = q_0$
(b) $\forall\ 1 \le i \le n$
$\delta(r_{i-1}, w_i) = r_i$

(c) $r_n = q$

$\boxed{\text{Fact: } q \text{ is unique}}$
because δ is a function, basically

Defn:

<u>M accepts w</u> $\Longleftrightarrow$ the state, q, reached by M after reading w is an accepting state, i.e. $q \in F$

Defn:

the Language <u>recognized</u> by M, $L(M) = \{ \underbrace{w}_{\in \Sigma^*} \mid M \text{ accepts } w \}$

Note

Every M recognizes exactly one language. Implicitly, it "recognizes" ~~when it must~~ both strings it must accept and those it must reject.

$L \subseteq \Sigma^*$ is <u>regular</u> iff $L = L(M)$
for some Finite Automaton M

$L(M) = \Sigma^*$

Note: M <u>accepts</u>
every palindrome, eg → aaabaaa
but also some nonpalindromes, like → ab
It <u>recognizes</u> Σ^*:
$L(M) = \Sigma^*$; $L_{pal} \subseteq L(M)$ but $\neq$

Regular Languages, Examples:
"even parity" is regular
"3rd from right is a"
odd length

Are there general ways to prove languages regular, other than making more & more example M's? <u>Yes</u>

Theorem

if $M = (Q\ \Sigma\ \delta\ q_0\ F)$

accepts L (i.e. $L = L(M)$)

Then $M' = (Q, \Sigma, \delta, q_0, Q - F)$

accepts $\Sigma^* - L$

Proof

M accepts $w \Leftrightarrow$

it is in a state $q \in F$

after reading w

$\Leftrightarrow$ M' is in state q

after reading w.

But $q \in F$ so

$q \notin Q - F$

$\therefore$ M' rejects w

Regular Languages are <u>Closed</u> under <u>complementation</u>