- Today: Turing machines, multi-tape, etc...
- Question: Is there a Turing machine that correctly solves the question $P \overset{?}{=} NP$?

=

Cheap shot: The is a TM that correctly answers "is P=NP?".

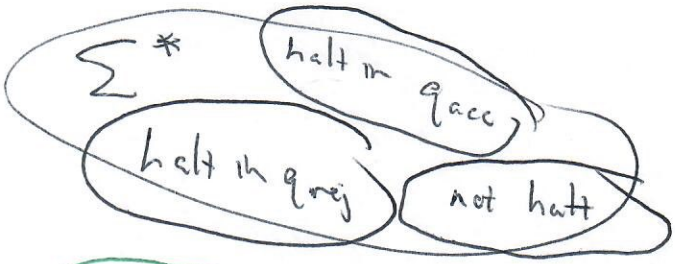
↳ Just by definition: yes, but we don't know which one...

Knowing there is a TM to solve "blah" is not the same as knowing which one...

In [Sip] 3.1: "recognizing" verses "deciding"

A Turing machine, M, on input $w \in \Sigma^*$ can (1) halt in q_{acc}, (2) halt in q_{rej} (3) not halt "looping"

Each M:



We say M recognizes $L \subset \Sigma^*$ if

$w \in L \Rightarrow M$ halts in q_{acc} on input w

$w \notin L \Rightarrow M$ " " q_{rej} or loops (doesn't halt) on input w

M decides L means M recognizes L and always halts in q_{acc} or q_{rej}

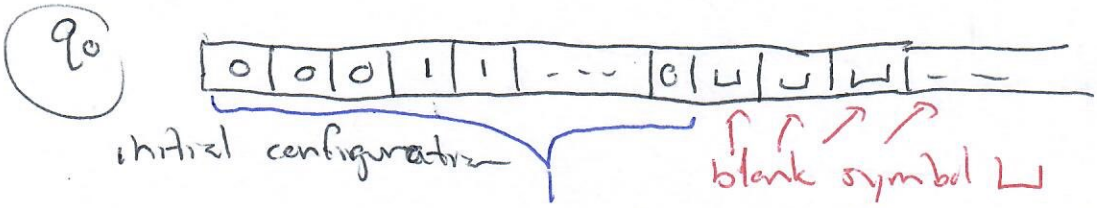
- Today: TM that <u>decides</u> $\{0^n 1^n\}$
- TM that takes input $\{0,1,\#\}$ accepts strings:

$$\{ s_1 \# s_2 \# s_3 \mid s_1, s_2, s_3 \in \{0,1\}^* \text{ and } s_1 = s_2 + s_3 \text{ in binary} \}$$

- multi-tape machine = high level description of $\delta, \Gamma, \ldots$ (precise) $\leftrightarrows$ imprecise

---

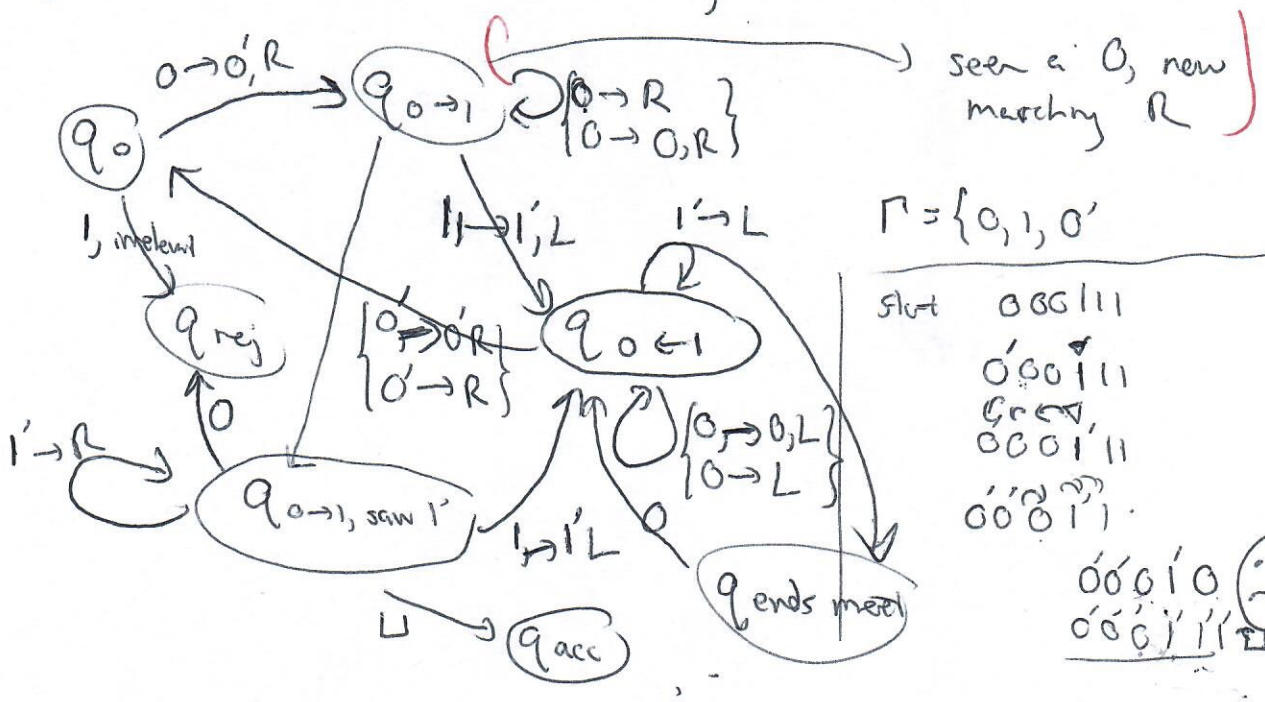
Last time started a Turing machine to recognize $\{0^n 1^n\}$:



input: string over Σ, work tape alphabet Γ contain Σ, contain ␣, ␣ $\notin \Sigma$

High level:
look for first 0, mark R through 0's to first 1
hitting first 1 we move left to 2nd 0, etc.

So:



We also showed notation on page 173, Example 3.9
And figure 3.4, top of page 169 for configuration notation