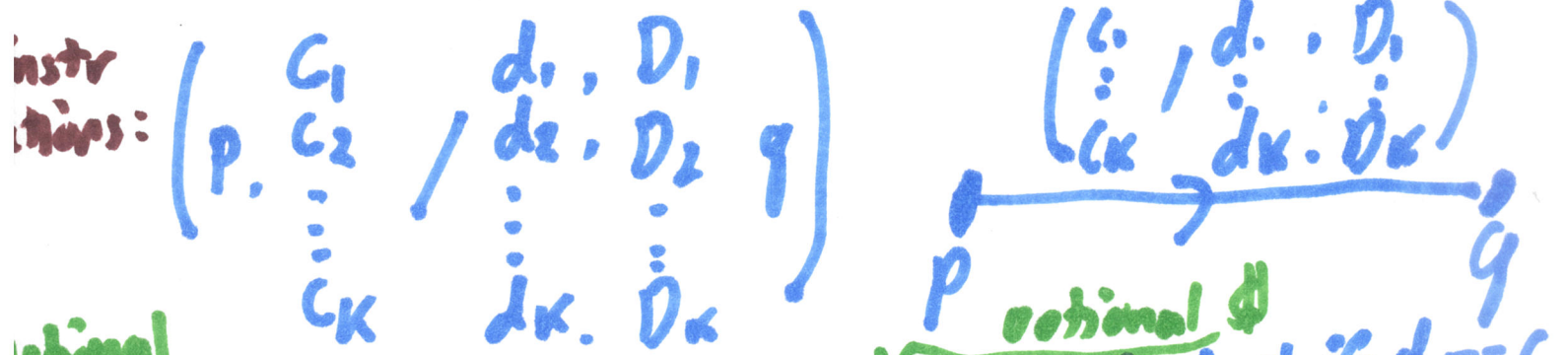


A multitape TM has  $M = (Q, \Sigma, \Gamma, \delta, \dots, s, \{q_{acc}\})$

with  $\delta \subseteq (Q \times \Gamma^k) \times (\Gamma^k \times \{L, R, S\}^k \times Q)$  for some  $k \geq 1$ .



optional: tapes are in write to the left as well.

$L(M) = \{x \in \Sigma^* : M \text{ on input } x \text{ has a legal way to } \}$   
execute instructions that ends in  $q_{acc}$

$M$  is det<sup>c</sup> (a DTM) if  $\delta$  is a function from  $(Q \setminus \{q_{acc}, q_{rej}\}) \times \Gamma^k$  to  $\Gamma^k \times \{L, R, S\}^k \times Q$ . "Extra Nice Form":  $M$  never writes a  $\perp$  between

two nonblank chars. <sup>between</sup>  
For a tape  $j$  writes  $d_j = \perp$  whenever the head moves  $D_j = L$ , then that tape is a stack. A PDA is a 2Tape TM whose input is <sup>one-way</sup> read-only and has a stack as tape 2.

# What basic ops can a TM perform?

• Check whether two strings  $x, y$  are equal.

Say stored as labels of

$[x \# \underline{v}]$  "registers"  
 $[y \# \underline{\quad}] [y' \# \underline{xv}]$

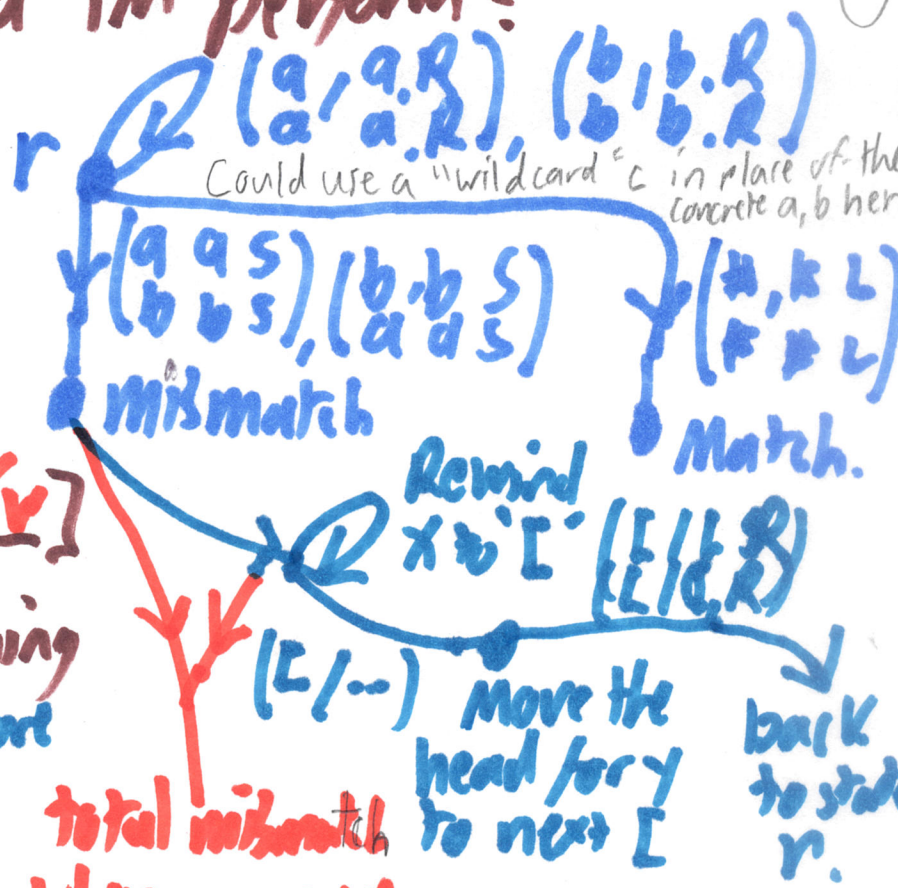
• Search for a  $y$  matching a given  $x$ . Put the above code inside a loop.

• Copy a string from one tape to another.

• Perform basic arithmetic:  $+$ ,  $-$ ,  $\times 2$  are enough  
Example: TM for the " $3n+1$ " problem. Likewise,

• Test whether a given string is empty or '1' (or zero).

• Do conditional jumps to instructions with a matching label.



total mismatch when no more [s on tape with y's.

These ingredients suffice to simulate a rich enough (!) assembly language.

in particular, every program in your favorite high-level language can be compiled to a DTM MP st. for all ASCII inputs  $x$ ,  $M_p(x) = P(x)$ .

[Show Universal RAM-TM Handout (Better than Univ. TM!)]  
Then bolted on to my simulator