

5/24/2010

Two more weeks in quarter.  
 Holiday: Monday, May 31  
 5 more lectures including this one.  
 1 more homework will be assigned Friday + due on last day of class.  
 Final exam is on Monday of finals week.

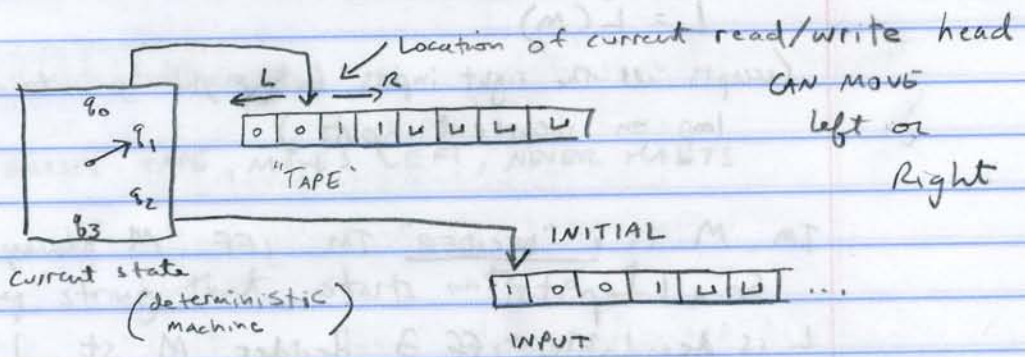
Turing Machines (TMs)

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

↓                      ↓  
 start                      Tape alphabet (we call it a "tape" alphabet for historical reasons)

$q_{acc}$  → M halts and accepts input

$q_{rej}$  → M halts and rejects input



$$\delta: Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$$

INPUT CAN BE USED OR IGNORED

TO STAY IN SAME LOCATION: JUST MOVE ONE WAY + THEN BACK AGAIN

CONFIGURATION:  $0q_1011$

- In state  $q_1$
- Reading the second 0

Final configuration's

0110  $q_{acc}$   
 ---  $q_{acc}$  ---  
 ---  $q_{rej}$  ---

$M$  halts IFF  $M$  enters  $q_{ACC}$  or  $q_{REJ}$

INPUT  $w$ :

$M$  on  $w$ : 2 outcomes

$M$  HALTS  $\begin{cases} \rightarrow \text{ACCEPTS} \\ \rightarrow \text{REJECTS} \end{cases}$

$M$  LOOPS

$M$  accepts  $w \iff M$  reaches  $q_{ACC}$

$M$  rejects  $w \iff M$  reaches  $q_{REJ}$  or  $M$  never halts

$L(M) = \{w \mid M \text{ accepts } w\}$

$L$  is Turing-Recognizable iff  $\exists$  a TM,  $M$  s.t.

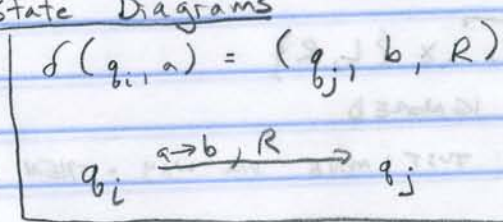
$L = L(M)$

(accepts all the right inputs, but might go into an infinite loop on incorrect inputs)

TM,  $M$  is a "DECIDER" TM IFF  $M$  always halts for all inputs

$L$  is decidable iff  $\exists$  decider  $M$  s.t.  $L = L(M)$

State Diagrams



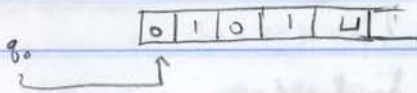
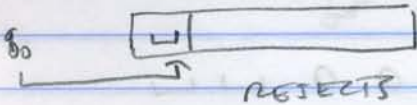
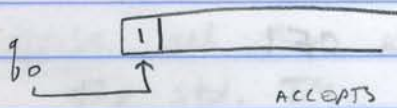
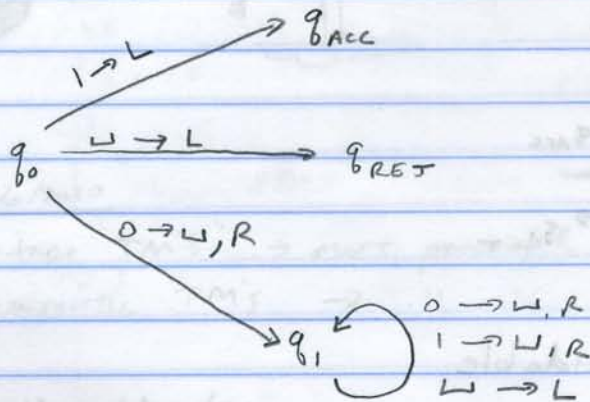
$q_i \xrightarrow{a, R} q_i$

$q_i \xrightarrow{a \rightarrow R} q_i$

} If you don't change the tape contents, just move to the right

$\sqcup = \text{BLANK}$

M:

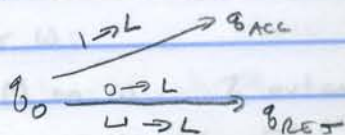


Accepts any string that starts with a 1:

$$L(M) = 1\Sigma^*$$

$1\Sigma^*$  is Turing-RECOGNIZABLE

DECIDER  $M'$ :



$\Sigma^*$  is decidable

$L = \{0^n 1^n 0^n \mid n \geq 0\}$  not a CFL

Is  $L$  decidable?



Use marker symbol "x"

Mark the 0,

move forward to first 1, mark it

move forward to first 0, mark it

Go back to beginning and repeat

That algorithm would break on this:

010010

So need a preprocessing step to make sure it's in the right format (0's, then 1's then 0's)