Announcements

- Last homework: **HW11** on computability due May 1st

Section plans: this week : quiz + computability practice

week of May 4-5: quiz + review of material since the prelim

Final (cumulative): Saturday May 9th at 9am.

Thursday May 7: 4820 review led by a few TAs time tba

- Friday, May 1st: divide and conquer
- Monday, May 4th ??

Wed: SAT is NP-complete

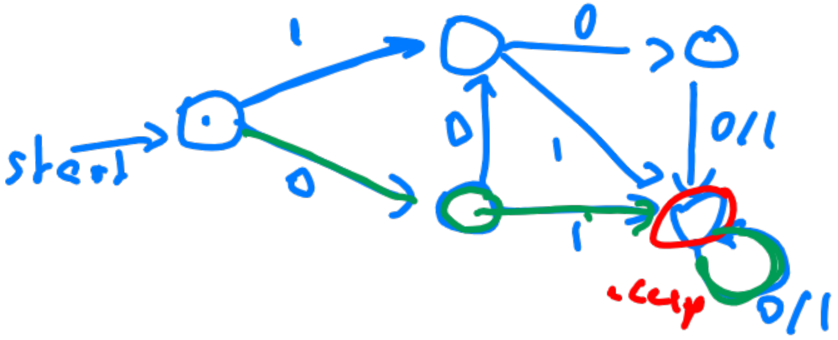
- CS theory club: https://theoryclub.cs.cornell.edu/
- **Wednesdays at 5:00 PM — CIS 450**
- **5:00-5:30 PM:** Reception/Social
- **5:30-6:30 PM:** Talk
- Pizza provided!
- Talk this Wednesday;: **Joey Rivkin,** first-year PhD student in theoretical computer science.
- **Defying Gravity with Range Minimum Queries**

The Range Minimum Query problem asks: given an array of integers, how do we answer the query: "What is the minimum element between indices *i* and *j*?" We will see some beautiful ideas and data structures for answering many such queries efficiently, with runtimes that seem to defy ~~intuition~~ **gravity**.

Today: Turing Machine: the simplest model of what is computation. Alan Turing (1936)

Recall: finite automata

Graph



input 0110

edges labeled by characters

accept if end on a node labeled accept

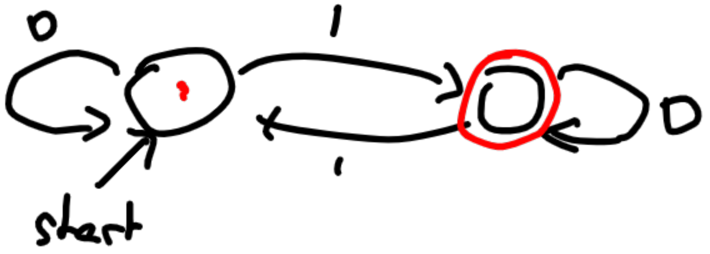
states: S

start $s \in S$

accept $\subseteq S$

transition: $S \times \Sigma \to S$

character



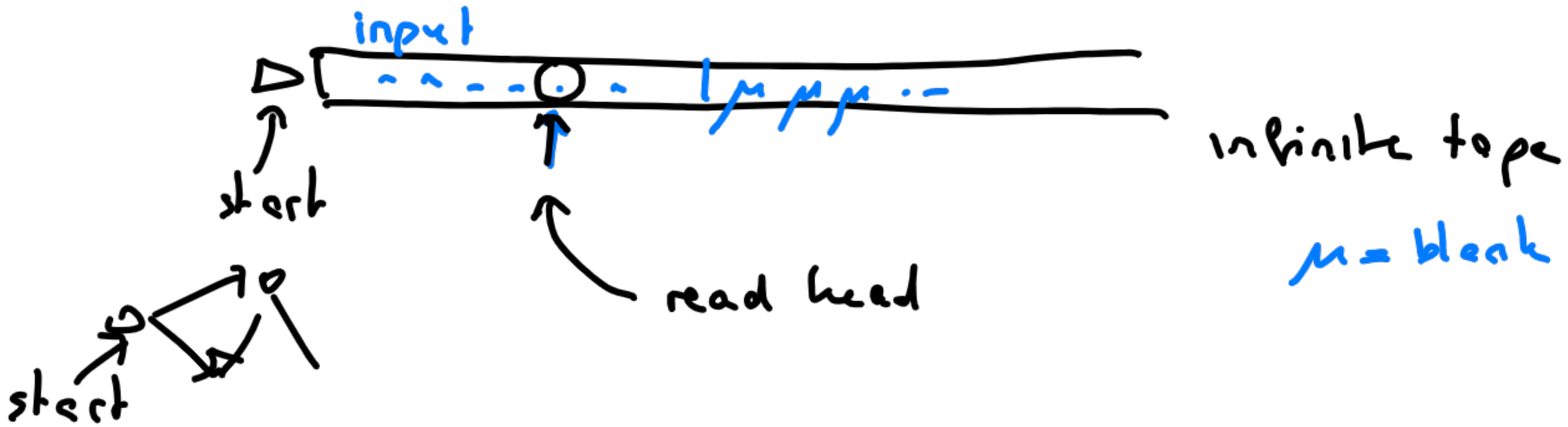
$L \Rightarrow 00\ldots010\ldots0$

= strings with odd # of 1's

Recall 2800 $L = \{0^n 1^n : n \text{ integer}\}$ 00001111

Today: Turing Machine: the simplest model of what is computation. Alan Turing (1936)

= finite automata with a tape



input : Σ

$\delta : S \times \Sigma \to S \times \Sigma \times \{+, 0, -\}$

in state $q \in \Gamma$ reading $\sigma \in \Sigma$

change to new state q', write σ' & move one character left/right/no move

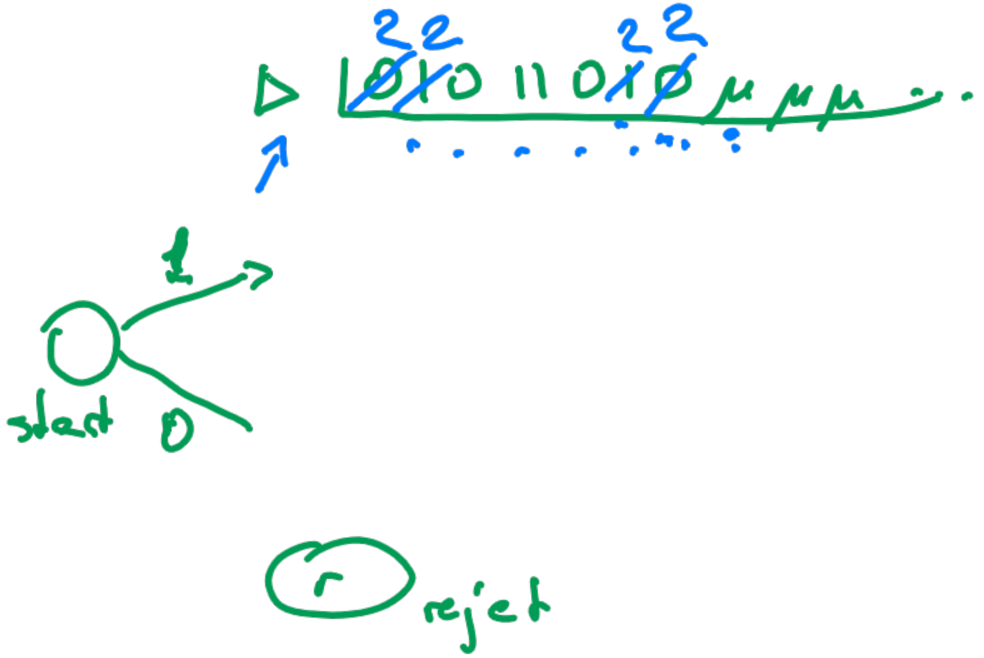
reach any accept state = accept

Turing Machine: examples

Recognizing palindrome:

01011010

$\Sigma = \{0, 1, 2\}$



start : first symbol & change to 2
go to the end
if last not the same ⇒ not palindrome
if same & change to 2

same backwards on next letters . . .

time: $O(n^2)$

Turing Machine: examples

$L = \{0^n 1^n : n \text{ integer}\}$

\# steps: $O(n^2)$

1. check if $0^n 1^m$
2. change one 0 → 2 & 1 → 2

accept if end no 0 or 1 left

Alternate:

$O(n^2)$

increment counter while seeing 0's

decrement — " — — " — 1

accept if counter ends at 0

Church-Turing Hypothesis

⇒ anything computable
is computable by a Turing machine

Poly time Phython/Java =? poly time on Turing machine

? Quantum computers
quantum computabe = computable by Phython
run time ? exponential

troble: ① discrete log poly time on quantum computer
given prime q
base g
& A
find a: $g^a \equiv A \pmod q$

② RSA
given two prime p & q (secret)
n = pq ~~given~~: find p & q

Turing Machine more examples

$\mathcal{L} = \{ 1^{2^n} : n \text{ integer} \}$

$\{1, 11, 1111, 11111111, \dots \}$

1's a power of 2

$\Sigma = \{0, 1\}$

1. check if input 1^m
   - if only one 1 ✓ accep
   - else check if # 1's even, reject if odd
   - change every other 1's to a 0 $\Longrightarrow$ 2^{n-1} 1s should be left

repeat: