Announcements

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

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

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

If you have a conflict, please fill out this form

Thursday May 7, 3-7pm, in Hollister B14: final review led by a few TAs

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

TA applications are due May 3rd

- 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 today: **Joey Rivkin**, first-year PhD student in theoretical computer science **& our TA**.
- **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**.

Review: materials Prelim 1+2 see handouts

for material since Prelim 2 & advice for final: handout will be on canvas later today

Today: SAT is NP-complete

So far SAT & 3-SAT NP-complete (promise)

We used it reduction to prove other things NP-complete

SAT $\leq_p$ Indep Set $\Rightarrow$ Indep Set NP-complete

to prove SAT NP-complete:

all $X \in NP$ $X \leq_p$ SAT

definition: $\exists$ algorithm A

takes input x & hint y $A(x,y)$ yes/no

x is yes for problem X if & only if $\exists$ y such that $A(x,y)$ yes

Example Hamiltonian Path: input x = graph G

y = Hamiltonian path in G $\Big\}$ A

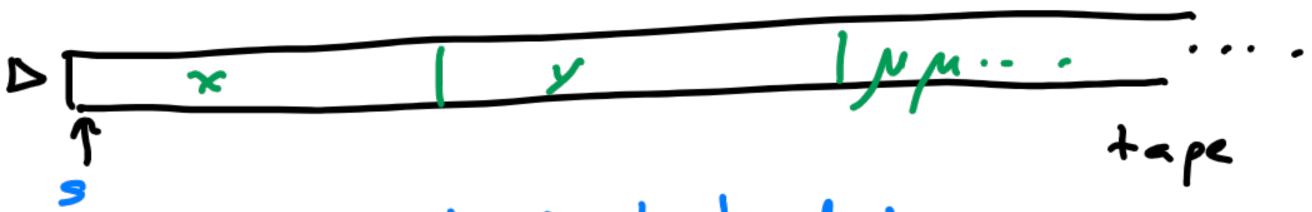
Recall Turing Machine (and the Church-Turing hypothesis)

$\Longrightarrow$ $\exists$ Turing machine M : input (x,y) answer yes/no

algorithm A as a Turing machine, poly time

Reminder:

finite automata + memory



Q = states

Σ = alphabet

δ : transition

head start at start of tape

$s \in Q$ start state

$\mu \in \Sigma$ blank

$$\delta : \Sigma \times Q \longrightarrow \Sigma \times Q \times \{+,-,0\}$$

symbol at head read

state

write

new state

move 1 position l or r

$A \subseteq Q$ accept if you reach these

Recall SAT

$$\phi = (x_1 \vee \bar{x}_2 \vee x_3 \vee \bar{x}_4) \wedge \dots$$

$x_1 \dots x_n$ variables

$c_1 \dots c_m$ clauses

$$\phi = \bigwedge_{j=1}^{m} c_j$$

clauses are $\vee$ or variables or their negations

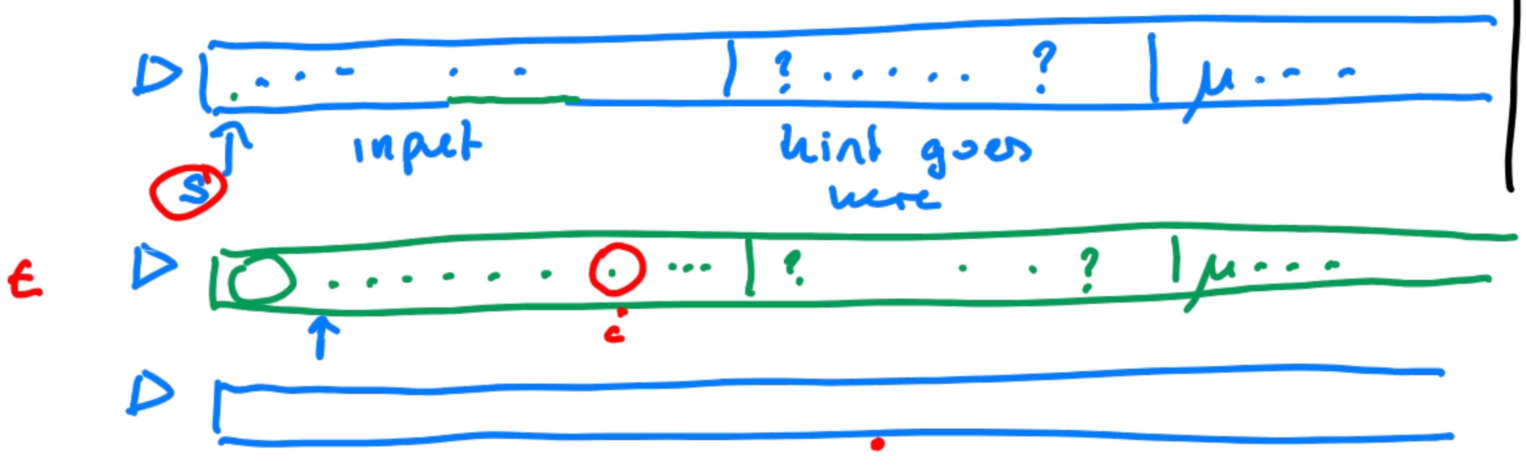
Reduction plan:

given Turing machine that proves our problem in NP

generate equivalent SAT formula

The Idea of the NP $\leq_P$ SAT

only till memory position T



state at start

state after 1 step

state after 2 steps

⋮

up to step T.

Idea 1: suppose: Turing machine time polynomial

max time for this algorithm T (poly in input size)

$\boxed{T^2 \text{ sized table}}$

List of variables to use in SAT

- $x_{t,i,\sigma}$ = true if at time t at position i on tape the symbol written is σ
- $y_{t,i}$ = true if at time t the head is at position i
- $z_{t,q}$ = true if at time t the state is q

Join by Web PollEv.com/evatardos772

How many variables did we define so far, if n is the input size and m and T the bounds on the size of hint and the running time (with m and T bounded as a polynomial in and n). Recall that the size of the alphabet and the number of states are constants

A. $O(n+m)$

B. $O(T)$

C. $O(T^2)$ ✓

D. More than either

E. I have no clue

- $X_{t,i,\sigma}$ = true if at time t at position i on tape the symbol written is σ

$T^2|\Sigma| = O(T^2)$

- $Y_{t,i}$ = true if at time t the head is at position i

size of table T^2

- $Z_{t,q}$ = true if at time t the state is q

$T|Q| = O(T)$ $|Q|$ finite

Constraints that should be true and how to write them as clauses

to make sure Turing machine proves that input x is yes for problem

- start state is s — $z_{0,s} = T$ — $(z_{0,s})$
- head at start — $y_{0,0} = T$ — $(y_{0,0})$

input $(\sigma_1 \ldots, \sigma_k)$ — $x_{0,i,\sigma_i} = T$ for all $i = 1, \ldots, k$

if accepted hint size is m — $x_{0,i,\mu} = T$ for $i \in [k+m+1, T]$

start a TM look correct input & "hint" &

- each position only one character $\forall i, \forall t$ $\left(\bigvee_{\sigma} x_{t,i,\sigma}\right) \wedge \bigwedge_{\sigma \neq \sigma'} \left(\bar{x}_{t,i,\sigma} \vee \bar{x}_{t,i,\sigma'}\right)$

at least on symbol

do not write two

- head should be at exactly one position each time

$$\forall t \quad \left(\bigvee_{i} y_{t,i}\right) \wedge \bigwedge_{i \neq j} \left(\bar{y}_{t,i} \vee \bar{y}_{t,j}\right)$$

...

Constraints that should be true and how to write them as clauses

similer for many more rule

With all rules added

$\exists$ input for y positions

so that M accepts iff only if input accepted by M