$(u,v).f$ $\quad f(u,v)$

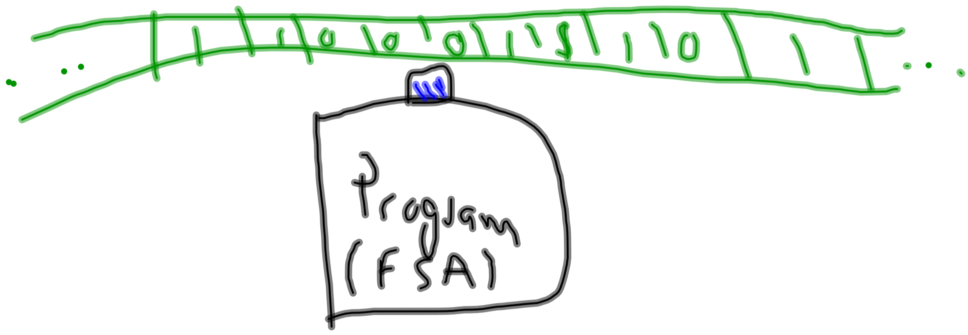
$v.d$ $\quad d(v)$

Nov 24-4:07 PM

NP-complete

Class of problems for which
either all have poly-time algs.
or none have poly " "

Nov 24-4:10 PM



Nov 24-4:27 PM

np.pdf (application/pdf Object) - Mozilla Firefox

http://www.columbia.edu/~cs2035/courses/csor4231.F09/np.pdf

Program to test if a binary number is even. Input is \$ termina
Output is written immediately after \$, 1 for yes, 0 for no.

- Read until \$ (state q_0)
- Back up, check last digit (state q_1)
- if even, write a 1 (states q_2, q_3, q_F)
- if odd, write a 0 (states q_4, q_5, q_F)

... 0|0|1|1|0|\$|1

1|0|1|1|\$|0

Here is a program. Each cell is (new state, write symbol move)

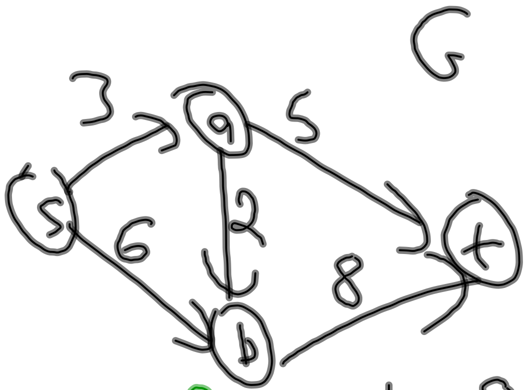
state	input 0	input 1	input \$
(q_0)	$(q_0, -, R)$	$(q_0, -, R)$	$(q_1, -, L)$
(q_1)	$(q_2, -, R)$	$(q_4, -, R)$	error
(q_2)	error	error	$(q_3, -, R)$
(q_3)	$(q_F, 1, -)$	$(q_F, 1, -)$	$(q_F, 1, -)$
(q_4)	error	error	$(q_5, -, R)$
(q_5)	$(q_F, 0, -)$	$(q_0, 0, -)$	$(q_0, 0, -)$
(q_F)	halt	halt	halt

Nov 24-4:36 PM

Any computer/compiler can be "simulated" on a turing machine with only a poly amt. of slowdown.

P is invariant to computer/language/compiler issues

Nov 24-4:40 PM



$S = \{s, a, t\}$ YES

$S = \{s, b, t\}$ NO

$S = \{s, b, a, t\}$ NO

Is L sorted?

2, 7, 10, 12

~~b = 8~~

longest path

b = 14

$S = \{s, b, t\}$

Nov 24-4:46 PM

SAT

$x_1 = F$ $x_4 = T$
$x_2 = F$
$x_3 = F$

$$\phi \equiv (\overset{F}{x_1} \vee \overset{T}{\bar{x}_2} \vee \overset{F}{x_3}) \wedge \overset{T}{\bar{x}_1} \wedge (\overset{F}{x_2} \vee \overset{T}{x_4}) \wedge (\overset{F}{x_1} \vee \overset{T}{\bar{x}_3})$$

is there a way to set vars. so that ϕ is true

YES

$$(x_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2)$$

NO

Nov 24-4:56 PM

Reductions

$Y \leq X$ we can use X as a "subroutine for solving Y"

Y is subtraction of two ints
X is addition of two ints (negation is easy)

$(6,4)$. You know how to negate. Your friend knows how to add.

Want to compute $6-4$

- Alg. Negate the 4 $\Rightarrow (6,-4)$. Give it to your friend.
- Friend return $6+-4 \rightarrow 2$.
- You have 2.

Nov 24-5:05 PM

np.pdf (application/pdf Object) - Mozilla Firefox

Definition $Y \leq X$ means

- Y is polynomial time reducible to X, which means

there exists a polynomial time computable function f that maps inputs to Y to inputs to X, such that

input y to problem Y returns "Yes" iff input $f(y)$ to problem X returns "Yes"

Informally $Y \leq X$ means that Y is "not much harder than" ("easier than") X

$y \to f(y)$ input to $X \to \frac{\text{Yes}}{\text{No}}$

Nov 24-5:11 PM

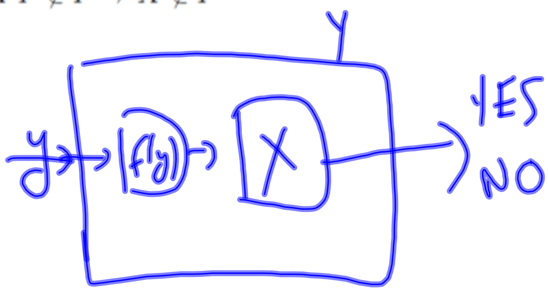
np.pdf (application/pdf Object) - Mozilla Firefox

Theorem

If $Y \leq X$ then $X \in P \Rightarrow Y \in P$

Contrapositive

If $Y \leq X$ then $Y \notin P \Rightarrow X \notin P$



$(6,4) \to (6,-4) \to X \to 2 —)$

Nov 24-5:16 PM

SAT $\leq$ 3-SAT $\leq$ CLIQUE

$\leq$ VC $\leq$ IND. SET

3-DM

$\leq$ Sched. w/ deadlines

Nov 24-5:21 PM

np.pdf (application/pdf Object) - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://www.columbia.edu/~cs2035/courses/csor4231.F09/np.pdf

11 / 14 100%

SAT

Levin

Theorem SAT is NP-complete

Proof idea: The turing machine program for any problem in NP can be verified by a polynomial sized SAT instance that encodes that the input is well formed and that each step follows legally from the next.

Implication We now have one NP-complete problem. We will now reduce other problems to it.

11.00 x 8.50 in

Nov 24-5:25 PM