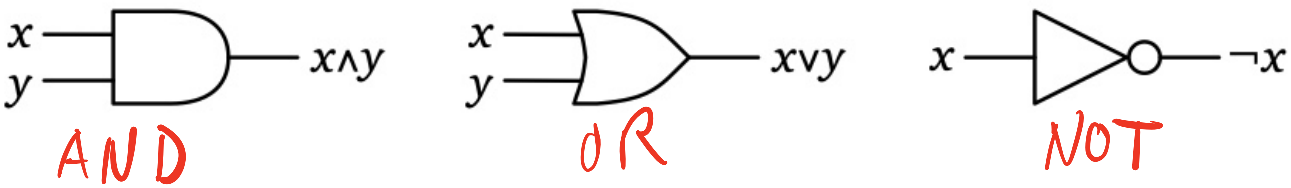
HW 10 due Tue Nov 18.

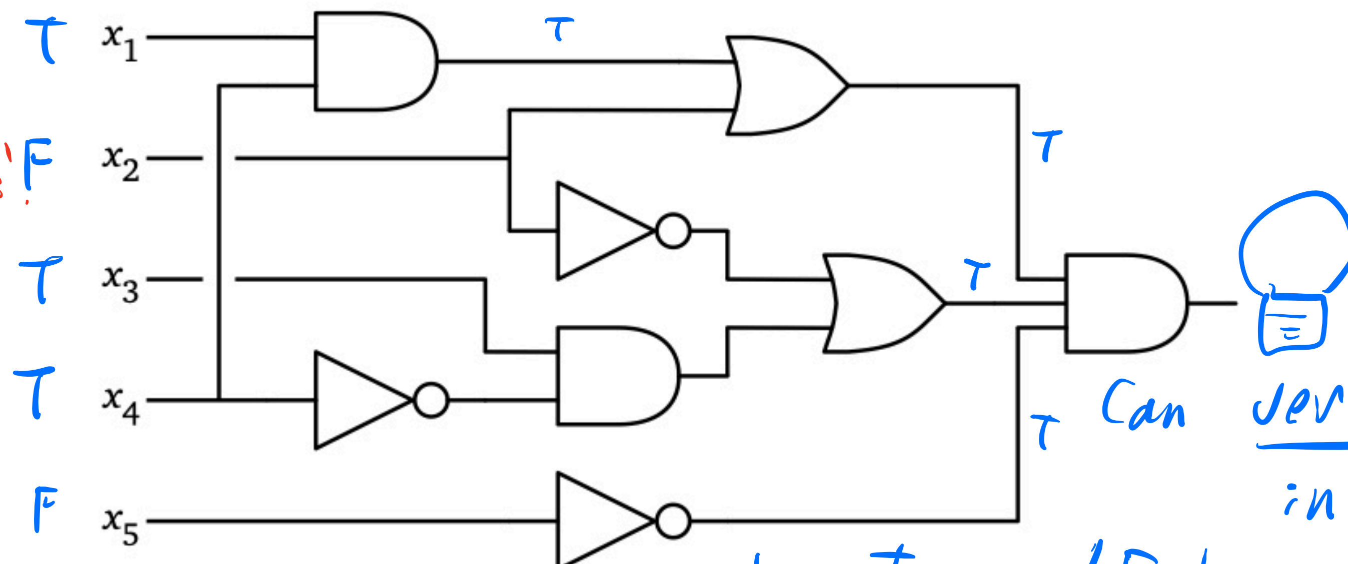
GPS 10 due <u>Wed Nov 19.</u> (do Zone 1 before HW 10!)

Given a Boolean circuit K.



Circuit SAT

yes, for this instance!



Can verify a solution in $O(n)$.

Can we set the variables to True/False so the light turns on?

<u>SAT</u>: Given a Boolean formula, can I set the variables

so the formula evaluates to True?

Could also nondeterministically try all settings and return True if <u>any</u> make the bulb turn on.

NP: Nondeterministic polynomial time

- <u>decision</u> problems (answer yes or not) you can solve nondeterministicly in poly time

-or- if the answer to an instance is yes, there is a "proof" or <u>certificate</u> you can verify in poly time.

Formaly: Decision problem A is in NP if

- there is a poly time procedure M called <u>the verifier</u> that takes

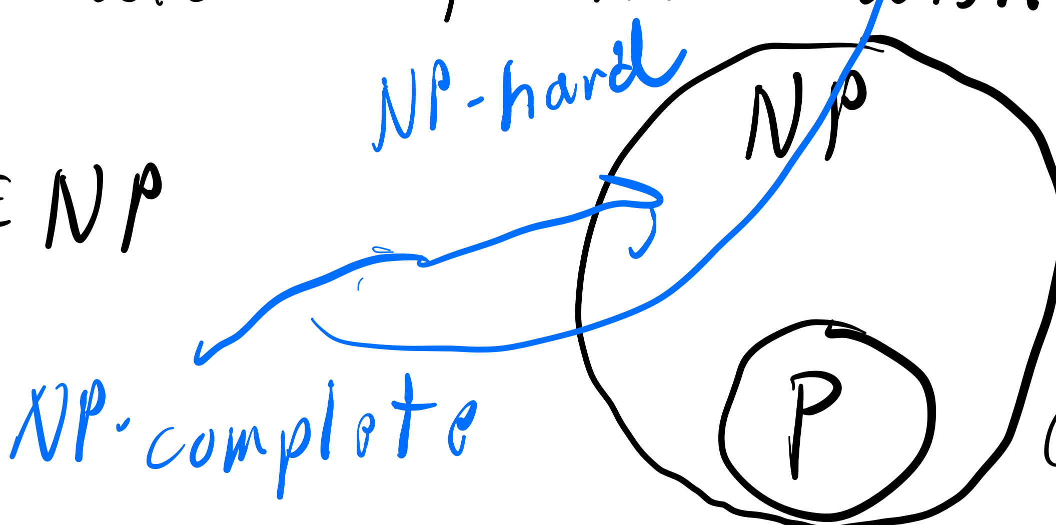
1) an input x of A
2) a certificate y s.t.

x is a yes instance of A iff there exist at least one poly size y s.t.
$M(x,y) = \text{True}$

e.g. y: setting of variables
y: a permutation of verts for HamCycle

P: deterministic polynomial time
- decision problems with poly time algs

$P \subseteq NP$



Question:
Does $P = NP$?

Clay Math Inst:

Consensus: $P \neq NP$.

$1,000,000 if you answer with good proof

Problem B is NP-hard if there is a poly time reduction from all $A \in NP$ to B.

strongly suggests no poly time alg for B, otherwise $P = NP$!

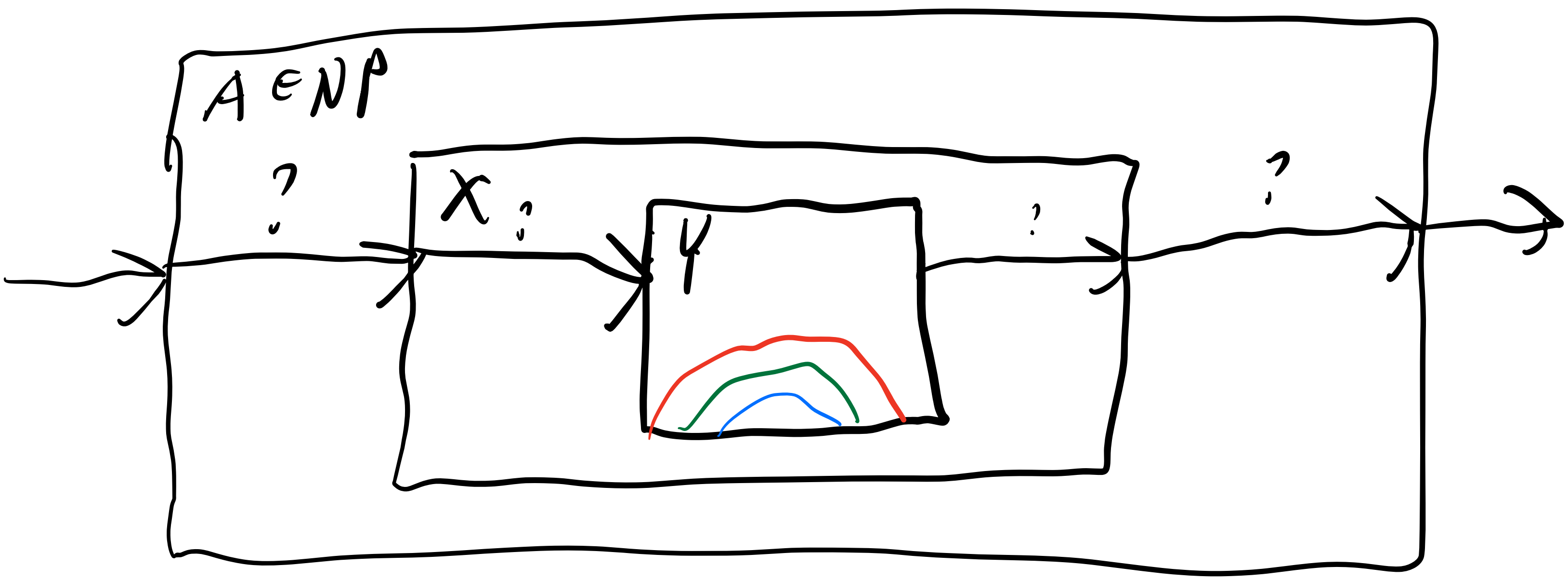
A is NP-complete if $A \in NP$ & A is NP-hard.

$\Rightarrow$ A has a poly time algorithm iff $P = NP$.

Cook-Levin: SAT is NP-complete
(also Circuit SAT)

To prove a problem Y is NP-hard, reduce a known NP-hard problem X to Y.

in poly time.



For a Boolean formula

variables : x, y

literals : $x, \bar{x}, y, \bar{y}$

<u>clause</u> : disjunction (or) of literals

$$(x \vee y \vee \bar{y})$$

<u>conjunctive normal form</u> (CNF) :

conjunction of clauses

$$(a \vee b \vee \bar{c}) \wedge (b \vee \bar{d} \vee e) \wedge \ldots$$

3CNF : each clause has <u>three</u> literals

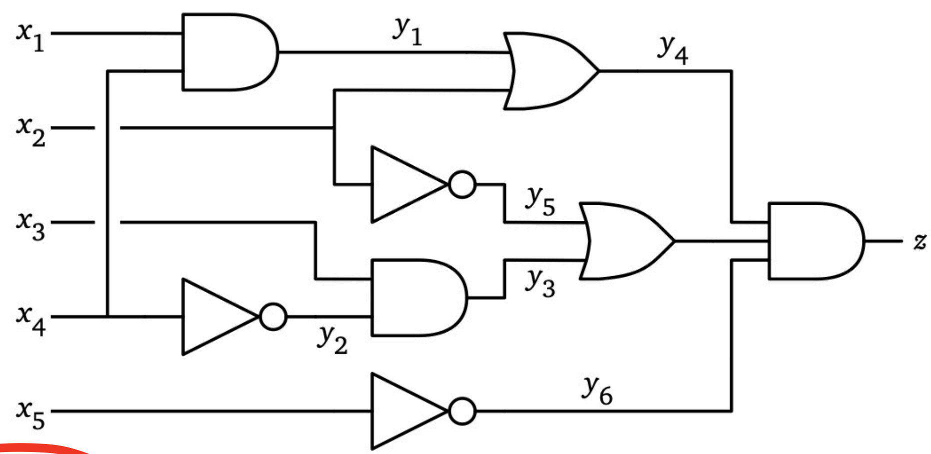
3SAT : Given a Boolean formula ϕ

in 3NF can we set the vars to
make it True?

Thm: 3SAT is NP-complete

- 3SAT in NP. ✓

Reduce (Circuit)SAT to 3SAT in poly time



O(n) time!

$$(y_1 = x_1 \wedge x_4) \wedge (y_2 = \overline{x_4}) \wedge (y_3 = x_3 \wedge y_2) \wedge (y_4 = y_1 \vee x_2) \wedge$$
$$(y_5 = \overline{x_2}) \wedge (y_6 = \overline{x_5}) \wedge (y_7 = y_3 \vee y_5) \wedge (z = y_4 \wedge y_7 \wedge y_6) \wedge z$$

can

$$(y_1 \vee \overline{x_1} \vee \overline{x_4}) \wedge (\overline{y_1} \vee x_1 \vee z_1) \wedge (\overline{y_1} \vee x_1 \vee \overline{z_1}) \wedge (\overline{y_1} \vee x_4 \vee z_2) \wedge (\overline{y_1} \vee x_4 \vee \overline{z_2})$$
$$\wedge (y_2 \vee x_4 \vee z_3) \wedge (y_2 \vee x_4 \vee \overline{z_3}) \wedge (\overline{y_2} \vee \overline{x_4} \vee z_4) \wedge (\overline{y_2} \vee \overline{x_4} \vee \overline{z_4})$$
$$\wedge (y_3 \vee \overline{x_3} \vee \overline{y_2}) \wedge (\overline{y_3} \vee x_3 \vee z_5) \wedge (\overline{y_3} \vee x_3 \vee \overline{z_5}) \wedge (\overline{y_3} \vee y_2 \vee z_6) \wedge (\overline{y_3} \vee y_2 \vee \overline{z_6})$$
$$\wedge (\overline{y_4} \vee y_1 \vee x_2) \wedge (y_4 \vee \overline{x_2} \vee z_7) \wedge (y_4 \vee \overline{x_2} \vee \overline{z_7}) \wedge (y_4 \vee \overline{y_1} \vee z_8) \wedge (y_4 \vee \overline{y_1} \vee \overline{z_8})$$
$$\wedge (y_5 \vee x_2 \vee z_9) \wedge (y_5 \vee x_2 \vee \overline{z_9}) \wedge (\overline{y_5} \vee \overline{x_2} \vee z_{10}) \wedge (\overline{y_5} \vee \overline{x_2} \vee \overline{z_{10}})$$
$$\wedge (y_6 \vee x_5 \vee z_{11}) \wedge (y_6 \vee x_5 \vee \overline{z_{11}}) \wedge (\overline{y_6} \vee \overline{x_5} \vee z_{12}) \wedge (\overline{y_6} \vee \overline{x_5} \vee \overline{z_{12}})$$
$$\wedge (\overline{y_7} \vee y_3 \vee y_5) \wedge (y_7 \vee \overline{y_3} \vee z_{13}) \wedge (y_7 \vee \overline{y_3} \vee \overline{z_{13}}) \wedge (y_7 \vee \overline{y_5} \vee z_{14}) \wedge (y_7 \vee \overline{y_5} \vee \overline{z_{14}})$$
$$\wedge (y_8 \vee \overline{y_4} \vee \overline{y_7}) \wedge (\overline{y_8} \vee y_4 \vee z_{15}) \wedge (\overline{y_8} \vee y_4 \vee \overline{z_{15}}) \wedge (\overline{y_8} \vee y_7 \vee z_{16}) \wedge (\overline{y_8} \vee y_7 \vee \overline{z_{16}})$$
$$\wedge (y_9 \vee \overline{y_8} \vee \overline{y_6}) \wedge (\overline{y_9} \vee y_8 \vee z_{17}) \wedge (\overline{y_9} \vee y_6 \vee z_{18}) \wedge (\overline{y_9} \vee y_6 \vee \overline{z_{18}}) \wedge (\overline{y_9} \vee y_8 \vee \overline{z_{17}})$$
$$\wedge (y_9 \vee z_{19} \vee z_{20}) \wedge (y_9 \vee \overline{z_{19}} \vee z_{20}) \wedge (y_9 \vee z_{19} \vee \overline{z_{20}}) \wedge (y_9 \vee \overline{z_{19}} \vee \overline{z_{20}})$$

Max Ind Set: Given undirected $G=(V,E)$.
Find max # of vertices not sharing an edge.

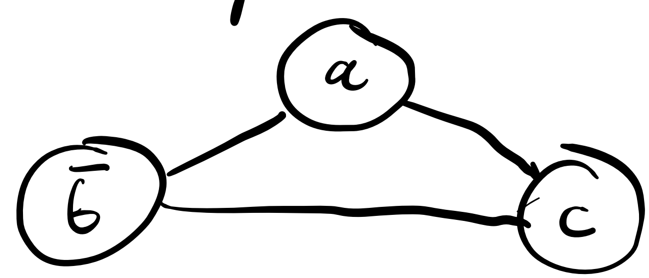
Thm: Max Ind Set is NP-hard.

Reduce from 3SAT to Max Ind Set.

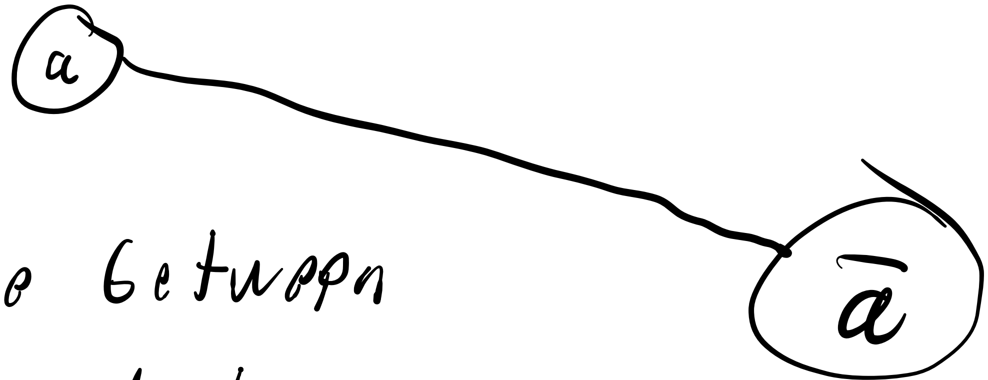
Let $\emptyset$ be a 3CNF Boolean formula.
$k \leftarrow$ # clauses.

Build graph $G=(V,E)$.
G has $3k$ vertices, one per literal.

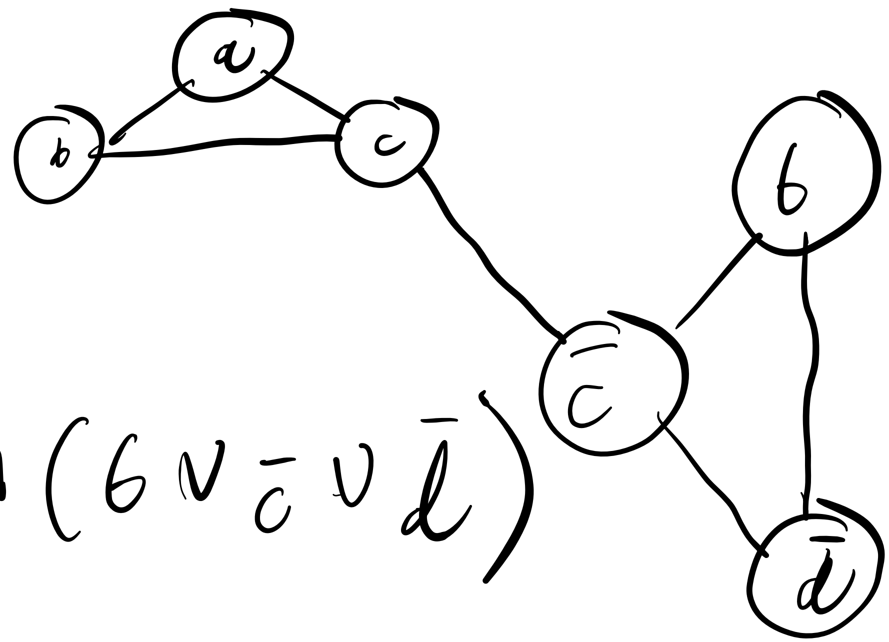
$(a \vee \bar{b} \vee c) \Rightarrow$



add edge between literals in same clause.

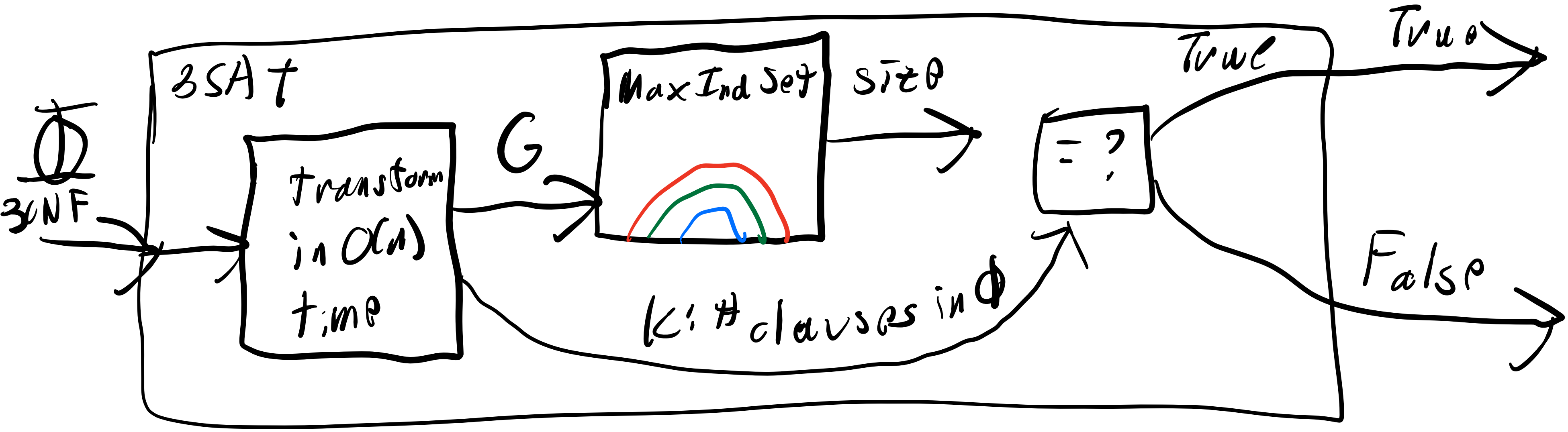


add edge between each literal + inverse



$(a \vee b \vee c) \wedge (b \vee \bar{c} \vee \bar{d})$

Run alg for Max Ind Set & return if size $\geq k$.



Need iff proof!