NP-Completeness

The Class P

P is the set of all decision problems for which there is a polynomial-time algorithm.

Is Ind-Set in P?

The Class NP

NP is the set of all decision problems with poly-time certifier.

Certifier: $B(s, t)$ poly-time algorithm

$s \in X$ for decision problem X iff there is a string t s.t. $B(s,t) = 1$ and $|t| = \text{poly}(|s|)$.

In practice: Can check solutions in poly time.

P vs NP

$P \subseteq NP$

Whether $P = NP$ or $P \neq NP$

is a million-dollar problem.

NP-Completeness

A decision problem X is NP-hard if there is a poly-time reduction from any problem in NP to X.

We say X is NP-complete if $X \in NP$ and it's NP-hard.

Boolean Satisfiability (SAT)

n boolean variables $X = x_1, \dots, x_n$, each in $\{0,1\}$

<u>clause</u> C to be a logical OR of some x_i or $\bar{x}_i$
($\bar{x}_i$ is the negation of x_i)

e.g. $x_1 \vee x_2$, $x_4 \vee \bar{x}_8 \vee x_{17}$

<u>boolean formula</u> is the logical AND of clauses.

eg. $(x_1 \vee x_2) \wedge (x_4 \vee \bar{x}_8 \vee x_{17})$

Boolean Satisfiability (SAT)

<u>boolean formula</u> is the logical AND of clauses.

e.g. $(x_1 \vee x_2) \wedge (x_4 \vee \bar{x}_8 \vee x_{17})$

A <u>truth assignment</u> is a function $v : X \to \{0, 1\}$
<u>satisfying</u> if all clauses evaluate to 1.

$x_1 = 1 \quad x_2 = 0 \quad x_4 = 0 \quad x_8 = 0 \quad x_{17} = 0$

$(1 \vee 0) \wedge (0 \vee \neg 0 \vee 0) = 1 \wedge 1 = 1.$

$(x_1) \wedge (\overline{x_1})$ - not satisfiable.

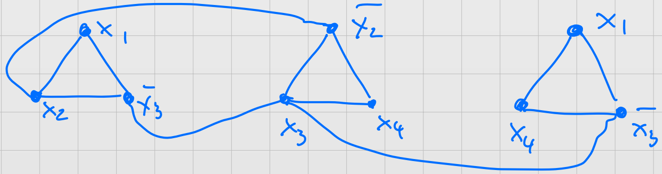
SAT is the decision problem of whether a given boolean formula has a satisfying assignment.

3SAT is SAT but each clause has 3 variables.

3SAT $\leq_p$ Ind-Set

$C_1, \ldots, C_k \rightarrow$ 3SAT Solver $\rightarrow 0/1$ $G, k \rightarrow$ Ind-Set Solver $\rightarrow 0/1$

$$(x_1 \vee x_2 \vee \bar{x}_3) \wedge (\bar{x}_2 \vee x_3 \vee x_4) \wedge (x_1 \vee x_4 \vee \bar{x}_3)$$



Circuit Satisfiability

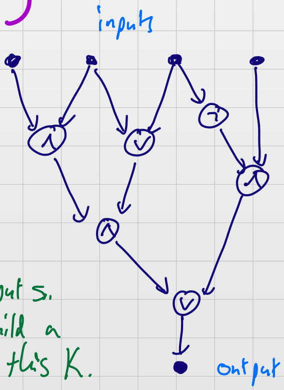
Boolean circuit K is a directed acyclic graph with

- Sources labelled either 1 or 0 (inputs)
- Other nodes labeled w/ $\{\wedge, \vee, \neg\}$
- Single node with not out-edges (output)

Circuit Satisfiability

Boolean circuit K is a directed acyclic graph with

- Sources labelled either 1 or 0 (inputs)
- Other nodes labeled w/ $\{\wedge, \vee, \neg\}$
- Single node with not out-edges (output)



Easy to show CircSAT $\in NP$.

Suppose $Y \in NP$. Want to solve for input s.
Take the certifier $B(s,t)$, and build a circuit for it, hard-coding s. Call this K.