COMPLEXITY AND ALGORITHMS

DEFINITION: DECISION PROBLEM:

INPUT

OUTPUT $\in \{0,1\}$

EXAMPLE: SAT

INPUT:

VARIABLES $x_i \in \{0,1\}$

literals $e_i^+ = x_i$, $e_i^- = \bar{x}_i$

clause $\bigvee_i e_i^+ \bigvee_i e_i^-$

OUTPUT:

1 IF ALL THE CLAUSE ARE SATISFIED

0 OTHERWISE

$x_1 \vee \bar{x}_2$

$x_2 \vee x_3$

$\bar{x}_1 \vee \bar{x}_3$

$(x_1 = 1 , x_2 = 1 , x_3 = 0)$ SATISFIES THE PROBLEM $\Longrightarrow$

OUTPUT $= 1$

DEFINITION: AN INSTANCE OF A PROBLEM IS A SPECIFIC PARAMETRIZATION OF THE PROBLEM

DEFINITION: THE COMPLEXITY OF AN ALGORITHM FOR A GIVEN PROBLEM WITH INPUT SIZE n IS

$O(t(n))$ IF $\exists k$: COMPLEXITY $\leq k\, t(n)$ FOR EVERY INSTANCE

$\Omega(t(n))$ IF $\exists k'$: COMPLEXITY $\geq k'\, t(n)$ FOR EVERY INSTANCE

$\Theta(t(n))$ IF BOTH

WHEN $t(n)$ IS A FUNCTION OF n

Observation: Given a problem (decision problem) we can analyze its complexity by considering all the possible algorithm for such a problem

Definition: A problem is in P (polynomial-time class) if there is a polynomial-time algorithm $(t(n) = poly(n))$

Definition: A problem is in NP (nondeterministic polynomial-time class) if there is a polynomial-time algorithm for verifying whether a solution satisfies the problem.

Example:

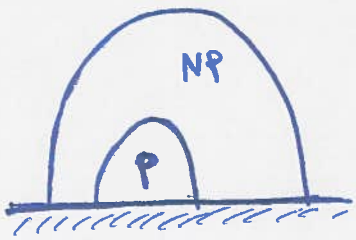
$x_1 \vee \overline{x_2}$

$x_2 \vee x_3$

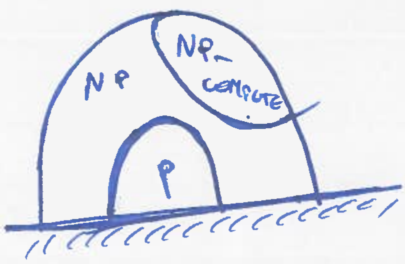
$\overline{x_1} \vee \overline{x_3}$

Given $(x_1 = 1, x_2 = 1, x_3 = 0)$ we can easily verify whether this solution satisfies the problem

Observation: $P \subseteq NP$



Definition: the hardest problem in NP are said NP-complete



Property: SAT is NP-complete

Definition: A problem in NP is NP-complete if SAT is reducible in polynomial time to such problem

PROPERTY: NP-COMPLETE PROBLEMS ARE COMMONLY CONSIDERED EXPONENTIAL-TIME PROBLEM $(t(n) = exp(n))$

EXAMPLE: 3SAT IS NP-COMPLETE

3SAT IS A SAT PROBLEM IN WHICH EACH CLAUSE IS COMPOSED OF EXACTLY 3 LITERALS

3SAT IS IN NP, BEING THE VERIFICATION PROBLEM EASY

REDUCTION FROM SAT: SAT $\leq_p$ 3SAT

EACH 3-CLAUSE IN SAT $\longrightarrow$ THE SAME IN 3SAT

EACH 1-CLAUSE IN SAT (e.g. x_i) $\longrightarrow$

$x_i \vee z \vee y$
$x_i \vee z \vee \bar{y}$
$x_i \vee \bar{z} \vee y$
$x_i \vee \bar{z} \vee \bar{y}$

TWO NEW VARIABLES FOR EACH 1-CLAUSE

EACH 2-CLAUSE IN SAT (e.g. $x_i \vee x_j$) $\longrightarrow$

$x_i \vee x_j \vee z$
$x_i \vee x_j \vee \bar{z}$

ONE NEW VARIABLE FOR EACH 2-CLAUSE

EACH K-CLAUSE IN SAT (e.g., $x_1 \vee x_2 \vee \dots x_k$) $\longrightarrow$

$x_1 \vee x_2 \vee z_1$
$\bar{z}_1 \vee x_3 \vee z_2$
$\bar{z}_2 \vee x_4 \vee z_3$
.....
$\bar{z}_{k-3} \vee x_{k-1} \vee x_k$

K-3 NEW VARIABLES FOR EACH K-CLAUSE

$$\left.\begin{array}{l} x_1 \vee \bar{x}_2 \vee x_3 \vee \bar{x}_4 \\ x_4 \vee \bar{x}_2 \\ x_1 \vee \bar{x}_3 \vee x_4 \vee x_5 \end{array}\right\} \longrightarrow \begin{array}{l} x_1 \vee \bar{x}_2 \vee z_1 \\ \bar{z}_1 \vee x_3 \vee \bar{x}_4 \\ x_4 \vee \bar{x}_2 \vee z_2 \\ x_4 \vee \bar{x}_2 \vee \bar{z}_2 \\ \\ x_1 \vee \bar{x}_3 \vee z_3 \\ \bar{z}_3 \vee x_4 \vee x_5 \end{array}$$

EXAMPLE: IND SET IS NP-COMPLETE

IND SET: INPUT: GRAPH G (UNDIRECTED, UNWEIGHTED)

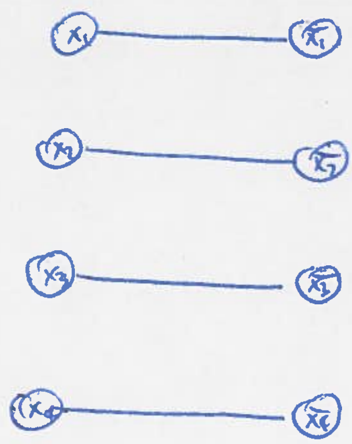
$K \in \mathbb{N}$

OUTPUT: 1 IF THERE ARE K VERTICES SUCH THAT NO PAIR OF VERTICES SHARE A LINK

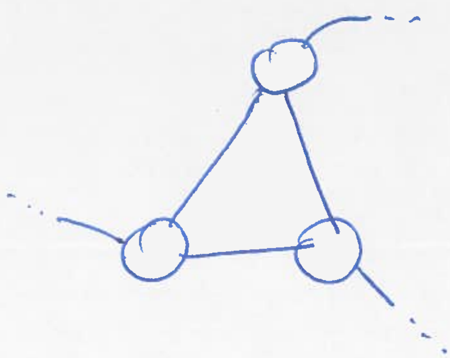
0 OTHERWISE

3SAT $\leq_p$ IND SET

VARIABLES IN 3SAT → TWO NODES IN INDSET SHARING AN EDGE



CLAUSE IN 3SAT → THREE NODES IN INDSET



CONNECTED TO THE THREE INVOLVED LITERALS

$x_1 \vee \bar{x}_2 \vee x_3$

⇓