NP-completeness

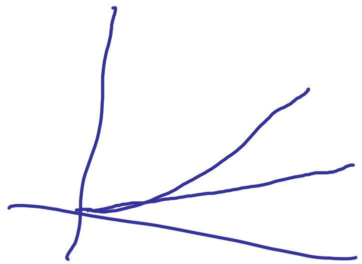
Computational problems : Fast "efficient" algorithms

Efficient : polynomial-time i.e. $O(n^c)$ time for inputs of length n c: constant, independent of n

n^{10} vs. 2^n impractical for even $n \approx 100$

Polytime time : - composition remains polytime

- mostly independent of input representation



Polynomial time	Not known & "unlikely" to have poly time algorithms
- Sorting	- vertex cover
- Graph traversal	- independent set
- Binary search	- longest path (simple)
- Shortest path	
- Min Spanning tree	

Optimization problems: Out of all possible solutions, one of min/max value.

<u>Decision problems</u>: Yes/No answer

- Is there a path of length $\leq k$ from s to t in graph G?
- Is there a vertex cover of size $\leq k$ in a given graph G?

- Longest path: Is there a path of length $\geq k$ from s to t in G

Class P: Problems which have polynomial time algorithms

NP-complete: Problems for which we believe polynomial time algorithms are unlikely

NP: Problems for which "yes" answer is easily verifiable through a simple certificate

<u>Halting problem</u>: Can we write an algorithm A which takes a program P & an x as input and determines if program P stops on i/p x or runs forever?

- undecidable

NP-complete problems: "Hardest" problems in NP.

$P \subseteq NP$

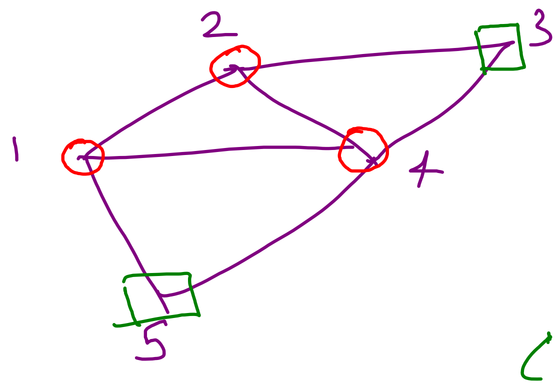
even an empty certificate works for "yes" & "no" answers.

NP- complete

Fact: A polytime ~~f~~ algorithm for one NP- complete problem
$\implies$ a polytime algorithm for ALL NP-complete problems

Every problem in NP can be "transformed" or "reduced" to any NP- complete problem in polynomial time.

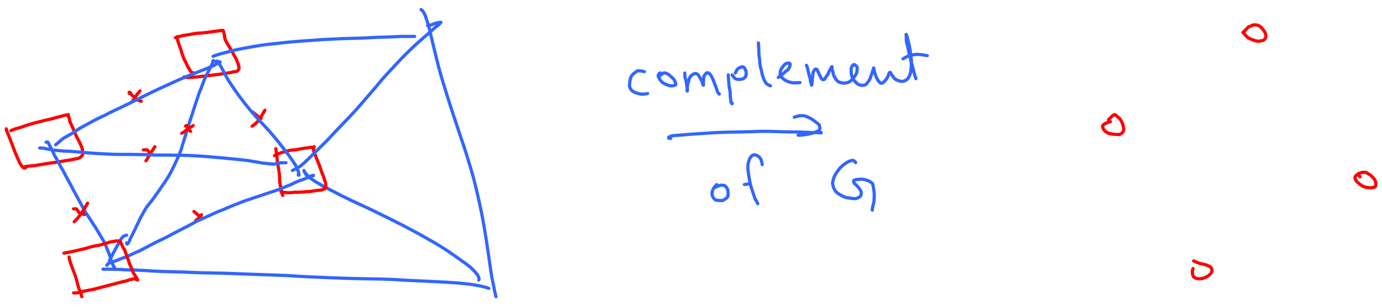
Vertex cover: Given a graph $G=(V,E)$, is there a set of vertices $S \subseteq V$ s.t. every edge has at least one end point in S, $|S| \leq K$?



Independent set: Given $G=(V,E)$ is there $S \subseteq V$ s.t. no two vertices in S have an edge between them. Is $|S| \geq k$?

Claim: A graph G with n vertices has a vertex cover of size $\leq k$ if and only if G has an independent set of size $\geq n-k$

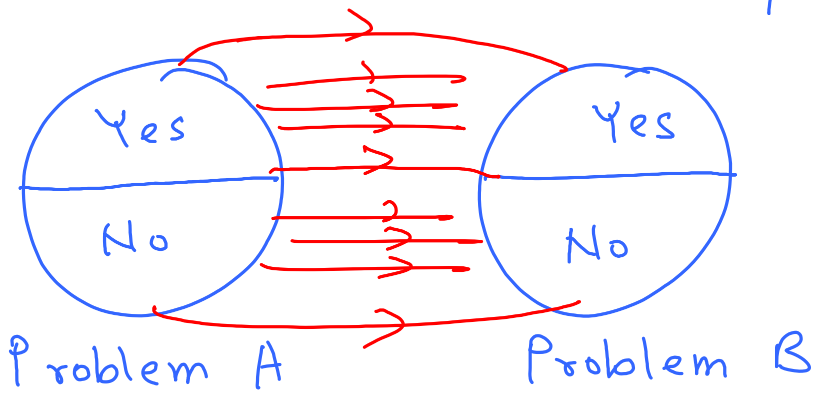
Clique problem: Given: Graph $G=(V,E)$, is there $S \subseteq V$ s.t. every pair of vertices in S has an edge between them, and $|S| \geq k$?



G

Vertex cover, clique, independent set can all be reduced to each other.

Reduction between decision problems:



The reduction must be polytime computable, "yes" instances get converted to "yes" instances, "no" instances get converted to "no" instances

How to show a Problem X to be NP-complete?

- Take any NP-complete problem A
- Show that Problem A reduces to problem X in polytime.

$\Rightarrow$ Any problem in NP reduces to problem X in polytime.

The first NP-complete problem: Satisfiability /SAT

Boolean formula : $(x_1 \vee x_2 \vee x_3) \underline{\underline{\wedge}} (\overline{x_1} \vee x_2 \vee \overline{x_3})$

$x_1, x_2, x_3 \in \{0, 1\}$

$\vee$: OR

$0 \vee 0 = 0$
$0 \vee 1 = 1$
$1 \vee 0 = 1$
$1 \vee 1 = 1$

AND $\wedge$

$0 \wedge 0 = 0$
$0 \wedge 1 = 0$
$1 \wedge 0 = 0$
$1 \wedge 1 = 1$

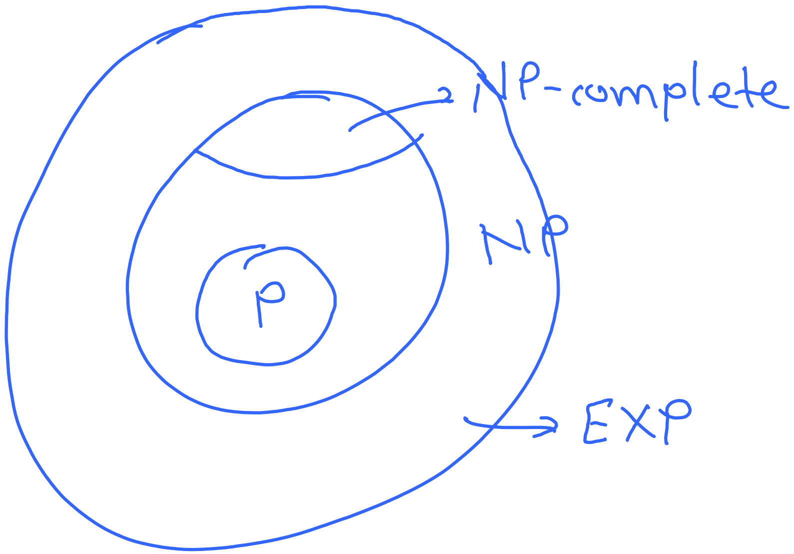
$\overline{x_1}$: Not x_1

$\overline{0} = 1$

$\overline{1} = 0$

SAT: Is there any assignment to variables s.t. the formula evaluates to 1?

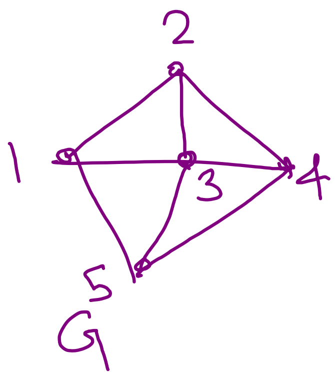
Cook-Levin theorem: SAT is NP-complete i.e. every problem in NP reduces to SAT in polytime.

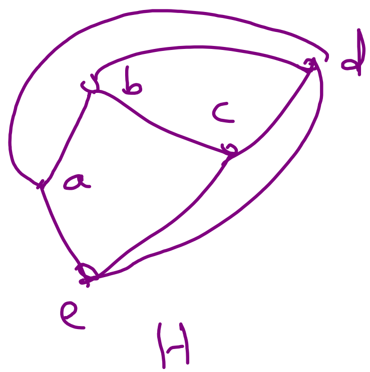


Is $P = NP$? - We don't know!

$P \neq EXP$

Graph isomorphism : (GI)





We don't know whether $GI \in P$.

Unlikely to be NP-complete.