1-) Write a polynomial-time reduction from the 3SAT problem to the independent set problem.

3SAT $\leq_p$ Independent Set

Input: Given a 3CNF formula φ

Goal: Construct a graph G_φ and number k such that G_φ has an independent set of size k iff φ is satisfiable.

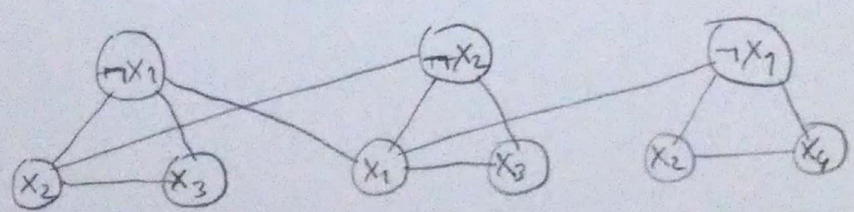
The reduction:

① G_φ will have one vertex for each literal in a clause.

② Connect the 3 literals in a clause to form a triangle; the independent set will pick at most one vertex from each clause, which will correspond to the literal to be set to true.

③ Connect 2 vertices if they label complementary literals; this ensures that the literals corresponding to the independent set do not have a conflict.

④ Take k to be the number of clauses.



$$\varphi = (\neg x_1 \vee x_2 \vee x_3) \wedge (\neg x_2 \vee x_1 \vee x_3) \wedge (\neg x_1 \vee x_2 \vee x_4)$$

2-) NP-complete problems are the hardest problems in NP. A decision problem L is NP complete if:

① L is in NP.

② Every problem is in NP is reducible to L in polynomial time.

3-) Cook-Levin Theorem: SAT is NP-complete. This means every problem $L \in NP$ can reduced to SAT in polynomial time. SAT is the hardest problem in NP, since we can solve any problem with only polynomial time overhead if we have an algorithm for SAT.

4-) Give three examples to NP-complete problems and define each of them.

1. Independent Set
2. Hamiltonian Cycle
3. Vertex Cover

Independent Set Problem: A given undirected graph G, is there an independent set of size $\geq k$?

Hamiltonian Cycle Problem: Given a directed graph G, with n vertices, does G have a Hamiltonian cycle? (Hamiltonian cycle is a cycle in the graph that visits every vertex in G exactly once.)

Vertex Cover Problem: Given an undirected graph, is there a vertex cover of size $\leq k$?

5-) If you were to prove a problem X is NP-complete, give a proof idea by writing the two steps that one needs to prove.

If X is NP-complete,

1. Since we believe $P \neq NP$,
2. and solving X implies $P = NP$.

X is unlikely to be efficiently solvable.

6-) NP-hard problems are at least as hard as NP-complete problems. NP-hard problems do not have to be in NP, and they do not have to be decision probs.

For any $Y \in NP$, we have that $Y \leq_p X$.