15-251: Great Theoretical Ideas In Computer Science

Recitation 9 : P and NP

New Phrases

- We say a language is in **P** if there exists a polynomial time algorithm that decides the language
- We say a problem is in **NP** if there exists a polynomial time verifier TM V such that for all $x \in \Sigma^*$, x is in L if and only if there exists a polynomial length certificate u such that $V(x, u) = 1$.
- We say there is a **polynomial-time many-one reduction** from A to B if there is a **polynomial-time** computable function $f : \Sigma^* \to \Sigma^*$ such that $x \in A$ if and only if $f(x) \in B$. We write this as $A \leq_m^P B$. (We also refer to these reductions as Karp reductions.)
- A problem Y is **NP-hard** if for every problem $X \in$ **NP**, $X \leq_m^P Y$.
- A problem is **NP-complete** if it is both in **NP** and **NP-hard**.

Not oPen

Show that **NP** is closed under union and intersection. Specifically, prove that if two languages, $L_1, L_2 \in$ **NP**, then $L_1 \cap L_2 \in$ **NP** and $L_1 \cup L_2 \in$ **NP**

No Peeking

We define a vertex covering of a graph as a set of vertices such that each edge in the graph is incident to at least one vertex in the set.

VERTEX-COVER: $\{\langle G, k\rangle : G$ is a graph, k a natural number, G contain a vertex covering of size $k\}$

Show VERTEX-COVER is **NP-Complete** (Try reducing from 3SAT).

(Extra) No Privacy

DOUBLE-CLIQUE: Given a graph $G = (V, E)$ and a natural number k, does G contain two vertex-disjoint cliques of size k each?

Show DOUBLE-CLIQUE is NP-Complete.

(Bonus) Never Pausing

Prove that the Halting Problem is **NP-hard**.