Polynomial-Time Reductions

The independent set problem:

Given an undirected graph $G = (V, E)$, a set $S \subseteq V$ of vertices is independent if no two vertices in $S$ are joined by an edge in $E$.

The independent set problem is to determine, given a graph $G$ and integer $K$, if $G$ has an independent set of size at least $K$. Note that this is a decision version of an optimization problem that asks for the largest independent set.
\{3, 4, 5\} is an independent set.
\{1, 6, 4, 5\} is a bigger one.

The independent set problem is one about which we don't know.

We don't know if the independent set problem can be solved in polynomial time. That is, we don't
Know if there is a poly-time algorithm that solves the problem.

The Vertex Cover problem: Given a graph $G = (V,E)$, a vertex cover is a subset $S \subseteq V$ such that each edge in $E$ has at least one of its endpoints in $S$.

In the example graph, $\{1, 2, 6, 7\}$ is a vertex cover. So is $\{2, 3, 7\}$.

The vertex cover problem asks, given a graph $G$ and integer $k'$, if there is a vertex cover of size at most $k'$. (A decision version of a minimization problem.)
Well, we don't know if there is a polynomial algorithm for vertex cover either.

But we do know that if there is a polynomial algo for vertex cover, there is one for independent set as well. And vice versa.

This is based on the simple observation: $S \subseteq V$ is an independent set in $G=(V, E)$ if and only if $V-S$ is a vertex cover.

So $G=(V, E)$ has an independent set of size at least $k$ iff $G$ has a vertex cover of size at most $|V|-k$.
Solving Independent set given a black box for vertex cover:

Suppose we want to know if $G$ has an independent set of size $\geq k$. We ask the black box if $G$ has a vertex cover of size at most $10k$. If black box says yes, we return yes. If it says no, we return no.

Similarly, we can solve vertex cover given a black box for independent set.

Note that these algorithms are polynomial-time, but they only contain calls to the black boxes.
Notice that if there is a polynomial-time algo for vertex cover, then replacing the black box for vertex cover with this algo yields a polynomial time algo for independent set.

**Def.** Problem $Y$ is polynomial-time reducible to problem $X$, denoted $Y \leq_p X$, if there is an algo that solves $Y$ using a polynomial number of Computational steps, plus a polynomial no. of calls to a black box for solving $X$.

So $\text{Independent-set} \leq_p \text{Vertex-Cover}$

and $\text{Vertex-Cover} \leq_p \text{Independent-Set}$. 
A consequence of the defn:

1. Suppose $Y \leq_p X$. If $X$ can be solved in polynomial time, then $Y$ can be solved in polynomial time as well.

2. Suppose $Y \leq_p X$. If $Y$ can't be solved in poly-time, then neither can $X$.

(2) is the direction in which we will exploit poly-time reducibility: we'll infer the hardness of a problem by reducing to it a known problem that's believed to be hard.

Let's do a more interesting example of a poly-time reduction...