Recap from last lecture:

Reductions: Problems A & B
(for example A = MST & B = TSP)

A \rightarrow B means reduce A to B.
Can solve A using B.

Given a poly-time alg. for B, we can use it as a black-box to solve A in poly-time.

Search problem:

Given input I & solution S, can verify that S is a solution to I in time Poly(|I|).
$NP = \text{all search problems.}$

Thus to show $A \in NP$, need to give an alg. to verify solutions to $A$.

$P = \text{search problems that can be solved in poly-time.}$

Clearly, $P \subseteq NP$.

Is $P = NP$?

If $P \neq NP$ then some problems in $NP$ (i.e., some search problems) cannot be solved in poly-time.

- which ones?

$NP$-complete problems = hardest problems in $NP$. 
Colorings is NP-complete means:

a) Colorings \in \text{NP}

b) if P \neq \text{NP} then we cannot solve colorings in poly-time

(b) is equivalent to (take contrapositive)
if colorings can be solved in poly-time then every \text{AeNP} can be solved in poly-time.

In other words,
for all \text{AeNP}, A \rightarrow \text{Colorings}.

How do we show that?

Suppose we know: SAT is NP-complete.
Thus, for all \text{AeNP}, A \rightarrow \text{SAT}.
So if we show SAT \rightarrow \text{Colorings}
then: A \rightarrow \text{SAT} \rightarrow \text{Colorings}
So: A \rightarrow \text{Colorings}.
To show Colorings is NP-complete we need to show:

a) Colorings ∈ NP
b) for a known NP-complete problem $B$, $B \Rightarrow$ Colorings.

Let's assume SAT is NP-complete. Next class we'll show 3-SAT is NP-complete but let's use it for now.

**3-SAT:**

**input:** Boolean formula $f$ in CNF with $n$ variables $x_1, \ldots, x_n$ and $m$ clauses $C_1, \ldots, C_m$ where every clause has $\leq 3$ literals.

**output:** Satisfying assignment if one exists NO if none exist.
Independent Set:

For a graph \( G = (V, E) \), a subset \( SCV \) is an independent set if no edges contained in \( S \):

this means for all \( (x, y) \in E \):

\( x \notin S \) & \( y \notin S \) (so doesn't include both endpoints)

Example:

\[ S = \{a, e\} \] is an independent set, so is \[ \emptyset \]
\[ S = \{a, f, i\} \] is an independent set of max size
Independent Set Optimization Problem:

- **Input:** Undirected graph \( G = (V, E) \)
- **Output:** Independent set \( S \) of maximum size

But given \( S \) how to verify its max size?

So add parameter \( g \) as goal.

Independent Set Problem:

- **Input:** Undirected graph \( G = (V, E) \) & goal \( g \)
- **Output:** Independent set \( S \) of size \( |S| \geq g \) if one exists.
  
  **NO** if none exist.
Theorem: Independent Set is NP-complete.

Proof:

a) First we'll show: Independent Set ∈ NP.

Given G, g, & S, in O(n+m) time we can go through all edges & for (x, y) ∈ E check that x ∈ S &/or y ∈ S.

b) We'll show: 3SAT → Independent Set.

Given input f for 3SAT we need to create input (G, g) for independent set. Let's delay & look at some simpler reductions first.
We'll assume SAT is NP-complete.
Next class, we'll use that to prove 3SAT is NP-complete.
Later today, we'll show Independent Set is NP-complete.
Now let's use IS to get some more graph problems as NP-complete.

Clique: fully connected subgraph: all pairs of vertices have an edge between them.

For \( G=(V,E) \), SCV is a clique if
for all \( x, y \) yes then \( (x,y) \in E \)
Want to find the largest clique.

Clique Problem:
\[
\text{input: } G=(V,E) \& \text{goal } g. \\
\text{output: } SCV \text{ where } S \text{ is a clique } \& |S| \geq g \\
\text{or NO if no such } S \text{ exists.}
\]
Theorem: Clique is NP-complete.

Proof:

a) Clique in NP:

Given a graph G, G in O(n^3) time can check for every pair x, y ∈ S that (x, y) ∈ E, & in O(n) time that |S| ≥ q.

b) Independent Set ⇒ Clique.

For G = (V, E), let G = (V, E̅) where:

E̅ = \{ (x, y) : (x, y) \notin E \}

(\text{so if } (x, y) \in E \Rightarrow (x, y) \notin E̅

\text{and if } (x, y) \notin E \Rightarrow (x, y) \in E̅)

Given input (G, q) for independent set

Run Clique on (G̅, q).
Claim: $S$ is a solution for Independent Set on $(G, g) \iff S$ is a solution for Clique on $(\overline{G}, \bar{g})$.

Proof:

$\Rightarrow$ if $S$ is an independent set in $G$ then for all $x, y \in S$ we know $(x, y) \in E$ and hence $(x, y) \in \overline{E}$.
Therefore, $S$ is a clique in $\overline{G}$.

$\Leftarrow$ if $S$ is a clique in $G$ then for all $x, y \in S$ we know $(x, y) \in \overline{E}$ and hence $(x, y) \in E$.
Therefore, $S$ is an independent set in $G$.

$\Box$
**Vertex cover (VC):**

For $G=(V,E)$,

$S \subseteq V$ is a vertex cover if it covers every edge:

for every edge $(x,y) \in E$,

either $x \in S$ \&/or $y \in S$

(So one or both endpoints are in the VC)

**Example:**

\[
\begin{array}{c}
\text{a} \\
\text{b} \\
\text{c} \\
\text{d} \\
\text{e} \\
\text{f}
\end{array}
\]

$S=\{a,b,d,e,f\}$ is a vertex cover

$S=\{b,d,f\}$ is a vertex cover of min size.

Want to find smallest vertex cover.
Vertex Cover Problem:

**Input:** $G = (V,E)$ & goal/budget $b$

**Output:** Vertex cover $S$ of size $\leq b$ if one exists

NO otherwise.

**Theorem:** Vertex Cover is NP-complete.

**Proof:**

a) $VC \in NP$:

Given $G, b \& S$ in $O(n+m)$ time we can check for every edge $(x,y) \in E$ that $x \in S \&/\text{or} y \in S$.

b) Independent Set $\rightarrow$ Vertex Cover.

Given input $(G,g)$ for IS,

run Vertex Cover on $(G,n-g)$

For solution $S$ to VC, output $S$ as solution to IS

& if NO solution to VC, then NO solution to IS.
Claim: $S$ is a vertex cover of size $|S| \leq b$ where $b = n - g$ if and only if $S$ is an independent set of size $|S| \geq g$.

Proof:

\(\Rightarrow\) Consider VS in $G$.

Hence, for every $(x, y) \in E$, $x \in S$ or $y \in S$, and thus $x \notin S$ and/or $y \notin S$.

Therefore, $S$ is an independent set.

And if $|S| \leq n - g$ then $|S| \geq g$.

\(\Leftarrow\) Consider independent set $S$ in $G$.

Hence, for every $(x, y) \in E$, $x \in S$ or $y \in S$.

Thus, $x \in S$ and/or $y \in S$.

And $S$ is a VC.

And if $|S| \geq g$ then $|S| \leq n - g$. 

\(\Box\)