Announcement:
HW 11 (keep 21 of 33 scores)

Last Time:
Prove undecidability

What about problems that are decidable but have no efficient algms?

^ polynomial-time

Similar Plan:
- Start with one hard problem
- Prove other problems hard by ^ reduction

Def $P =$ class of all langs/decision problems that can be solved in poly.time

Main Def Given decision problems $L_1, L_2$, a poly-time reduction from $L_1$ to $L_2$ is an algm $f$ s.t. $\forall$ input $x$,

output of $L_1$ on $x$ is yes $\iff$ output of $L_2$ on $f(x)$ is yes

Notation Write $L_1 \leq_P L_2$. 
**Fact 1**
If \( L_1 \leq_{p} L_2 \) and \( L_2 \in P \), then \( L_1 \in P \).

**Pf Sketch:**
\[
\begin{array}{c}
X \\
\xrightarrow{f} f(x) \\
\xrightarrow{\text{solver for } L_2} \text{yes or no}
\end{array}
\]

*(note: composition of two polynomials is polynomial)*

**Contrapositive:** If \( L_1 \leq_{p} L_2 \) and \( L_1 \notin P \), then \( L_2 \notin P \).

**Fact 2**
If \( L_1 \leq_{p} L_2 \) and \( L_2 \leq_{p} L_3 \), then \( L_1 \leq_{p} L_3 \).

**Pf:** By composition. \( \square \)

**Examples of Reduction**

**Example 1**
**Vertex Cover:** *(decision vers.)*

- **Input:** undirected graph \( G = (V, E) \), integer \( k \)
- **Output:** yes iff \( \exists \) vertex cover of size \( \leq k \)
  iff \( \exists \) subset \( S \subseteq V \) of size \( \leq k \)
  s.t. \( \forall u \in V \Rightarrow u \notin S \) or \( u \in S \)

![Graph Example](image)
\[ \text{vc: } \{A, C, E\} \quad k=3 \]

**Set-Cover:**

Input: set \( U, A_1, \ldots, A_n \subseteq U \), integer \( k \)

Output: yes iff \( \exists I \subseteq \{1, \ldots, n\} \),

\[ \text{st. } \bigcup_{i \in I} A_i = U. \]

**Example:** \( U = \{1, 2, 3, 4, 5\} \)

\[ \{1, 3, 5\}, \{2, 5\}, \{1, 2, 3\}, \{3, 4\} \quad k=3 \]

**Vertex-Cover \( \leq_p \) Set-Cover:**

**Proof:** Given input to Vertex-Cover: \( G = (V, E), k \),

Construct input to Set-Cover: \( U, A_1, \ldots, A_n, k' \)

where \( U = E \), \( k' = k \)

\[ A_i = \{ e \in E \mid e \text{ is incident to } v_i \} \]

This construction takes poly time.

**Correctness:**

\[ \exists \text{ vertex cover } S \text{ of size } \leq k \text{ in } G \]

\[ \iff \exists I \subseteq \{1, \ldots, n\} \text{ of size } \leq k \]

\[ \text{st. } \bigcup_{i \in I} A_i = E. \]

**Ex. 2** Independent Set: (decision vers)

Input: undir graph \( G = (V, E) \), integer \( k \)

Output: yes iff \( \exists \text{ indep set } S \text{ of size } \geq k \)

\[ S \subseteq V \text{ st. } \]
Indep Set $\leq_p$ Vertex Cover:

**Problem:** Given input to Indep Set: $G=(V,E)$, $k$

**Construct input to Vertex Cover:** $G'$, $k'$

where $G' = G$

$k' = n - k$

**Correctness:**

$\exists$ Indep set $S$ of size $\geq k$

$\implies$ $\exists$ vertex cover $S'$ of size $\leq n-k$.

$S' = V - S$

Vertex-Cover $\leq_p$ Indep-Set:

Same

---

How to get first hard problem??

**Def:** $NP =$ class of all decision problems of the form

Input: $x$

Output: yes iff

$\exists y$ s.t. $C(x,y)$ is true

where $y$ has poly size

$C$ runs in poly time

$C$ certifies/verifier
Ex: Vertex-Cover ∈ NP:
Certificate: subset $S \subseteq V$ ∈ poly size
Certificate: check $|S| \leq k$
& $\forall u \in V \in E \Rightarrow u \in S$ or $v \notin S$
∈ poly time

Set-Cover ∈ NP:
Indep. Set ∈ NP.

Rank: NP stands for Nondeterministic Polytime

Idea: the "hardest" problem in NP??