1. Prove that $x$ satisfies $L$ if and only if $y$ satisfies $L$.

2. Describe an algorithm that computes a function $f$ mapping every instance of $L$ to an instance of $L'$.

3. Select a known $NP$-complete problem $L'$.

4. Prove $L \in NP$.

- Method for proving that $L$ is $NP$-complete:

  If $L \in NP$, then $L \in NP$.

  If $L \in NP$, then $L \in NP$.

  If $L$ is a problem such that $L \leq_p L'$ for some $L'$ in $NP$, then $L \in NP$.

Lemma 3.4.8
5. Prove that the algorithm computing \( f \) runs in polynomial time.

Example 3: Prove that SAT is NP-complete using Circuit-SAT to do reduction.
\[(
\begin{array}{c}
L \lor b \lor g \lor 8 \lor x = o^1x \\
L \land g \lor b \lor x = g^0x \\
9 \land 5 \land 3 \lor x = 4 \lor x \\
5 \lor 3 \lor x = 4 \lor x \\
7 \lor 5 \lor c \lor o \lor x = 4 \lor x \\
(x \lor y \lor z \lor 0 \lor x = 4 \lor x) \\
5 \lor 3 \lor x = 4 \lor x \\
\end{array}
)\]

\[\text{(circuit-sat)} = \text{O(reduction)} + \text{O(sat)}\]
Proof: satisfiability of boolean formula is NP-complete.

To prove \( \text{SAT} \in \mathsf{NP} \), we show \( \text{Circuit-SAT} \leq_p \text{SAT} \). We need to reduce any instance:

To prove \( \text{SAT} \in \mathsf{NP} \), we show that a certifying instance consisting of a satisfying assignment for an instance of \( \text{Circuit-SAT} \leq_p \text{SAT} \).
Given a circuit C, the algorithm produces \( \phi \) in polynomial time. Given C, let E be the set of \( \mathcal{V}(C) \) and \( \mathcal{E}(C) \) be the set of \( \mathcal{E}(C) \) for \( \mathcal{E}(C) \). Let \( V \subseteq \mathcal{V}(C) \) be a vertex cover of \( \mathcal{V}(C) \). If \( \mathcal{V}(C) \), then \( \mathcal{V}(C) \) or \( \mathcal{V}(C) \) (or both).

A vertex cover of an undirected graph \( G=(\mathcal{V},\mathcal{E}) \) is a subset \( V \subseteq \mathcal{V}(C) \) such that each edge of \( \mathcal{E}(C) \) is incident to a vertex in \( V \).

Example 4: Vertex cover problem.

Complete the proof.
The vertex cover problem is to find a vertex cover of minimum size in a given graph. The size of a vertex cover is the number of vertices in it.
③ Prove that the vertex cover is NP-hard. Clause NP-vertex cover

\[ \text{\textbf{Claim:}} \quad \Delta(G) \] \( \leq 4 \) 

\[ 6 \leq \Delta(G) \] 

\[ G = (V, E) \text{ is a graph.} \]

\[ k = \left| V \right| \] 

\[ k' = \left| E \right| \]

\[ (G, k) \text{ is a yes instance to CLIQUE} \]

```
Proof: Show that vertex cover is NP-hard.
```
2. Conversely, suppose \( C \) has a vertex cover \( \mathcal{V}_C \) where \( |\mathcal{V}_C| = k \). Then claim that \( T - V \) is a chain in \( G \).

1. Suppose that \( G \) has a chain, \( T \), of size \( |T| = k \). We claim that \( T \subseteq C \).

Graph \( G \) has a vertex cover of size \( |V| - k \). Prove that if graph \( G \) has a chain of size \( |T| \) if and only if the \( \mathcal{V}_C \) contains \( \mathcal{V}_C \) \( \subseteq \mathcal{C} \) \n \n1 \n\nIf \( C \), \( |V| - k \), \( \mathcal{V}_C \), \( \mathcal{C} \), the vertex cover problem. The reduction algorithm takes as input an instance \( G \), \( k \), of the chain problem.