Consider the problem \textsc{false-sat}:

Given a boolean expression \( E \) that is false when all its variables are made false, is there some other truth assignment that makes \( E \) false, besides all-false?

Decide whether the problem is in \( \text{NP} \) or \( \text{coNP} \).

Describe its complement.

If the problem or its complement is \( \text{NP-complete} \), prove it.

Proof:

The problem is \( \text{NP-complete} \).

- In \( \text{NP} \): given a boolean expression \( E \), we need to check:
  1) that \( E \) is false when all variables are assigned false
  2) that there is some other truth assignment making \( E \) false

  (1) can be done in poly-time by a DTM
  (2) can be done in poly-time by a NTM

  guess a truth assignment \( T \) different from all false, and answer yes if \( T \) and \( E \) evaluate to false

- \( \text{NP-hard} \): by a reduction from \( \text{SAT} \)

  Let \( E \) be a boolean expression with variables \( x_1, \ldots, x_n \).
  We construct an expression \( E' \) s.t.: \( E \in \text{SAT} \) iff \( E' \in \text{FALSE-SAT} \)

  1) Test if \( E \) is true when all variables are false (polynomial)
    If so, \( E \in \text{SAT} \), and we convert it to a fixed expression
      that is in \text{FALSE-SAT}, e.g. \( x \lor \neg y \).
2) Otherwise, let $E'$ be $\neg E \land (x_1 \lor \ldots \lor \lor x_n)$. Clearly, the reduction is poly-time.

We have that $E'$ is false when all of $x_1, \ldots, x_n$ are false. Notice that in case (2), $E$ is false when all variables are false. Hence, if $E \in SAT$, then it is satisfied by a truth assignment $T$ different from all-true.

Thus, $\neg E$ is made false by $T$, and $E' \in FALSE-SAT$.

Conversely, if $E' \in FALSE-SAT$, then since $x_1, \ldots, x_n$ is false only for the all-true truth assignment, there must be some other truth-assignment $T$ that makes $\neg E$ false. Then $T$ makes $E$ true, and $E \in SAT$. 
Exercises on problems in P, NP, and NP-complete

Exercise

Consider the following optimization version of SAT:

\text{MAXSAT}: \text{ Input: a propositional formula } F \text{ in CNF, and an integer } k
\text{ Output: yes, if there is a truth assignment that satisfies at least } k \text{ clauses of } F
\text{ no, otherwise}

What is the complexity of MAXSAT?

a) MAXSAT is \text{ NP } immediate, by the following \text{ NP algorithm}
\text{ 1) guess a truth assignment } \pi \text{ (non-deterministic polynomial)}
\text{ 2) count the } \# \text{ of clauses satisfied by } \pi \text{, and answer yes if it is } \geq k \text{ (deterministic polynomial)}

b) MAXSAT is \text{ NP-hard}

This follows from the fact that CSAT is a special case of MAXSAT.

Formally, we can polynomially reduce CSAT to MAXSAT, i.e.

\text{ SAT } \leq \text{ poly MAXSAT }

Given an instance } F \text{ of CSAT, we construct an instance } (F, k) \text{ of MAXSAT, where } k \text{ is the } \# \text{ of clauses of } F.

Obviously, } k \text{ can be obtained in polynomial time from } F, \text{ and }

F \in \text{ CSAT } \iff (F, k) \in \text{ MAXSAT} \quad \Box
Consider the following problems:

1) Vertex-cover (VC)

Given an undirected graph $G = (V, E)$ and an integer $k \geq 2$, is there a subset $C$ of $V$ with $|C| \leq k$ such that $C$ covers all edges of $G$ (i.e., for each edge $\{v_i, v_j\} \in E$ with $v_i \neq v_j$, $\{v_i, v_j\} \cap C \neq \emptyset$).

2) Independent-set (IS)

Given an undirected graph $G = (V, E)$ and an integer $k \geq 2$, is there a subset $C$ of $V$ with $|C| \geq k$ such that for all $v_i, v_j \in C$ with $v_i \neq v_j$, $\{v_i, v_j\} \notin E$.

3) Clique

Given an undirected graph $G = (V, E)$ and an integer $k \geq 2$, is there a subset $C$ of $V$ with $|C| \geq k$ such that for all $v_i, v_j \in C$ with $v_i \neq v_j$, $\{v_i, v_j\} \in E$.

Show that VC, IS, and Clique can be reduced to each other in polynomial time.

N.B. In the definitions of VC, IS, and Clique we have ignored self-loops (since we required $v_i \neq v_j$).
IS $\leq$ poly CLIQUE

Given an instance $(G, k)$ of IS, we construct an instance $(G', k')$ of CLIQUE as follows:

$k' = k$

Set $G' = (V, E)$.

Then $G' = (V, E')$, where $E' = V \times V \setminus E$ (i.e., the edges of $E'$ are obtained by connecting all pairs of nodes that are not connected in $E$.)

E.g., $G$

\[\begin{array}{ccc}
1 & \rightarrow & 2 \\
\uparrow & & \downarrow \\
2 & \rightarrow & 5
\end{array}\]

IS

\[\begin{array}{ccc}
3 & \rightarrow & 4 \\
\uparrow & & \downarrow \\
5 & \rightarrow & 4
\end{array}\]

CLIQUE

The reduction works because the maximum independent set of $G$ is precisely the maximum clique in the complement graph of $G$.

VC $\leq$ poly IS

Given an instance $(G, k)$ of VC, we construct an instance $(G', k')$ of IS as follows:

$k' = |V| - k$

$G' = G$

The reduction works because the vertices in a vertex-cover $C$ cover all edges of $G$. Hence the set $V \setminus C$ must have no edges between its elements, and is thus an independent set.

E.g.,

\[\begin{array}{ccc}
1 & \rightarrow & 2 \\
\uparrow & & \downarrow \\
2 & \rightarrow & 3 \\
\uparrow & & \downarrow \\
3 & \rightarrow & 4
\end{array}\]

The marked nodes $\{2, 3, 5\}$ are a VC of $G$.

Hence, there cannot be an edge $\{1, 4\}$.
Given an instance \((G, k)\) of Clique, we construct an instance \((G', k')\) of VC as follows:

\[ k' = |V| - k \]

Let \(G = (V, E)\).

Then \(G' = (V, E')\), where \(E' = V \times V \setminus E\).