Barani: Reduction from 3-SAT to CSAT
(see textbook 10.3.4)

Given a CNF formula \( E = C_1 \land C_2 \ldots \land C_n \)
with each \( C_i = \bigvee_{j=1}^{k_i} l_{j,i} \),
we construct a 3-CNF formula \( F \) as follows.

For each clause \( C_i \) of \( E \)

1) if \( C_i = (l) \) (i.e., a single literal)
   introduce two new variables \( u, v \), and replace
   \( C_i \) by 4 clauses
   \[(l + u + v), \]
   \[(l + u + \overline{v}), \]
   \[(l + \overline{u} + v), \]
   \[(l + \overline{u} + \overline{v}) \]
   Since \( u, v \) appear in all 4 combinations, the 4 clauses can be satisfied only if \( l \) is true.

2) if \( C_i = (l_1 + l_2) \)
   introduce a new variable \( \overline{z} \), and replace
   \( C_i \) by 2 clauses
   \[(l_1 + l_2 + \overline{z}), \]
   \[(l_1 + l_2 + \overline{\overline{z}}) \]
   \( \therefore \overline{z} \) is 1

3) if \( C_i = (l_1 + l_2 + l_3) \), just leave it

4) if \( C_i = (l_1 + l_2 + \ldots + l_m) \) with \( m \geq 4 \)
   introduce \( y_3, y_4, \ldots, y_{m-3} \) and replace \( C_i \) by
   \[(l_1 + l_2 + y_3), \]
   \[(l_3 + \overline{y_3} + y_4), \]
   \[ \ldots \]
   \[+ (l_{m-2} + \overline{y_{m-4}} + y_{m-3}), \]
   \[(l_{m-1} + l_m + \overline{y_{m-3}}) \]
An assignment $T$ satisfying $E$ makes at least one literal of $C_i$ true. Let it be $l_j$.

Then, by making $j_{i-1}, j_{i-2}$ true and $j_{i-1}, \ldots, j_{m-3}$ false,

we satisfy all clauses replacing $C_i$. Thus we can extend $T$ to satisfy $F$.

Conversely, if $T$ makes all $l_j$ of $C_i$ false, then not all new clauses can be satisfied.

Why? Each $j_k$ can make at most 1 clause true, but there are $m-2$ clauses and $m-3$ $j_k$'s.

The 3-CNF formula $F$ is linear in $E$ and can be constructed in linear time.

We get: $\text{CSAT} \not\leq_{poly} \text{3-SAT}$

$\Rightarrow$ from $\text{CSAT} \not\in \text{NP}$, we get $\text{3-SAT} \not\in \text{NP}$

We also know $\text{3-SAT} \in \text{NP}$ (since SAT $\in \text{NP}$)

$\Rightarrow$ 3-SAT is NP-complete.
Exercise (10.3.2): The problem $4TA - SAT$ is defined as follows:

Given a propositional formula $E$, does $E$ have at least 4 satisfying truth assignments?

Show that $4TA - SAT$ is in NP-complete.

Proof:

1) $4TA - SAT$ is in NP

We devise a non-deterministic poly-time algorithm:

1) give 4 truth-assignments $T_1, T_2, T_3, T_4$
2) check that $T_1, T_2, T_3, T_4$ all satisfy $E$

Note that both steps require time polynomial in the size of $E$

2) $4TA - SAT$ is NP-hard

We show this by reducing SAT to $4TA - SAT$.

Let $E$ be a propositional formula, and let $x_1, ..., x_n$ be all variables in $E$.

We construct a new formula $E'$ as:

$$E \in SAT \iff E' \in 4TA - SAT$$

Let $y_1, y_2$ be two new variables. Then

$$E' = E \lor ((x_1 \land \neg x_2 \land ... \land x_n) \land (y_1 \land \neg y_2) \lor (\neg y_1 \land y_2))$$
Consider the truth assignments for $x_1, \ldots, x_n, y_1, y_2$.

<table>
<thead>
<tr>
<th>$x_1 \cdot x_2 \cdot x_n$</th>
<th>$y_1 \cdot y_2$</th>
<th>Case 1 $E \notin SAT$</th>
<th>Case 2 $E \in SAT$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T \cdots T$</td>
<td>$T \ T$</td>
<td>$F \ T$</td>
<td>$? \ T$</td>
</tr>
<tr>
<td>$T \cdots T$</td>
<td>$T \ F$</td>
<td>$F \ T$</td>
<td>$? \ T$</td>
</tr>
<tr>
<td>$T \cdots T$</td>
<td>$F \ T$</td>
<td>$F \ T$</td>
<td>$? \ T$</td>
</tr>
<tr>
<td>$T \cdots T$</td>
<td>$F \ F$</td>
<td>$F \ F$</td>
<td>$? \ T$</td>
</tr>
</tbody>
</table>

$\vdash \exists 3 \geq 4$

Alternative solution:

$E' = E \land (y_1 \lor y_2 \lor y_3)$

- If $E$ is unsatisfiable, then $E'$ is unsatisfiable, and hence $E' \notin SAT$

- If $E$ is satisfiable, then $E'$ has at least 7 satisfying truth assignments; these are obtained by combining:
  - a TA for $x_1, \ldots, x_n$ satisfying $E$ with
  - the 7 TAs for $y_1, y_2, y_3$ satisfying $y_1 \lor y_2 \lor y_3$