NP-Complete \( (B) \)

1. \( B \in \text{NP} \)

2. Every \( A \in \text{NP} \) is poly. time reducible to \( B \) \( (A \leq_p B) \) \( (\text{NP-Hard}) \)
SAT is NP-Complete

∀A ∈ NP, A ≤_p SAT

Suppose A is in NP
A is decided by a non-det poly-time TM, D

ϕ is satisfiable iff
w is accepted by D.
If $B$ is NPC and $B \leq_p C$, for $C \in \text{NP}$, then $C$ is NPC.
3-SAT is NP complete.

a. Is 3-SAT NP?

NP algorithm
- non-deterministically generate all assignments to all variables
- evaluate the formula for the assignments
- if evaluation is true, accept
  else reject
b 3-SAT is NP-Hard
SAT ≤p 3-SAT

1. Convert φ to CNF
2. Convert φ so that all terms have exactly 3 literals

<table>
<thead>
<tr>
<th>terms in φ</th>
<th>terms in φ'</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 literals</td>
<td>copy</td>
</tr>
<tr>
<td>1 literal</td>
<td>(x)</td>
</tr>
<tr>
<td>2 literals</td>
<td>(x ∨ y)</td>
</tr>
<tr>
<td>4 (x₁ ∨ x₂ ∨ x₃ ∨ x₄)</td>
<td>(x₁ ∨ x₂ ∨ a)</td>
</tr>
<tr>
<td></td>
<td>( \land (\overline{a} ∨ x₃ ∨ x₄) )</td>
</tr>
<tr>
<td>5 (x₁ ∨ x₂ ∨ x₃ ∨ x₄ ∨ x₅)</td>
<td>(x₁ ∨ x₂ ∨ a)</td>
</tr>
<tr>
<td></td>
<td>( \land (\overline{a} ∨ x₃ ∨ b) )</td>
</tr>
<tr>
<td></td>
<td>( \land (b ∨ x₄ ∨ x₅) )</td>
</tr>
</tbody>
</table>

- 3-SAT is NPC
- CLIQUE is NPC

3-SAT ≤p CLIQUE
CLIQUE is in NP.