Find the errors in the proofs of the following two wrong claims.

1. Wrong claim: The following problem is NP-complete.
   - Input: A CNF $\phi$.
   - Question: Does $\phi$ have exactly one truth assignment that satisfies it?

   **Proof** The problem is obviously in NP. We take a truth assignment as a certificate and verify in polynomial time whether it is the only solution that satisfies $\phi$. If it is the unique solution then, we output YES, and otherwise we output NO. Note that if $\phi$ is a YES input, then such a truth assignment always exists, but if it is NO input, every truth assignment will be rejected by our certifier.

   To prove completeness we reduce SAT to this problem. Consider an input $\phi$ to SAT with variables $x_1, \ldots, x_n$. We add a new variable $y$, and construct a new formula $\psi$ as in the following. We add the term $y$ to each clause, and moreover for each variable $x_i$ we create a new clause ($y \lor x_i$). For example if $\phi = (x_1 \lor x_2) \land (x_1 \lor \neg x_2)$ then $\psi = (x_1 \lor x_2 \lor y) \land (x_1 \lor \neg x_2 \lor y) \land (\neg y \lor x_1) \land (\neg y \lor x_2) \land (\neg y \lor x_3)$. Note that if $\phi$ is not satisfiable, then the only way to satisfy $\psi$ is to assign $y = True$, which (with the newly added clauses) will force $x_1 = \ldots = x_n = True$. On the other hand if $\phi$ is satisfiable, then in addition to this solution, we can also take the truth assignment that satisfies $\phi$ and extend it by setting $y = False$. We conclude that $\phi$ is satisfiable if and only if there is NOT a unique solution that satisfies $\psi$. Hence we can ask an oracle for the above problem whether $\psi$ has a unique solution, and if it says YES we say $\phi$ is NOT satisfiable, and if it says YES we say that $\phi$ is satisfiable. This oracle algorithm solves SAT efficiently.

   1 The error is here. There is no way to check if is the only solution? The rest of the proof is correct and it shows SAT $\leq_p$ UniqueSAT.

2. Wrong claim: The following problem is NP-complete.
   - Input: A 4-regular (i.e. all the vertices have degree 4) graph $G$.
   - Question: Is there a subset $S \subseteq V(G)$ that is both an independent set and a vertex cover.

   **Proof:** Given a set $S$ we can check in polynomial time to see whether $S$ is both an independent set and a vertex cover. If it is we accept the certificate and otherwise we do not. Note that this is an efficient certifier: For YES inputs, there is always an acceptable $S$, and for NO inputs, no $S$ will be accepted.

   To prove completeness, we reduce Independent Set problem to our problem\(^2\). Consider a 4-regular graph $G$ on $n$ vertices. Note that $G$ has exactly $2n$ edges. Moreover note that if $S$ is both an independent set and a vertex cover, then every edge is covered with exactly one vertex (we can’t pick both endpoints in $S$ because of the independence property), and on the other hand every vertex covers exactly 4 edges (since the graph is 4-regular), and hence $|S| = 2n/4 = n/2$.

   Note also that if $S$ is an independent set of size $2n/4$, then it has to be a vertex cover, because every vertex in $S$ will cover exactly 4 edges and those are not covered by any other vertex (since $S$ is an independent set), and thus all the $2n = 4 \times |S|$ edges are going to be covered. We showed $G$ has an independent vertex cover $\iff G$ has an independent set of size $n/2$.

   Hence we can ask the oracle of Independent Set Problem to see if $G$ has an independent set $S$ of size $k$ for $k = n/2$. If it returns YES then we know that $G$ has an independent vertex cover and if it says No, then we know that $G$ does not have an independent vertex cover.

---

\(^2\)The rest of the proof gives a correct reduction from Independent VC to Independent Set Problem (which is the opposite of what we wanted to prove).