1. The problem SET-COVER asks, given a collection $S_1, \ldots, S_n$ of sets whose union is U, and a natural number k, does there exist a subcollection of at most k of the sets whose union is U?

Instance: Sets $S_1, \ldots, S_n$, and $k \in \mathbb{N}$
Question: Does there exist $C \subseteq \{1, \ldots, n\}$ s.t. $|C| \leq k$ and $\bigcup_{i \in C} S_i = \bigcup_{i=1}^{n} S_i$?

Prove that SET-COVER is **NP**-complete.

Solution: First, show that SET-COVER $\in$ **NP**. The non-deterministic algorithm guesses $C \subseteq \{1, \ldots, n\}$ of size k and for each element u checks that $u \in \bigcup_{i \in C} S_i$. If the test succeed for every $u \in \bigcup_{i=1}^{n} S_i$ then the algorithm accepts; otherwise it rejects.

Now, to prove that SET-COVER is **NP**-hard, show that VERTEX-COVER $\leq_p$ SET-COVER. The idea is, given an instance of VERTEX-COVER, to think of each vertex as a set that covers the edges of which it is an endpoint. Finding k such sets whose union contains all the edges is then equivalent to finding a vertex cover of size k. Formally, the reduction maps $\langle G = (V, E), k \rangle$ to $\langle S_1, \ldots, S_n, k \rangle$, where $n = |V|$ and $S_i = \{e \in E \mid v_i \in e\}$ is the set of edges that contain v_i.

Claim 1. $\langle G, k \rangle \in$ VERTEX-COVER $\iff$ $\langle S_1, \ldots, S_n, k \rangle \in$ SET-COVER

Proof. ($\Rightarrow$) Assume that $\langle G, k \rangle \in$ VERTEX-COVER, and let $C \subseteq V$ be a vertex cover of size k or less. Define $C' = \{i \mid v_i \in C\}$. Then C' is a set cover for $S_1, \ldots, S_n$, since $\bigcup_{i=1}^{n} S_i = E$: if $e \in E$, then e has an endpoint $v_i \in C$, and therefore $i \in C'$, which implies that $e \in \bigcup_{i \in C'} S_i$. Also $|C'| = |C| \leq k$.

($\Leftarrow$) Assume that $\langle S_1, \ldots, S_n, k \rangle \in$ SET-COVER. Then there exists $C' \subseteq \{1, \ldots, n\}$ such that $|C'| \leq k$ and $\bigcup_{i \in C'} S_i = E$. Define $C = \{v_i \mid i \in C'\}$. Then $|C| = |C'| \leq k$, and C is a vertex cover for G, as every edge $e \in E$ is contained in a set S_i with $i \in C'$, implying that $v_i \in C$ covers e. ☐

The reduction can be computed in polynomial time: the algorithm simply goes through the vertices of G one by one, and for each vertex v_i makes a list S_i of the edges of which v_i is an endpoint. There are $|V|$ iterations, and each takes time $O(|V|^2)$, so the total time is $O(|V|^3)$.

2. An independent set in a graph $G = (V, E)$ is a subset $I \subseteq V$ of vertices such that there are no edges between vertices in I, i.e. for all $u, v \in I$, $\{u, v\} \notin E$. The INDEPENDENT-SET decision problem is defined below.

Instance: A graph $G = (V, E)$, and $k \in \mathbb{N}$
Question: Does G have an independent set of size k or larger?

Prove that INDEPENDENT-SET is **NP**-complete.

Solution: First, we need to prove that INDEPENDENT-SET $\in$ **NP**. The non-deterministic machine guesses a subset $I \subseteq V$ of size k, and verifies that there is no edge between any pair of vertices in I. This takes polynomial time.

To prove that INDEPENDENT-SET is **NP**-hard, we'll show VERTEX-COVER $\leq_p$ INDEPENDENT-SET. The key observation is that if C is a vertex cover, then $V \setminus C$ is an independent set, and if I is an independent set, then $V \setminus I$ is a vertex cover. So G contains a vertex cover of size k iff G contains an independent set of size $n - k$, where n is the number of vertices in G. The reduction is therefore $f(\langle G, k\rangle) = \langle G, n - k\rangle$.

To see that the relationship claimed above between vertex covers and independent sets is true, first assume C is a vertex cover. Then every edge has at least one endpoint in C, no edge has both endpoints in $V \setminus C$, and therefore $V \setminus C$ is an independent set. On the other hand, if I is an independent set, then no edge has both endpoints in I, so every edge has at least one endpoint in $V \setminus I$, so $V \setminus I$ is a vertex cover.

3. Prove that CLIQUE is **NP**-complete.

Solution: We already proved CLIQUE $\in$ **NP** in lecture: the non-deterministic machine guesses a set of k vertices and checks that it is a clique.

To prove that CLIQUE is **NP**-hard, show INDEPENDENT-SET $\leq_p$ CLIQUE. For a graph $G = (V, E)$, let $\overline{G} = (V, \overline{E})$ denote the graph that contains an edge $\{u, v\}$ iff $\{u, v\} \notin E$. The key observation is that every independent set of G is a clique in $\overline{G}$, and vice-versa, so G has an independent set of size k iff $\overline{G}$ has a clique of size k. The reduction simply maps $\langle G, k\rangle$ to $\langle \overline{G}, k\rangle$, which can easily be done in polynomial time.