Exercise 1. Show that $\text{coNP} \subseteq \text{EXP}$.

Exercise 2. Consider the following problem Reverse-3SAT:

**Instance:** A propositional formula $\varphi$ in 3CNF—that is, a formula of the form $\varphi = c_1 \land \cdots \land c_m$, where each $c_j$ is of the form $c_j = \ell_{j,1} \lor \ell_{j,2} \lor \ell_{j,3}$, where $\ell_{j,1}, \ell_{j,2}, \ell_{j,3}$ are propositional literals.

**Question:** Is there a truth assignment $\alpha$ to the variables occurring in $\varphi$ that sets at least one literal in each clause $c_j$ to false?

Prove that Reverse-3SAT is NP-complete—that is, prove that it is in NP and that it is NP-hard. To show NP-hardness, you may give a reduction from any known NP-complete problem.

- **Hint:** reduce from 3SAT.

Exercise 3. Consider the following problem CLIQUE:

**Instance:** An undirected graph $G = (V,E)$, and a positive integer $k \in \mathbb{N}$.

**Question:** Does $G$ contain a clique of size $k$—that is, is there a set $C \subseteq V$ of vertices with $|C| = k$ such that for each $v,v' \in C$ with $v \neq v'$ it holds that $\{v,v\}' \in E$?

In this exercise, we will show that CLIQUE is NP-complete.

(i) Prove that CLIQUE is in NP.

To show that CLIQUE is NP-hard, we will give a polynomial-time reduction $f$ from 3SAT to CLIQUE. We describe this reduction $f$ as follows: for an arbitrary instance $\varphi$ of 3SAT, we describe what the instance $f(\varphi) = (G,k)$ looks like.

Let $\varphi = c_1 \land \cdots \land c_m$ be an arbitrary 3CNF formula, containing propositional variables $x_1, \ldots, x_n$, where $c_j = \ell_{j,1} \lor \ell_{j,2} \lor \ell_{j,3}$ for each $1 \leq j \leq m$. Then we construct the graph $f(\varphi)$ as follows.

- We introduce vertices $v_{j,1}, v_{j,2}, v_{j,3}$, for each $1 \leq j \leq m$. That is, for each clause $c_j$ we add three vertices—one for each literal occurring in the clause.

- Two vertices $v_{j,l}$ and $v_{j',l'}$ are connected with an edge if and only if $j \neq j'$ and the literals $\ell_{j,l}$ and $\ell_{j',l'}$ are not each other’s negation.

Finally, we set $k = m$.

Let $\varphi_{ex} = (x_1 \lor \overline{x}_2 \lor x_3) \land (x_1 \lor \overline{x}_2 \lor \overline{x}_3) \land (x_2 \lor x_3 \lor \overline{x}_1)$ be an example 3CNF formula.

(ii) Let $f(\varphi_{ex}) = (G_{ex},k_{ex})$. Compute $k_{ex}$ and draw the graph $G_{ex}$.

(iii) Show that $\varphi_{ex}$ is satisfiable. Use a satisfying assignment for $\varphi_{ex}$ to produce a clique of size $k_{ex}$ for $G_{ex}$.

(iv) Prove, for an arbitrary 3CNF formula $\varphi$, that $\varphi$ is satisfiable if and only if $f(\varphi) = (G,k) \in \text{CLIQUE}$.

(v) Explain why the function $f$ is polynomial-time computable.
Exercise 4 (reduction from HamCycle to HamPath). Consider the following problem HamPath:

**Instance:** An undirected graph $G = (V, E)$, and two vertices $s, t \in V$.

**Question:** Is there a Hamiltonian path in $G$ from $s$ to $t$—in other words, a path from $s$ to $t$ that visits each vertex exactly once?

Consider also the following problem HamCycle:

**Instance:** An undirected graph $G = (V, E)$.

**Question:** Is there a Hamiltonian cycle in $G$—in other words, a cycle that visits each vertex exactly once?

Give a polynomial-time reduction from HamCycle to HamPath.

Exercise 5 (self-reducibility of 3SAT). Suppose that you have a polynomial-time algorithm $A$ for (the decision problem) 3SAT. Show that you can use $A$ to construct a polynomial-time algorithm $B$ that, when given as input a 3CNF formula $\varphi$, outputs a satisfying assignment $\alpha$ for $\varphi$ if such an assignment exists, and that outputs 0 otherwise.