In the following text, (almost) every word is terminology. If anything is unclear, please ask! You are not expected to know most of them.

1. Recall the following problems.
   3SAT
   **Input:** Boolean formula $\varphi$ on variables $x_1, \dots, x_n$ in conjunctive normal form (CNF) where each clause has length at most 3.
   **Output:** Is there an assignment to $x_1, \dots, x_n$ which makes $\varphi$ satisfiable?
   Clique
   **Input:** Graph $G$ and an integer $k$.
   **Output:** Does $G$ contain a subgraph $H$ isomorphic to $K_k$, a complete graph on $k$ vertices?[1]
   Independent Set (IS)
   **Input:** Graph $G$ and an integer $k$.
   **Output:** Does $G$ contain a subgraph $H$ isomorphic to $\overline{K_k}$, the complement to the complete graph on $k$ vertices?
   Vertex Cover (VC)
   **Input:** Graph $G = (V, E)$ and an integer $k$.
   **Output:** Is there a subset of vertices $W \subseteq V$ of size $|W| \leq k$ such that every edge has at least one endpoint in $W$? Equivalently, is $G \setminus W$ an edgeless graph?
   Dominating Set (DS)
   **Input:** Graph $G = (V, E)$ and an integer $k$.
   **Output:** Is there a subset of vertices $W \subseteq V$ of size $|W| \leq k$ such that every vertex is either in $W$ or at least one of its neighbours is in $W$?
   3-Coloring
   **Input:** Graph $G = (V, E)$.
   **Output:** Is there a mapping $f : V \to \{1, 2, 3\}$ such that for every edge $uv \in E$ we have $f(u) \neq f(v)$?

2. Show the following polynomial reductions.

   - From 3SAT to Clique.
   - From Clique to Independent Set.
   - From any of the problems mentioned so far to Vertex Cover.

   Note that if we assume that 3SAT is NP-*hard*, then we have shown NP-*hardness* of all problems in this task. Can you also show that they are NP-*complete*?

   For extra exercise, you can show NP-completeness of the remaining problems in the preceding task. These are classical results that can be found in most complexity textbooks.

3. Suppose you have a 3SAT solver running in time $T$ whose only output is whether the input formula is satisfiable or not. Can you use it to find a satisfying assignment to input variables in time $T \cdot \text{poly}(T)$?

4*. Maximum Independent Set has a trivial algorithm running in time $\mathcal{O}^*(2^n)$. The goal of this task is to design an algorithm which runs in time $\mathcal{O}^*(c^n)$ where $c < 2$.

   I will give you an algorithm on the other side of this sheet, and your task is only to analyze its correctness and running time. But if you want to invent your own algorithm, feel free to do so.

[1] A complete graph is called a *clique*.

Let $S$ be some optimum solution of size $|S| = k$.

The algorithm will be recursive: Let $u$ be a vertex of minimum degree in $G$. For every $v \in N[u]$, we recurse on $G \setminus N[v]$ while adding $v$ to the solution, and we return the choice which maximizes the solution size. If the graph on input has size 0, return 0.

The correctness is based on the following observation. Prove it.

**Observation:** For every $u \in V$, $N[u] \cap S \neq \emptyset$.

**Task:** Bound the running time of the algorithm.