For problems that ask to prove that a given problem $X$ is \textbf{NP}-hard, a full-credit solution requires the following components:

- Specify a known \textbf{NP}-hard problem $Y$, taken from the problems listed in Erickson’s notes.

- Describe a polynomial-time algorithm for $Y$, using a black-box polynomial-time algorithm for $X$ as a subroutine. Most \textbf{NP}-hardness reductions have the following form: given an arbitrary instance of $Y$, describe how to transform it into an instance of $X$, pass this instance to a black-box algorithm for $X$, and finally, describe how to transform the output of the black-box subroutine to the final output solving the original instance of $Y$. A diagram can be helpful.

- Prove that your reduction is correct. As usual, correctness proofs for \textbf{NP}-hardness reductions usually have two components, representing that the answer is true/false, or representing that the answer is too-large/too-small.

All (non-optional) problems are of equal value.

1. The directed Hamiltonian-path problem seeks to decide whether a given directed graph $G = (V, E)$ has a path that visits each vertex exactly once. Suppose you have a black-box algorithm for solving the directed Hamiltonian-path problem (note that this algorithm only answers ‘yes’ or ‘no’). Using this black-box algorithm, describe a polynomial-time algorithm that, given a directed graph $G = (V, E)$, outputs a Hamiltonian-cycle in $G$ if it has one, or returns ‘no’ otherwise.

\textit{Note:} You are allowed to use the algorithm solving the directed Hamiltonian-path problem more than once.

2. An instance of \textsc{SubsetSum} consists of $n$ non-negative integers $a_1, a_2, \ldots, a_n$ and a non-negative target integer $B$. The goal is to decide if there is a subset of the $n$ numbers whose sum is exactly $B$. The \textsc{2Partition} problem is the following: given $n$ (not necessarily non-negative) integers $a_1, a_2, \ldots, a_n$, is there a subset $S$ such that the sum of the numbers in $S$ is equal to $\frac{1}{2} \sum_{i=1}^{n} a_i$. Describe an efficient reduction from \textsc{SubsetSum} to \textsc{2Partition}.

\textit{Note:} One can also show that \textsc{2Partition} reduces to \textsc{SubsetSum}, but this requires slight additional work as the \textsc{SubsetSum} problem does not allow negative $a_i$.

3. Given an undirected graph $G = (V, E)$ a subset $S \subseteq V$ is a \textit{dominating set} for $G$ if for all $v \in V$, we have that $v \in S$ or there is a neighbor of $v$ in $S$. The \textsc{DominatingSet} problem is the following: given $G$ and an integer $k$, does $G$ have a dominating set of size $\leq k$?

\textit{(a)} \textit{(optional, not for submission)} Reduce \textsc{DominatingSet} to \textsc{SetCover}.

\textit{(b)} Reduce \textsc{SetCover} to \textsc{DominatingSet}.