1. **(30 pts)**

   A **vertex cover** is a set $S$ of vertices so that every edge of $G$ has an endpoint in $S$.

   Consider the following problem:

   **WEIGHTED-VERTEX-COVER**

   Input: A graph $G$ with a positive integer weight on every vertex.
   
   Output: A vertex cover $S$ with the smallest possible total weight.

   We’ll use the abbreviation WVC.

   Find a Yes/No problem in NP which is equivalent to WVC.

   Prove (i) that any polytime algorithm for your problem can be used to provide a polytime algorithm for WVC, and (ii) that any polytime algorithm for WVC can be used to provide a polytime algorithm for your problem.

   Then prove that your problem is in NP, using either a verification algorithm or a Non-deterministic Turing Machine.

2. **(25 pts)** The Bin Packing problem is: You are given a set of items (eg. files), each with a certain size. You must pack them into bins (eg. discs) each with the same given capacity, using as few bins as possible. The decision version is:

   **BIN-PACK**

   Input: A list of integers $x_1, ..., x_n$, a capacity $W$ and a target $T$.

   Question: Is there a partition of $\{1, ..., n\}$ into $B_1, ..., B_t$, for some $t \leq T$, such that for every $j$ we have $\sum_{i \in B_j} x_i \leq W$?

   (a) Prove that BIN-PACK is in NP using a verification algorithm.

   (b) Prove that BIN-PACK is in NP using a Non-deterministic Turing Machine.

   (c) Prove that BIN-PACK is NP-complete.
3. **(30 pts)**

**KILL-CYCLES**

Input: A directed graph $G$, and a target $T$.

Question: Is it possible to remove $T$ or fewer edges from $G$ so that the remaining graph does not have any directed cycle?

A *directed cycle* is a cycle where all the edges point in one direction around the cycle. Eg. $u \rightarrow v, v \rightarrow w, w \rightarrow x, x \rightarrow y, y \rightarrow u$.

(a) Prove that KILL-CYCLES is in NP.

(b) Prove that KILL-CYCLES is NP-complete.

   Hint: Show that VERTEX-COVER $\leq_p$ KILL-CYCLES. Given an input $G$ for VERTEX-COVER, you must construct an input $D$ for KILL-CYCLES. For each vertex $v \in G$, there will be two vertices $v_1, v_2 \in D$. Think about: in VERTEX-COVER you are looking for a small set of vertices, while in KILL-CYCLES you are looking for a small set of edges. So each vertex in $G$ should correspond to an edge in $D$.

4. **(30 pts)** For each of the following, either prove that the problem is in $P$ or prove that it is NP-complete. You do not have to prove that the problems are in NP.

   (a) **5-SAT**
      
      Input: A CNF boolean formula $F$ in which every clause contains exactly 5 literals.
      
      Question: Is $F$ satisfiable?

   (b) **LONG-PATH**
      
      Input: An undirected graph $G$ with two specified vertices $u, v$, and a target $T$.
      
      Question: Does $G$ have a path from $u$ to $v$ of length at least $T$?
      
      Clarification: A path does not repeat any vertices. Its length is the number of edges.

   (c) **CONNECTED-SUBGRAPH**
      
      Input: An undirected graph $G$ with an integer weight $w_e \geq 0$ on each edge $e$, and a target $T$.
      
      Question: Does $G$ contain a connected subgraph using every vertex of $G$ whose edges have total weight at most $T$?

5. **(12 pts)** Prove that for every language $L \in NP$, there is a constant $c$ such that $L$ can be decided in time $O(2^{n^c})$.

6. **(12 pts)** Prove that if there is an NP-complete language $Q$ in co-NP, then NP=co-NP.