1. **(GI is Self Reducible.)** Define a language $L$ to be *downward self reducible* if there’s a polynomial-time algorithm $R$ that for any $n$ and $x \in \{0, 1\}^n$, $R^{L_{n-1}}(x) = L(x)$ where by $L_k$ we denote an oracle that solves $L$ on inputs of size at most $k$.

**The GI (Graph Isomorphism) problem:** given two graphs $G_1 = (V_1, E_1), G_2 = (V_2, E_2)$ decide if they are isomorphic, that is, there exists a 1-1 and onto mapping $\pi : V_1 \rightarrow V_2$ such that $(u, v) \in E_1$ iff $(\pi(u), \pi(v)) \in E_2$.

Show that the problem GI is downward self reducible. That is, prove that given two graphs $G_1, G_2$ on $n$ vertices and access to a subroutine $P$ that solves the GI problem on graphs with up to $n - 1$ vertices, we can decide whether or not $G_1$ and $G_2$ are isomorphic in polynomial time.

2. **(Oracle Machines.)**
   
   (a) Are $P^{NP}$ and NP the same?
   
   (b) Prove that $P^{PSPACE} = PSPACE$.
   
   (c) Which complexity class is $PSPACE^{PSPACE}$?
   
   (d) Which complexity class is $P^{EXP}$?
   
   (e) Which complexity class is $EXP^{EXP}$?

3. **(NP and co-NP.)**
   
   (a) Let $L_1, L_2$ be two languages in NP.
   Show that $L_1 \cap L_2 \in NP$ and that $L_1 \cup L_2 \in NP$.
   
   (b) Let $L_1, L_2$ be two languages in NP $\cap$ co-NP and let
   \[ L_1 \oplus L_2 = \{ x : x \text{ is in exactly one of } L_1, L_2 \} \]. Show that $L_1 \oplus L_2$ is in NP $\cap$ co-NP as well.

4. **(PRIME)** Let $PRIME$ be the language of all prime numbers. Show that $PRIME$ is in co-NP (without using the fact that $PRIME$ is in $P$).

5. **(Pratt Certificate - Bonus)** Consider the following algorithm that verifies whether a given integer $n > 2$ is prime:
• Guess an integer $1 < r < n$.
• Verify that: $r^{n-1} \mod n = 1$.
• Guess $q_1, ..., q_k$ and verify that:
  - $q_1 \cdot q_2 \cdot ... \cdot q_k = (n - 1)$
  - For every $1 \leq i \leq k$, $q_i$ is prime.
  - For every $1 \leq i \leq k$, $r \frac{n-1}{q_i} \mod n \neq 1$.

Answer the following questions:

(a) Prove the correctness of the algorithm, assuming the following theorem:

**Lucas test:**
Let $n$ be a positive integer. If there exists an integer $1 < a < n$ such that $a^{n-1} \mod n = 1$ and for every prime factor $q$ of $n - 1$ the following holds: $a^{(n-1)/q} \mod n \neq 1$, then $n$ is prime (otherwise $n$ is either 1 or composite).

(b) Prove that the language accepted by this algorithm is in NP. What is the bound on the running time you found?

(c) Given a prime integer $n$ as an input for this algorithm, describe the witness $w_n$ for showing that $n$ is prime. What is the bound on the length of the witness?