Do four of the following five problems:

1. **FP** is the set of functions from \( \{0,1\}^* \) to \( \{0,1\}^* \) that can be computed by a deterministic Turing Machine in polynomial time. Show that computing the permanent of a matrix with integer entries can be done in **FP**\#**SAT**. Note that the integer entries may be negative but you will get partial credit if you prove this under the restriction of non-negative entries.

2. 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 \( k \). Prove that if \( L \) is downward-self-reducible, then \( L \in \text{PSPACE} \).

3. A **strong** non-deterministic Turing Machine is one that has three possible outcomes: "yes", "no" and "maybe". We say that such a machine decides a language \( L \) if the following is true: whenever \( x \in L \), then all computations end up with "yes" or "maybe" and at least one ends up with "yes". If \( x \not\in L \), then all computations end up with "no" or "maybe" and at least one ends up with "no". Show that if \( L \) is decided by a strong non-deterministic machine in polynomial time then \( L \in \text{NP} \cap \text{co-NP} \).

4. Prove that every language \( L \) in **NL** that is not the empty set or \( \{0,1\}^* \) is complete for **NL** under polynomial time reductions.

5. Recall the definition of **QSAT**:

\[
\text{QSAT} = \{ \Phi(x_1, \ldots, x_n) \mid \exists x_1 \forall x_2 \ldots \forall x_n \Phi(x_1, \ldots, x_n) = 1 \},
\]

where \( \Phi \) is a 3-CNF formula. Show that \( P^{\text{QSAT}} = N P^{\text{QSAT}} \).