1. The *Generalized Eulerian Cycle* problem is given a directed graph $G = (V, E)$ with $n$ nodes, $m$ edges numbered 1 to $m$, and a list of $m$ non-negative integers $k_1, k_2, \ldots, k_m$, where $0 \leq k_i \leq n$, is there a cycle that crosses edge $e_i$ exactly $k_i$ times (for $1 \leq i \leq m$)?

   Show that the Generalized Eulerian Cycle problem is in NP. Make sure to state what the certificate is, and argue that the verification is polynomial time.

2. Consider the following formula in Conjunctive Normal Form (CNF).

$$(A \vee B \vee C)(\bar{A} \vee B \vee C)(A \vee \bar{B} \vee C)(A \vee B \vee \bar{C})$$

   (a) Find a satisfying assignment.

   (b) Find an assignment that satisfies exactly three clauses.

3. Given a formula in CNF, even if it is not satisfiable, we might want to find an assignment to the variables that satisfies as many clauses as possible. This is the *Maximum Satisfiability* problem. To keep things simple, you would want an algorithm to not only determine the number of clauses that are satisfiable, but at the same time also find the maximum satisfying assignment.

   (a) Define the decision version of the Maximum Satisfiability problem. (Here you are only concerned about the number of clauses that are satisfied, not the assignment itself.)

   (b) Show that the decision version of the Maximum Satisfiability problem is in NP.

   (c) Show that if you could solve the optimization version in polynomial time that you could also solve the decision version in polynomial time.

   (d) Show that if you could solve the decision version in polynomial time that you could also solve the optimization version in polynomial time. Note that there are two steps here: First find the number of clauses that are satisfied, and then find the assignment itself.