1. (20 points) Show that the language HALT is \( \text{NP} \)-hard. Is it \( \text{NP} \)-complete?

2. (20 points) Show that, if \( P = \text{NP} \), then every language \( A \in P \), except \( A = \emptyset \) and \( A = \Sigma^* \), is \( \text{NP} \)-complete. Here \( \Sigma \) is the alphabet, and you may assume that it is \{0, 1\}.

3. (20 points) Let \( \phi \) be a 3CNF. An \( \neq \)-assignment to the variables of \( \phi \) is one where each clause contains two literals with unequal truth values.
   (a) Show that any \( \neq \)-assignment automatically satisfies \( \phi \), and the negation of any \( \neq \)-assignment to \( \phi \) is also an \( \neq \)-assignment.
   (b) Let \( \neq \text{SAT} \) be the collection of 3CNFs that have an \( \neq \)-assignment. Show that we obtain a polynomial time reduction from 3SAT to \( \neq \text{SAT} \) by replacing each clause

   \[ c_i = (y_1 \lor y_2 \lor y_3) \]

   with the two clauses

   \[ (y_1 \lor y_2 \lor z_i) \text{ and } (\overline{z}_i \lor y_3 \lor b), \]

   where \( z_i \) is a new variable for each clause \( c_i \) and \( b \) is a single additional new variable.
   (c) Conclude that \( \neq \text{SAT} \) is \( \text{NP} \)-complete.

4. (20 points) Let \( G \) be an undirected graph and let

   \( \text{LPATH} = \{ \langle G, a, b, k \rangle | G \text{ contains a simple path of length at least } k \text{ from } a \text{ to } b. \} \)

   Show that \( \text{LPATH} \) is \( \text{NP} \)-complete. You may use the \( \text{NP} \)-completeness of the undirected graph Hamiltonian path problem.

5. (20 points) A subset of the nodes of a graph \( G \) is a dominating set if every other node of \( G \) is adjacent to some node in the subset. Let

   \( \text{DOMINATING-SET} = \{ \langle G, k \rangle | G \text{ has a dominating set with } k \text{ nodes} \}. \)

   Show that it is \( \text{NP} \)-complete by giving a reduction from VERTEX-COVER.