1. Prove that the problem of deciding whether a digraph is strongly connected is complete for \textbf{NL}.

2. Define a \textit{coding} \( \kappa \) to be a mapping from \( \Sigma \) to \( \Sigma \). Note that \( \kappa \) need not be one-to-one. If \( x = \sigma_1 \ldots \sigma_n \), where each \( \sigma_i \in \Sigma \), then we define \( \kappa(x) = \kappa(\sigma_1) \ldots \kappa(\sigma_n) \). If \( L \) is a language, then \( \kappa(L) \) is defined to be \( \{ \kappa(x) \mid x \in L \} \).

   (a) Prove that \( \text{NP} \) is closed under codings. That is, show that if \( L \in \text{NP} \) and \( \kappa \) is a coding defined on the alphabet of \( L \), then \( \kappa(L) \in \text{NP} \).

   (b) We expect that \( \text{P} \) is not closed under codings, but we can not prove this without establishing that \( \text{P} \neq \text{NP} \). Instead, show that \( \text{P} \) is closed under codings if and only if \( \text{P} = \text{NP} \).

3. Prove that if \textit{every} unary language in \( \text{NP} \) is also in \( \text{P} \), then \( \text{EXP} = \text{NEXP} \). Recall that a language is unary if and only if it is a subset of \( 1^* \).

4. Say that class \( C_1 \) is \textit{superior} to \( C_2 \) if there is a language \( L_1 \) in \( C_1 \) such that for every language \( L_2 \in C_2 \) and every \( n \) sufficiently large, there is an input whose length is between \( n \) and \( n^2 \) on which \( L_1 \) and \( L_2 \) differ. That is, there is a string \( x \) whose length is between \( n \) and \( n^2 \) such that either \( x \in L_1 \) and \( x \notin L_2 \) or \( x \notin L_1 \) and \( x \in L_2 \).

   (a) Is \( \text{DTIME}(n^4) \) superior to \( \text{DTIME}(n) \)?

   (b) Why does our proof of the Non-deterministic Time Hierarchy not prove that \( \text{NTIME}(n^{1.5}) \) is superior to \( \text{NTIME}(n) \)?

5. In the Arora-Boak text gives an alternative definition of the class \( \text{NL} \) which makes use of a Turing Machine with a special read-once tape. The head on a read-once tape starts at the left-most end of the non-blank symbols written on the tape and can only move to the right or stay in the same place (i.e. it can never move left). The alternative definition says that a language \( L \) is in \( \text{NL} \) if there is a deterministic Turing Machine \( M \) (called a \textit{verifier}) with a special read-once tape and a polynomial \( p \) such that for every \( x \in \Sigma^* \),

\[
x \in L \iff \exists u \in \Sigma^{p(|x|)} \text{ such that } M(x, u) = 1,
\]

where \( M(x, u) \) is the output of \( M \) when \( x \) is placed on the input tape and \( u \) is placed on its special read-once tape and \( M \) uses \( O(\log n) \) space on its work tape for every input \( x \).

Prove that this definition is equivalent to the definition using non-deterministic Turing Machines discussed in class.
6. Prove that if in the above definition, the read-once tape is replaced with a read-only tape (which could be read many times), then the resulting class is NP.