1. (Based on [MR95], problem 4.1):
   Suppose you are given a biased coin that has \( \Pr[\text{HEADS}] = h \), for \( a \leq h \leq b \), for some fixed \( a \) and \( b \).
   You are not given any other information about \( h \) (i.e., you can not assume that it is chosen randomly).

   (a) Using Chebyshev’s inequality, devise a procedure for estimating \( h \) by a value \( \hat{h} \) such that you can guarantee that \( \Pr[|h - \hat{h}| > \rho h] < \mu \), for any choice of the constants \( 0 < a, b, \rho, \mu < 1 \). Let \( N \) be the number of times you need to flip the biased coin to obtain the estimate, where \( N \) is a function of \( a, b, \rho, \) and \( \mu \). What is the smallest value of \( N \) for which you can still give this guarantee?

   (b) In Lecture 16, we saw the following two Chernoff bounds:
      \[
      \Pr[B(n,p) \leq (1 - \delta)np] \leq e^{-\delta^2 np/2}, \quad \text{and} \quad \Pr[B(n,p) \geq (1 + \delta)np] \leq e^{-\delta^2 np/3},
      \]
      where \( B(n,p) \) is a random variable representing the number of heads seen in \( n \) tosses of a coin that is heads with probability \( p \). Using these bounds, what is the smallest value of \( N \) for which you can still give the guarantee that \( \Pr[|h - \hat{h}| > \rho h] < \mu \)?

2. CLR, Problem 36-1 (Page 961).

3. The decision version of the clique problem can be stated as follows:
   INPUT: An undirected graph \( G \) and an integer \( k \).
   QUESTION: Does \( G \) contain a clique of size \( k \)?
   Prove that if we can solve the decision version of the clique problem in polynomial time, then we can find the largest clique in any graph in polynomial time.

4. A Boolean formula is in \( k \)-CNF form if it is in conjunctive normal form, and has exactly \( k \) literals in each clause. We can thus define, for each \( k \), the problem \( k \)-SAT:
   INPUT: A Boolean formula \( \Phi \) in \( k \)-CNF.
   QUESTION: Is \( \Phi \) satisfiable?

   (a) Using the fact that 3-SAT is NP-Complete, show that \( k \)-SAT is NP-Compete, for any \( k > 3 \).

   (b) Prove that 2-SAT is not NP-Complete unless \( P=NP \).
      (Hint: construct a directed graph, where vertices correspond to literals in \( \Phi \), and edges correspond to “implications” from the clauses. The graph should have the property that \( \Phi \) is satisfiable if and only if the graph does not contain a directed path of a certain form.)