1. For the randomized routing algorithm from lecture 11, suppose that you want to guarantee that all packets reach their destination in time $T(n)$ with probability $1 - 2^{-n}$. How small can $T(n)$ be?

2. Consider a program that computes $B = f(A)$, where $A, B$ are $n \times n$ matrices with rational coefficients, $A$ is the input, and $B$ the output from the program. $B$ is supposed to be the inverse of $A$. Give a randomized program checker for this task. What is the complexity of your checker? Assume arithmetic operations have unit cost.

3. Suppose you are given a large polynomial $p(a_{11}, \ldots, a_{ij}, \ldots, a_{nn})$ which is claimed to be the determinant of the matrix $A = \{a_{ij}\}$. Describe a program checker, compute is running time and error bound.

4. Let $a = a_1a_2 \cdots a_n$ be an $n$-character string. Your task is to compute fingerprints of all length $k$ contiguous substrings of this string (for pattern matching). How would you do this? You should be able to do it in $O(n)$ time independent of $k$. 