Fall 08. CS513. HW 4, Due Oct 16. You can assume any of the algorithms we discussed in the class: you should precisely state any such result you use without going into details. Your goal is as always to design the fastest algorithm possible and to use as little space as possible.

1. Design an efficient algorithm to verify the Vandermonde's identity

$$\Pi_{i<j}(x_i - x_j) = \begin{vmatrix} 1 & x_1 & .. & x_1^{n-1} \\ 1 & x_2 & .. & x_2^{n-1} \\ ... & ... & ... & ... \\ 1 & x_n & .. & x_n^{n-1} \end{vmatrix}.$$

   In particular, make sure the numeric computations don't get too large. Assume determinant $|M|$ for a matrix M of $n \times n$ entries of integers of $O(\log n)$ bits can be computed in $T(n)$ time.

2. Given integers a, b and n, design an efficient algorithm to compute $\frac{a}{b}$ mod n.

3. Given strings $S[1...n]$ and $T[1..m]$, $n \geq m$, find all locations i where $s[i, ...i + m - 1]$ and T differ in no more than k places, that is, for such i, there should be at most k of j's such that $S[i + j - 1] \neq T[j]$. Use Karp-Rabin fingerprints.

4. Extend the closest pairs algorithm in the class to 3 dimensions, and then more generally to d dimensions. What is the complexity of your algorithm?

5. Assume you have access to a blackbox that tests if a given number is a prime. Design a fast randomized algorithm to pick a prime number in $[x, y]$, and analyze it in terms of x and y.