Problem 1. Arrange the following functions in order of increasing growth rate, $\sqrt{n}, 10^n, n^{1.5}, 2^{\sqrt{lg_2 n}}$, $n^{5/3}$.

Problem 2. Suppose you're given n points in the plane, each specified by (x, y) coordinates, and you'd like to find the pair of points that are closest together. Intuitively think about the brute force algorithm you would use to find the solution. Don't write the pseudo-code just mention what you would do and explain the runtime of your algorithm.

Problem 3. We are given sets $S_1, S_2, \ldots, S_n$, each of which is a subset of $\{1, 2, \ldots, n\}$, and we want to know whether there are any pairs of sets that are disjoint (sets that have no elements in common). You can assume that each set S_i is represented in such a way that we can check in constant time ($O(1)$) whether a given element x belongs to S_i. Write brute force pseudo-code in English sentences to solve this problem. What is the asymptotic runtime of your algorithm?

Problem 4. You are given two arrays, A and B, of lengths, m and n, respectively. Write pseudo-code (as shown in class) to find whether or not there is a number x in both A and B. Return TRUE if you find such number and FALSE otherwise. Find the exact total work (number of comparisons) in the worst case. Show your work.

Problem 5. You are given two sorted arrays, A and B, of lengths, m and n, respectively. Write an efficient algorithm to merge them into a single sorted array, C, in the worst case. What is the asymptotic runtime of your algorithm?