MAT344 Problem Set 4
(due 1pm June 4)

Note: For all the questions, always *explain your reasoning* and refer to the results you are using. Just a number (even if it is the correct final answer) will **not** get you full credit.

Part A

In these questions, you will *analyze a counting problem by proving an exact or approximate enumeration* and *identify when an exact solution is intractable, and use estimates to describe its approximate size*. Two of these questions will be marked.

Problem 1. One version of the algorithm `bogosort` (also known as `slowsort` or `stupid sort`) can be described as follows: Given a list of integers, check if the list is sorted. If it is sorted, return the sorted list. If not, pick a random permutation of the elements, then repeat. Assuming that we never use the same permutation twice, what is the complexity (in Big Oh notation) of the algorithm? As with other sorting algorithms, you should use the number of comparisons as the operation.

Problem 2. A group of 30 students are standing in a line. We plan to sort them into order of increasing student number by comparing two at a time and swapping them if they are not in order. What is the maximum number of swaps that we will need to do? You can assume that all students have distinct student numbers.

Problem 3. Show that given a square of area 1, any way of drawing five dots within the square will yield at least two dots that are within distance $\frac{3}{4}$ of each other.

Problem 4. There are 141 students currently enrolled in MAT344 and four tutorials. You want to make an argument to the U of T administration that current tutorial sizes are too large, but you do not have access to the exact tutorial enrollment statistics. You want to make a statement of the form:

There is a tutorial with at least n people currently enrolled.

You want to make your statement effective (i.e. n as large as possible), but you also don't want to lie. What is the most effective statement you can make?

Part B

In these questions, you will *select and justify appropriate tools to analyze a counting problem* and *analyze a counting problem by proving an exact or approximate enumeration.* Two of these questions will be marked.

Problem 5. Prove by induction that `Merge sort` correctly sorts a list. **Hint:** Let S_n be the statement "`Merge sort` correctly sorts a list with n elements".

Problem 6. Prove that if you have 100 integers, you can choose 15 of them so that the difference of any two is divisible by 7.

Problem 7. Each of the following combinatorial quantities are functions of a positive integer n.

1. Let $a(n)$ denote the number of possible pairs of people in an n person class.
2. Let $b(n)$ denote the number of number of comparisons needed to determine whether if a specific integer is in a sorted list using binary search.
3. Let $c(n)$ denote the number of n-digit positive integers.
4. Let $d(n)$ denote the number of possible ways to rank n contestants.

Order the following functions by their asymptotic growth rates. In other words, list the functions as $f_1(n), f_2(n), \ldots$, where for all i, we have that $f_i(n) = O(f_{i+1}(n))$.

Part C

This question relates to the learning outcomes *construct counting problems which show the usefulness or limitations of combinatorial tools.* It will be marked for completeness only.

Problem 8. Give an example of a counting problem that may be difficult to solve exactly with tools from previous chapters, but where you can give a reasonable approximation.