Homework 15: Due Friday, May 5

Note: For each of the counting problems below, you must explain your solution. For example, if your answer is a product, describe the sequence of choices you are making and explain where each term comes from. Numerical answers without written justification will receive no credit.

Problem 1: Suppose that a lottery draws 6 numbers from $\{1, 2, \ldots, 60\}$ without replacement and where order drawn doesn't matter. What percentage of possible lottery numbers have 3 evens and 3 odds?

Problem 2: In class, we talked about the number of paths starting at $(0, 0)$ and ending at (m, n) where each step was either one step north or one step east. How many such paths are there from $(0, 0)$ to $(12, 9)$ which do not go through the point $(5, 4)$? Think of needing to avoid that intersection because of construction.

Problem 3: How many 5-card poker hands have at least one card of every suit?

Problem 4: Snow White and the seven dwarfs (Bashful, Doc, Dopey, Grumpy, Happy, Sleepy and Sneezy) plan to eat dinner at a long rectangular table with four seats on each of two sides, and no seats on the two ends (so there there are four pairs of two seats opposite each other). Suppose that all seats are considered distinct.
a. Grumpy gets excessively grumpy if he is seated directly across the table from Sneezy. How many ways can the eight be seated so that those two are not across from each other?
b. Doc insists on sitting directly between Dopey and Sleepy to ensure they don't make fools of themselves. How many such seating arrangements are there?
c. How many arrangements satisfy both of the demands of part a and part b?

Problem 5: In this problem, you will write an ML function choose that works without computing factorials. Instead, it will compute rows of Pascal's triangle recursively.
a. Write an ML function addLists that takes two inputs, each of type IntInf.int list, and produces the IntInf.int list obtained by adding the corresponding entries of the lists. For example, addLists([1, 2, 3], [4, 5, 6]) should produce [5, 7, 9]. If one list is longer than the other, the tail of the corresponding list should be ignored. For example, addLists([1, 2, 3], [4, 5]) should produce [5, 7].
b. Write an ML function nextPascalRow that takes as input an IntInf.int list, assumed to be a full row of Pascal's triangle (up to, but not including the 0's), and produces the next row of Pascal's triangle. For example, nextPascalRow([]) = [1], nextPascalRow([1]) = [1, 1], and nextPascalRow([1, 2, 1]) = [1, 3, 3, 1].
c. Write an ML function pascalRow that takes an input n of type int (assumed to be nonnegative), and produces the nth row of Pascal's triangle. For example, pascalRow(0) = [1], pascalRow(1) = [1, 1], and pascalRow(4) = [1, 4, 6, 4, 1]. Your function should work very quickly on inputs up to 2000.
d. Using part c, write an ML function choose that takes two inputs n and k with $0 \le k \le n$, each of type int, and produces the value $\binom{n}{k}$ (of type IntInf.int). A function you wrote in a previous homework might be useful.
Aside: If you are only interested in one particular value of $\binom{n}{k}$, it is typically faster to use factorials. However, if you are interested in the values across a given row of Pascal's triangle, then part c makes much more sense.