CSE 5311 Lab Assignment 1

Due February 20, 2019

Goals:

1. Understanding of coupon collecting.
2. Understanding of enumeration.
3. Understanding of random simulation to verify a probability result

Requirements:

1. The following paper does a variety of interesting probabilistic analyses:

   P. Flajolet et.al., "Birthday paradox, coupon collectors, caching algorithms and self-organizing search", *Discrete Applied Mathematics 39* (1992), 207-229. (`http://algo.inria.fr/flajolet/Publications/FlGaTh92.pdf`)

   It includes the following formula, which provides the expected number of coupons needed under a general probability distribution P for m coupons:

   $$\sum_{q=0}^{m-1} (-1)^{m-1-q} \sum_{|J|=q} \frac{1}{1-P_J} \qquad \text{(14b)} \qquad \text{where } P_J = \sum_{i \in J} P_i$$

   For m=3 and $(p_1, p_2, p_3) = (a, b, c)$, the paper simplifies (14b) to:

   $$1 - \frac{1}{1-a} - \frac{1}{1-b} - \frac{1}{1-c} + \frac{1}{1-a-b} + \frac{1}{1-b-c} + \frac{1}{1-c-a}.$$

   Your task is write a C/C++ program to 1) evaluate (14b), or an equivalent expression, directly by enumerating the powerset of the indices for P and 2) implement a simple random simulation of generating coupons for the generalized situation. Your program must compile and execute on at least one of `omega.uta.edu` or Visual Studio.

   (Formula (1) in `https://web.cs.wpi.edu/~hofri/CCP.pdf` says essentially the same thing as the formula above.)

2. Submit your C/C++ code on Blackboard before 3:45 p.m. on February 20. Be sure to include comments regarding how to compile and execute your code.

Getting Started:

1. m will not exceed 30.
2. The input is very simple:
   a. The first input is m.
   b. The next m values are positive frequency values, in ascending order, that may be used to compute P. These may appear across a number of input lines.
   c. The last input line will be the number of times the random simulation should be ran, along with a seed for the random number generator. (The only significance of the seed is in the reproducibility of the experiments.)
3. Your powerset approach should use $\Theta\left(2^m\right)$ time, i.e. $\Theta(1)$ amortized time per subset

(see `http://page.math.tu-berlin.de/~muetze/cos/`) and $\Theta(m)$ space. Also see Notes 03, Elementary Example 3.