CSC 226 SUMMER 2020
ALGORITHMS AND DATA STRUCTURES II
ASSIGNMENT 1
UNIVERSITY OF VICTORIA

1. An n-degree ***polynomial*** $p(x)$ is an equation of the form

$$p(x) = \sum_{i=0}^{n} a_i x^i$$

where x is a real number and each a_i is a real constant, with $a_n \neq 0$.

   a. Describe a simple $O(n^2)$-time method for computing $p(x)$ for a particular value of x. Justify the runtime.
   b. Consider now the nested form of p(x), written

$$p(x) = a_0 + x\left(a_1 + x\big(a_2 + x(a_3 + \cdots + x(a_{n-1} + xa_n)\cdots)\big)\right).$$

   Using the big-Oh notation, characterize the number of multiplications and additions this method of evaluation uses.

2. Suppose that if the linear selection algorithm used subsequences of 3 instead of 5 or 7, then show that the worst-case runtime would no longer be $O(n)$. You may follow the lecture slide examples from lecture 4 when developing your recurrence equation $T(n)$ for this version of the algorithm. That is, you can use upper bounds to get rid of the ceiling notation and assume an in-place implementation which costs nothing to separate into subsequences. [Note: For a subsequence of 3 elements, it takes at most 3 comparisons to sort them.]

3. Write an in-place version of the linear selection algorithm, using subarrays of size 7 for the median of medians selection. Use the linearSelect(S,k) algorithm in slide 4 of lecture 4 and the inPlaceQuickSort(S,a,b) algorithm in the Goodrich and Tamassia textbook (page 255) as your guideline for writing this algorithm. The algorithm should have the following heading,

   Algorithm inPlaceLinearSelect(S,a,b,k)
   Input: An array, S, of distinct elements; integers a and b such $a \leq b$; and integer $k \in [a+1, b+1]$
   Output: The kth smallest element of S

4. Consider the experiment of tossing a fair coin four times,
   a. What is sample space associated with this experiment? Label a heads H and a tails T.
   b. Let A be the event that heads are flipped first. What is $\Pr(A)$?
   c. Let B be the event that exactly two heads and two tails are flipped (in any order). What is $\Pr(B)$?
   d. Let A and B be as defined above, what is $\Pr(A \cap B)$?
   e. Let X be the number of heads flipped. What is the expected value of X?

5. Suppose you would like to sort n music files using a comparison-based sorting algorithm (i.e. no bucket sort), but you only have an old, unreliable computer, which you have nicknamed "Rustbucket". Every time Rustbucket compares two music files, x and y, there is an independent 50-50 chance that it has an internal disk fault and returns the value -1, instead of the correct result of 1 for **true** or 0 for **false**, to the question, "$x \leq y$?" Otherwise, Rustbucket correctly performs every other kind of operation (including comparisons not involving music files.) Describe an efficient algorithm that can use Rustbucket to sort n music files correctly and show that your algorithm has *expected* running time that is $O(n \ \log n)$.