CSC 225 SPRING 2017
ALGORITHMS AND DATA STRUCTURES I
ASSIGNMENT 2
UNIVERSITY OF VICTORIA

1. Consider an implementation of a stack using an extendible array. That is, instead of giving up with a "StackFullException" when the stack becomes full, we replace the current array S of size N with a larger one of size $f(N)$ and continue processing the push operations. Suppose that we are given two possible choices to increase the size of the array: (1) $f(N) = N + c$ (for convenience, we start with an initial array of size 0) (2) $f(N) = 2N$ (we start with an initial array of size 1). Compare the two strategies and decide which one is better.

   To analyse the two choices, assume the following cost model: A "regular" push operation costs one unit of time. A "special" push operation, when the current stack is full, costs $f(N) + N + 1$ units of time. That is, we assume a cost of $f(N)$ units to create the new array, N units of time to copy the N elements and one unit of time to copy the new element.

2. Suppose that we are given an array A with n keys and k inversions. Here, an *inversion* is defined as a pair of entries that are out of order in the array. What is the running time of Insertion sort when it is used to sort A in Big Oh notation? Why?

3. Develop a $O(n \log n)$ algorithm for computing the number of inversions in a given array.

4. Solve the following recurrence equation to get a closed-formula for $T(n)$. Assume the n is a power of two.

$$\begin{aligned} T(n) &= 1 \text{ if } n = 1 \\ &= 4T\left(\frac{n}{2}\right) + n \log n \text{ if } n \geq 2 \end{aligned}$$

5. Suppose we are given a sequence S of n elements, each of which is an integer in the range $[0; n^2 - 1]$. Describe a simple method for sorting S in $O(n)$ time.