Home Work #4
DUE: 1159am Sunday Sep 20, 2020
(upload portrait-mode PDF on Canvas)

☞ Handwritten assignments will not be accepted.

Start your assignment with the following text provided you can honestly agree with it.

- I certify that every answer in this assignment is the result of my own work; that I have neither copied off the Internet nor from any one else's work; and I have not shared my answers or attempts at answers with anyone else.

1. Consider the following BUBBLESORT algorithm and two alleged loop invariants $\mathcal{I}_1$ and $\mathcal{I}_2$ for the outer loop (line 2) and inner loop (Line 3) respectively. You may assume that EXCHANGE(A, i, j) swaps $A[i]$ with $A[j]$ leaving other elements of A undisturbed.

   BUBBLESORT(A)
   1 $\quad n \leftarrow A.length$
   2 $\quad$ **for** pass $\leftarrow$ 1 **to** $(n-1)$ **do**
   3 $\qquad$ **for** $j \leftarrow 1$ **to** $(n - pass)$ **do**
   4 $\qquad\quad$ **if** $A[j] > A[j+1]$ **then**
   5 $\qquad\qquad$ EXCHANGE$(A, j, (j+1))$

   $\mathcal{I}_1$: $\quad$ *$A[(n-pass+2)\cdot\cdot\ n]$ is sorted in non-decreasing order **and** all values in that segment are greater than or equal to the values in $A[1\cdot\cdot(n-pass+1)]$ **and** $A[1..n]$ is a permutation of the original input array.*

   $\mathcal{I}_2$: $\quad$ *$A[j]$ is greater than or equal to the elements in $A[1..(j-1)]$, **and** the subarray $A[1..j]$ is a permutation of the same segment in the previous iteration **and** $A[(j+1)..n]$ is unchanged from the previous iteration.*

   (a) For each of the following, first answer *Yes/No*, and then prove your answer.
      i. Is $\mathcal{I}_2$ valid on initialization,
      ii. Is $\mathcal{I}_2$ preserved by each iteration, and
      iii. Is $\mathcal{I}_2$ useful at termination (i.e., it gets used in part (ii) for $\mathcal{I}_1$).

   (b) For each of the following, first answer *Yes/No*, and then prove your answer.
      i. Is $\mathcal{I}_1$ valid on initialization (i.e., when *pass* has just been assigned 1)?
      ii. Is $\mathcal{I}_1$ preserved by each iteration? (You may make use of $\mathcal{I}_2$.) and
      iii. Is $\mathcal{I}_1$ appropriate (i.e., does it demonstrate that the algorithm has achieved its intent) at termination ?
      iv. If your answer was *No* for any of the above, then make the smallest possible correction and justify your modified invariant.

2. Using the Substitution Method, verify the solution $T(n) = \Omega(n^2)$ for the following recurrence equations:

$$\begin{aligned} T(1) &= 1 \\ T(n) &= n + T(n-1) \quad \text{for } n > 1 \end{aligned}$$

3. Given the following recurrence equations, find a solution for $T(n)$ using a recursion tree. (You can assume that n is a power of 2 in order to eliminate floors and ceilings.) You must derive a polynomial in n and then infer a $\Theta()$ bound without deriving the constants c_1, c_2 and the thresold n_0.

$$\begin{aligned} T(1) &= 1 \\ T(n) &= n^2 + 2T(\tfrac{n}{2}) \quad \text{for } n > 1 \end{aligned}$$