SWE-1202 / CSC 1203 Data Structures and Algorithms

Test I

Date: 5th July 2023 at 1500 hours

Instructions

- Attempt all questions individually.
- This test has 2 questions, for a total of 50 points

Akandwanaho Joseph 2022/BCS/022/PS

1. (a) (6 points) Write down the formal definition of an algorithm.

   (b) (3 points) Fill in the blanks for INSERTION-SORT algorithm that sorts elements in **non-decreasing order.**

Algorithm 1 : Given a sequence of n elements $\langle a_1, a_2, \ldots, a_n \rangle$ as Input instance (I), return a permutation(reordering) $\langle a'_1, a'_2, \ldots, a'_n \rangle$ of I such that $a'_1 < a'_2 < \ldots < a'_n$

```
1: procedure INSERTION-SORT(A)
2:     for j ← 2 , to length[A] do
3:         key ← A[j]
4:         i ← j − 1                    ▷ Insert A[j] into sorted sequence A[1 . . . j − 1]
5:         while i > 0 and A[i] > key do
6:             A[i + 1] ← A[i]
7:             i ← j − 1
8:         A[ i + 1 ] ← key
```

   (c) (2 points) State the loop invariant(s) for INSERTION-SORT you wrote in (part b) above.

   (d) (2 points) What is the purpose of the while loop (lines 5-7)?

   (e) (4 points) Using figure 1 as a model, illustrate the operation of insertion sort on the array $\langle 31, 29, 59, 26, 29, 58 \rangle$

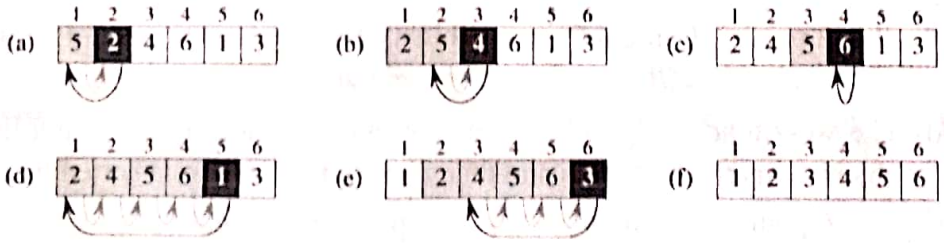

Figure 1: The operation of INSERTION-SORT on the array $A = \langle 5, 2, 4, 6, 1, 3 \rangle$

(f) (5 points) Loop invariants helps us prove the correctness of an algorithm. Provide an informal proof for correctness of INSERTION-SORT by explaining how the loop invariant is satisfied for **Initialisation, Maintenance** and **Termination.**

(g) (4 points) Show that the time complexity for INSERTION-SORT is $\mathcal{O}(n^2)$.

2. Recursion and iteration are some techniques used to solve problems in computer science.

(a) Finding the sum of the first n terms is equivalent to finding the nth triangular number;

i. (3 points) Devise a recursive solution for the above problem

ii. (2 points) Re-write the above solution as an iterative solution (Remember: iteration involves loops).

(b) A recurrence relation is said to have two primary properties; 1 - an equation defining how the nth term is attained, 2 - initial conditions that specify the terms that precede the first term where the relation takes effect.

i. (2 points) Given the following recurrence relation $a_n = a_{n-1} + 2a_{n-2}$ for $n \geq 2$ where $a_0 = 2$ and $a_1 = 5$; Generate the sequence of terms from $\{ a_0, a_1, ..., a_7 \}$.

ii. (1 point) Re-write the recursive portion of the Fibonacci algorithm as a recurrence relation.

(c) The Fibonacci algorithm is a means of finding the nth Fibonacci number;

i. (4 points) Write the algorithm above (in pseudo-code).

ii. (2 points) Represent the above algorithm as a mathematical function.

iii. (2 points) Using a recursive binary tree, compute the 6^{th} Fibonacci number (use "fib" as the procedure name).

(d) The MERGE-SORT algorithm is unique in that it partitions an input list of object, sorts the partitions and then merges them into one sorted list;

i. (6 points) Use pseudo-code to represent the algorithm.

ii. (2 points) Given the following unsorted list { 19, -1, 25, 0, -13 }, use the merge sort algorithm above to sort it. (Clearly show each step).