

Data Structures (IT205) 2014-15
First Midsemester-semester Exam
28th August, 2014

Time: 2 hours

marks: 60

This question paper consists of 3 questions printed on a single page back-to-back. Check that your question paper is complete. Each question is worth 20 marks.

1.
 - (a) Can the same data item move both left and right during the course of execution of bubble-sort on an input consisting of n distinct numbers? Justify your answer.
 - (b) When executing bubble sort on an array $A[1...n]$ of distinct elements, element $A[i]$ moves left if and only if $A[i]$ is NOT the _____ among _____.
 - (c) Modify the code of bubble sort such that the final output orders elements according to the total number of times they change position during the execution of the usual bubble sort. For those elements which move the same number of times sort them according to the usual convention of increasing order. You may assume that all the elements are distinct and the movement should be counted only with respect to the original bubble sort procedure; not movements on account of the extra requirements.
 - (d) In insertion sort involving distinct elements can the same element move both left and right? What about selection sort?
 - (e) Consider the input 6, 3, 1, 2, 4, 8, 5, 7 for a comparison based sorting algorithm \mathcal{A} for sorting in increasing order. If an almost identical algorithm \mathcal{A}' were designed the only change being to sort in decreasing order, then in that new algorithm \mathcal{A}' which input would cause behaviour similar to the specified input for \mathcal{A} ?
2. DS_1 : Suppose you are given a queue of five stacks each stack having the capacity of five integers. The insert operation in this two level data structure is such that an insert is like an enqueue (back most position of the queue), however each position being a stack of five integers, if that stack is full then the enqueue happens in the next stack. For the delete operation, the dequeue happens at the head of the queue and the corresponding stack's topmost element is returned.
 DS_2 : One can similarly define a stack of five queues each queue having a capacity of five integers.
 - (a) Suppose the elements $1, \dots, 25$ are inserted in the first data structure one after the other and then deleted and we print the keys upon deletion, what would be the output.
 - (b) Answer the same if the same 25 elements are inserted into the second data structure and deleted and printed.
 - (c) Now what will happen if the for second data structure the inserts are in the order in which elements are deleted from the first and we subsequently delete the second and print? That is, the elements $1, \dots, 25$ are inserted successively in DS_1 and then deleted from DS_1 and deleted into DS_2 in the same order in which they are deleted from DS_1 .
 - (d) Answer the same question if we were to transfer to the first from the second.

3. Procedure(L, key)

```
1.  $x \leftarrow HEAD[L]$ 
2. WHILE ( $x \neq null$ )
3.     IF ( $key > key[x]$ )
4.         THEN  $key \leftarrow key - 5$ 
5.          $x \leftarrow PREV[x]$ 
6.     ELSE  $key \leftarrow key + 10$ 
7.          $x \leftarrow NEXT[x]$ 
```

- (a) Does this code always, sometimes or never go into infinite loop.
- (b) If the answer is always or never, then justify. If the answer is sometimes then explain one with an input for which it goes into infinite loop and one for which it does not.

NOTE:The linked list L is assumed to have at least 5 elements. Do not give answers for degenerate/trivial cases like an empty list or a list with fewer than five elements. The input key to the procedure is an integer and the key values of the nodes in the linked list are also integers.