

Birla Institute of Technology & Science - Pilani, Hyderabad Campus
Second Semester 2013-2014 Test – 2 Date: 28.03.2014
CS F211 / IS F211: Data Structures & Algorithms
Type: Closed Book Time: 60mins Marks: 60

All parts of the same question should be answered together.

1.a. Suppose we need to maintain two stacks of the same type of items in a program. If the two stacks are stored in separate arrays, then one stack might overflow while there are considerable unused spaces in the other. Design a method for keeping two stacks within a single linear array so that neither stacks overflows until all of memory is used. [3 Marks]

1.b. Design a decent strategy or method to accommodate three stacks in an array solving the same issues that are mentioned in the above problem.
Hint: Moving data of one stack to different locations inside the array is accepted but these movements should be as minimal as possible. [5 Marks]

1.c. Devise an algorithm to multiply two long positive integers represented by singly linked circular lists. [8 Marks]

1.d. Write an algorithm to determine if an input character string is of the form $a D b D c D \dots D z$ where each string a, b, \dots, z is of the form of the string defined in previous problem. (Thus a string is in the proper form if it consists of any number of such strings separated by the character 'D'.) At each point you may read only the next character of the string. [4 Marks]

2.a. Suppose two singly linked lists have n_1 and n_2 nodes and both of them are intersecting at one node. Provide a naïve algorithm of $O(n_1 n_2)$ complexity to find the first intersecting node. Also devise an efficient algorithm of $O(n_1 + n_2)$ complexity for the same problem. [2 + 6 Marks]

2.b. Devise an algorithm for finding the k smallest elements of an unsorted set of n integers in $O(n + k \log n)$ complexity. [4 Marks]

2.c. Show how to implement a first-in, first-out queue with a priority queue. Show how to implement a stack with a priority queue. [8 Marks]

3.a. The operation $\text{HEAP-DELETE}(A, i)$ deletes the item in node i from heap A . Give an implementation of HEAP-DELETE that runs in $O(\lg n)$ time for an n -element max-heap. [6 Marks]

3.b. A ***d*-ary heap** is like a binary heap, but (with one possible exception) non-leaf nodes have d children instead of 2 children.

(i). How would you represent a d -ary heap in an array? [2 Marks]

(ii). What is the height of a d -ary heap of n elements in terms of n and d ? [3 Marks]

(iii). Give an efficient implementation of EXTRACT-MAX in a d -ary max-heap. Analyze its running time in terms of d and n . [4 Marks]

(iv). Give an efficient implementation of INSERT in a d -ary max-heap. Analyze its running time in terms of d and n . [5 Marks]