BIRLA INSTITUTE OF TECHNOLOGY, MESRA, RANCHI
(END SEMESTER EXAMINATION)

CLASS: M.TECH **SEMESTER : I**
BRANCH: CS/IT/IS **SESSION : MO/18**

SUBJECT: CS502 ADVANCE DATA STRUCTURE

TIME: 3.00 HOURS **FULL MARKS: 50**

INSTRUCTIONS:
1. The question paper contains 5 questions each of 10 marks and total 50 marks.
2. Attempt all questions.
3. The missing data, if any, may be assumed suitably.
***4*. Before attempting the question paper, be sure that you have got the correct question paper.**
5. Tables/Data hand book/Graph paper etc. to be supplied to the candidates in the examination hall.

Q.1(a) Implement symbol table in compiler design using hash function. [5]

Q.1(b) Construct an algorithm to compute the number of collisions required in a long random sequence of insertions using linear probing, quadratic probing, and double hashing. [5]

Q.2(a) Apply the routines to perform insertion, deletion, and searching in skip lists. [5]

Q.2(b) Justify that the expected time for the skip list operations is O(logN). [5]

Q.3(a) Justify that every AVL tree can be colored as a red-black tree. Are all red-black trees AVL? Write the deletion procedure for red-black trees. [5]

Q.3(b) Insert the numbers 1, 5, 2, 6, 3, 7, 4, 8 into a Splay tree. Construct the tree after each insertion, the tree after each Zig, Zag-zag and Zig-zig rotation. [5]
Analyze the Big-Ohs for Splay trees operations and its applications in real world.

Q.4(a) Differentiate the properties of binomial heap with Fibonacci heap. What can you say about the structure of binomial heaps and Fibonacci heaps? What do they have in common and in what way do they differ? Also compare binomial heaps and Fibonacci heaps in terms of running times. Can you think of scenarios in which it is better to use a binomial heap and scenarios in which it is better to use a Fibonacci heap? [5]

Q.4(b) Perform the following sequence of operations on an initially empty binomial heap. Create the binomial heap after each of the operations. [5]
a. insert(10) b. insert(3) c. insert(5) d. insert(2) e. insert(7)
f. decrease-key(10,1) g. insert(4) h. delete-min() i. delete-min() j. delete-min()

Q.5(a) Justify that string matching algorithm for the Knuth-Morris-Pratt is better than Boyer Moore. Construct string matching algorithm for the Knuth-Morris-Pratt. [5]

Q.5(b) Create the trie of all the suffixes of the word ‘banana’. Design an Algorithm for Insertion and Search in Trie. Write its different applications areas of Trie. [5]

:::::30/11/2018:::::M