(Manipal University)

<sup>\*\*</sup> II SEMESTER B.S. DEGREE EXAMINATION – APRIL/ MAY 2017

**SUBJECT: DATA STRUCTURES (CS123)** 

(BRANCH: CS & CE)

Wednesday, 3 May 2017

Time: 3 Hours

Max. Marks: 100

- ✓ Answer ANY FIVE full Questions.
- $\checkmark$  Missing data, if any, may be suitably assumed

1A. Explain in brief how the performance of a program is measured?1B. Write an algorithm for evaluation of post fix expression using stack. Also evaluate the following postfix expression using the same algorithm.

1 2 8 \* + 3 -

2A. Write a recursive functions for the following.

- i) To find GCD of two numbers using Euclid's technique.
- ii) Tower of Hanoi problem.

2B. What is a linear queue? Implement a linear queue with constructor, destructor, insert, deletion and display along with isFull and isEmpty methods. Show the method invocation in main().

(10M + 10M)

(10M + 10M)

3A. Write the following functions for the class list (singly linked list) .

i) int list:: isPresent ( int data) { } to search for a node with data as info field.

ii) void list ::ins\_at\_front( int new\_data) { }

3B. Write a member function to find intersection of two unsorted linked lists with the signature, list list :: getIntersection (list 12) { ...} using the functions defined in question 3A.

(12M + 8M)

4A. What are the merits of doubly linked list over singly linked list ?

4B. Write the following member functions

- i) To reverse a doubly linked list.
- ii) To count number of node of a list.
- 4C. What is Big oh notation?

5A. A Define the following terms with examples.

- i) Binary tree
- ii) Strict binary tree
- iii) Complete binary tree
- iv) Almost complete binary tree

(4M + 10M + 4M)

- 5B. ) What is an expression tree? Write a function to create a binary tree for the given postfix expression. (12M + 8M)
- 6A. Write complete C++ code to implement multiple Stack.
- 6B. Write a C++ function to delete largest element of Binary Search tree.

(12M + 8M)

7A. Construct a Binary search tree for the list of alphabets given below by taking the first element as root: J, R, D, G, T, E, M, H, P, A, F, Q Also write down the output of inorder and preorder traversal for the above tree.

7B. Give the function for Quick Sort. Also mention the time complexity of Quick Sort. Trace the function for the following input values: 45, 26, 27, 70, 14, 90

(8M + 12M)

- 8. Write short notes on the following:
- A. Adjacency Matrix
- B. Depth First Search
- C. Hashing.
- D, Non Linear data structure

(4M + 4M + 4M + 4M)

##