

Roll No.-----

**Bharati Vidyapeeth's
Institute of Computer Applications and Management
A-4, Paschim Vihar, New Delhi-63.**

MCA – 2nd Semester (2016-19)

Internal Test

Data & File Structures – MCA-102

Note: All the questions are compulsory

Max. Marks: 45

Max. Time: 02 Hrs.

Q1 Answer any 10 of the following briefly:

[1.5x10]

- i) What is the difference between applications of the functions declared as firstFunc (struct Node * n) and secondFunc (struct node **n) respectively?
- ii) Consider an implementation of a singly linked list with a head and a tail pointer. State the operations that could be implemented in O(1) time?
- iii) What is the postfix representation of the expression: $(12 - a) * (b + 9) / (d * 4)$
- iv) A circular queue is implemented using an array such that ENQUEUE and DEQUEUE operations are performed efficiently. State the time complexity of insert and delete operations?
- v) How do the height and width of a B-tree compare to those of a BST? What is the impact of the change?
- vi) What makes an AVL tree more efficient as compared to a BST?
- vii) What data structure would you mostly likely see in a nonrecursive implementation of a recursive algorithm? Why?
- viii) How can a search tree be processed in order to get the information stored in descending order?
- ix) Which data structure would you use to enable fair sharing of a resource among multiple consumers?
- x) If a node in a BST has two children, then comment on the nature and count of child nodes of its inorder predecessor.
- xi) Traversal order for cloning a tree should necessarily be pre-order, why?

Q2. Answer any three of the following:

[3x3]

- a) Assume that a linked list is given to you as a pre-compiled library. Implement push and pop operations for a dynamic stack using the given library.
- b) Consider the following pseudo code. Assume that IntQueue is an integer queue. Dry run and explain the working of the function Q2B?

```
void Q2B(int n)
{
    IntQueue q = new IntQueue();
    q.enqueue(0);
    q.enqueue(1);

    for (int i = 0; i < n; i++)
    {
        int a = q.dequeue();
        int b = q.dequeue();
        q.enqueue(b);
    }
}
```

```
        q.enqueue(a + b);
        ptint(a);
    }
}
```

- c) Write a function to compute height of a BST.
- d) Given an expression in postfix form, write pseudo code for building an expression tree.

Q3. Attempt any three of the following: [3x3]

- a) Explain the need of multi-stacks and multi-queues with the help of example situations.
- b) Describe, how can stacks be used to check balancing of parenthesis in an arithmetic expression. How would you identify different error conditions?
- c) Explain, with help of a diagram, the steps involved in resolving LR imbalance in an AVL tree.
- d) State and demonstrate the rules that govern the insert operation in a 5-way B-tree when the keys 12, 2, 34, 45, 3, 10, 15, 7, 22, 37, 42, 44, 36 are inserted in the given order.

Q4. Attempt any two of the following: [6x2]

- a) Write a function to add a node to a linked list ensuring that the data is always sorted in ascending order.
- b) Assuming that the required library for linked list is already available, define the function *buildPolynomial* that adds a term to a polynomial.
- c) Write functions add node to a right in-threaded tree.

***** All the Best *****