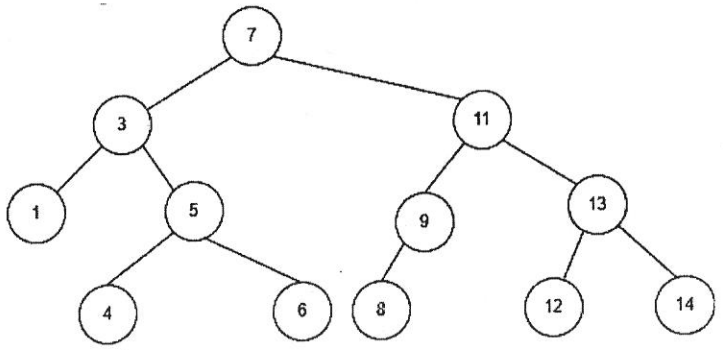
7. Consider the following Binary Search Tree and answer the following:



(a) Write the breadth-first traversal of the above BST (4)

(b) Draw the tree after inserting the node with key 10 in above BST. (5)

(c) Write the post-order traversal and in-order traversal of the resultant tree. Is the resultant tree height-balanced tree? Justify your answer. (6)

[This question paper contains 8 printed pages.]

Your Roll No.....

Sr. No. of Question Paper : 4659 **J**

Unique Paper Code : 2344000024

Name of the Paper : Data Structures Using Python

Name of the Course : **Generic Elective**

Semester : IV/VI

Duration : 3 Hours Maximum Marks : 90

Instructions for Candidates

1. Write your Roll. No. on the top immediately on receipt of this question paper.
2. **Section A** is compulsory.
3. Attempt **any 4** questions from **Section B.**
4. Parts of the question must be answered together.

SECTION A

1. State True/False for the following statements and justify your answer (3)

   (i) Stack uses FIFO access method.

(ii) A doubly linked list uses more memory than the circular linked list.

(iii) In-order traversal of Binary Search Tree gives a sorted sequence.

Which is the most appropriate data structure that can be used in following situations: (3)

(i) When you need to implement a browser's back and forward navigation.

(ii) When you need to store data with hierarchical relationships.

(iii) When you need to store data in Last-In-First-Out manner.

What is AVL tree? Explain with the help of suitable example Give one application. (3)

How a priority queue is different from a normal queue? Write one application of priority queue. (3)

Write a code snippet in Python to insert an element at the front of an already existing singly linked list. (3)

(b) Write a program in Python to compute the sum of 'n' numbers of a list using recursive function. (5)

(c) Arrange the following functions in the increasing order of their asymptotic complexity: (4)

$f_1(n) = 2^n$
$f_2(n) = n^{2/3}$
$f_3(n) = n \log n$
$f_4(n) = \log n$

6. (a) Use recurrence tree method to solve the following recurrence relation: (4)

$T(n)\ T\ (n\text{-}1) + n$

(b) Solve the following recurrence relation: (5)

$T(n) = 2T\ (n/4)+n^2$

(c) Explain what Big O notation represents. Show the function $8n+5$ is O(n) (6)

4. (a) Consider a heap of height h. What will be the maximum and minimum number of elements in the heap? (4)

(b) Create a binary tree whose following traversals are given: (5)

Inorder: x y z a p q r

Preorder: a y x z q p r

(c) Construct the max-heap for the given array A given as (6)

A = [4,1,3,2,16,9,10,14,8,7]

Show all the steps clearly.

5. (a) Convert the given iterative function to recursive function: (4)

```
def power (base, exponent):
        result = 1
        for i in range(exponent):
            result *= base
        return result
```


Explain any two properties of binary heap. How is it different from a binary search tree? (3)

Implement a Python function that counts the number of nodes in a circular linked list. (4)

Write a program in Python to print Fibonacci numbers via Binary recursion. (4)

Differentiate between Big Oh, Big Ω (Omega) and Big θ (theta) notations. (4)

2. (a) Convert the following infix expression to a postfix expression. A* (B+D)/E – F* (G+H/K)

Show all the steps clearly. (4)

(b) Consider the following sequence of operations performed on a stack S of size 5. Show the return value and contents of stack after each operation: (5)

S.push(5)

S.push(3)

S.pop( )

S.is_empty()

S.pop()

S.push(9)

S.top( )

S.push(4)

S.push(15)

len (S)

(c) What is an abstract data type? (6)

Write the time-complexity for the following operations:

(i) The push operation of stack implemented using linked list.

(i) The pop operation of stack implemented using linked list.

(iii) To delete all the elements from the end in a singly linked list with n elements.

(iv) To search an element in Balanced Binary Search Tree (BST) in best case scenario.

3. (a) Suppose an initial empty queue Q has executed a total of 32 enqueue operations, 10 first operations and 15 dequeue operations, 5 of which raised Empty errors that were caught and ignored. What is the current size of Q? (4)

(b) What are the drawbacks of using linear queue? How can it be resolved through circular queue? (5)

(c) Consider an initially empty circular queue of the size 4 implemented using arrays. Perform the sequence of operations and show the position of front and rear after each operation. (6)

enqueue (14)

dequeue( )

dequeue( )

enqueue (3)

enqueue (7)

enqueue (9)

enqueue (0)

enqueue (2)