(c) Construct a max heap for the given array A of elements,

A = [2, 4, 8, 1, 12, 11, 15, 6, 3].

Also, report the number of comparisons. (5)

[This question paper contains 8 printed pages.]

Your Roll No.....

Sr. No. of Question Paper : 2934 **L**

Unique Paper Code : 2342571201

Name of the Paper : Data Structures

Name of the Course : **B.A. (P) With CS as Minor/ Non-Major, B.Sc (Phy. Sc.)/ SOL**

Semester : II

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 **four** questions from **Section B**.
4. Parts of a question must be answered together.

Section A

1. (a) Differentiate between an Array and a Linked List. (2)

(b) Show that the Run Time Complexity of the following function: (3)

$3n\log_{10}n + 4n + 5\log_{10}n$

is $\theta(n\log_{10}n)$

(c) Sort the following given array: (3)

A = [6, 4, 12, 0, 1, 7, 9, 4] using Insertion Sort.

Show the steps after each iteration.

(d) Write a recursive function in C++ to find the factorial of a number. (3)

(e) Explain the significance of balancing the Binary Search Tree. Explain with the help of a suitable example. (3)

6. (a) Give the recursive functions for Inorder and Preorder traversal in a Binary search Tree. (6)

(b) Give the time complexity of the following basic operations in a Queue implemented using Arrays:

(i) Enqueue()

(ii) Dequeue()

(iii) Empty()

(iv) Full() (4)

(c) What is an AVL tree? Why is the AVL tree called a self-balancing tree? Explain LR rotation used for balancing. (5)

7. (a) Explain the concept of Tail recursion and write a Tail recursive function. (6)

(b) Write a function to find the length of the linked list. (4)

(v) Is the resultant tree a Complete Binary Tree? Justify your answer.

5. (a) Write a C++ program to create a class DList implementing a Doubly Linked List with appropriate data members. Declare and define the following functions :

(i) Insert values in the list.

(ii) Delete a value from the i^{th} position in the list. (6)

(b) Write a function to search an element x in a Singly Linked List. (4)

(c) Consider the following arithmetic expression:

[P/Q – (X *· Y) + (R %S)]

Use stack to check if the above expression is valid i.e. it contains nested parenthesis and brackets. Show the contents of the stack at each step. (5)

(f) Write a function in C++ to insert an element at the end of a Doubly Linked List. (3)

(g) Suppose a Circular Queue of capacity n elements is implemented with an array of n elements. Assume that the insertion and deletion operations are carried out using FIRST and LAST as array index variables respectively. Initially FIRST = LAST = –1. Write the conditions to detect if the queue is full and empty. (3)

(h) What are the different operations that can be performed on a Deque. Explain using a suitable example. (5)

(i) Describe the output of the following series of Stack operations:

push(7), push(2), pop(), push(5), pop(), top(), pop(), pop(), push(8), pop().

Show the contents of the Stack after each operation. (5)

Section B

2. (a) Write a code snippet for Linked List representation of Stack. Write the necessary class definitions and define the following functions: Push(), Pop(), Display(), Empty(), Full(). (6)

(b) Write the function to reverse the elements of an Array. (4)

(c) Solve the following recurrence relation using Master Theorem: (5)

$$T(n) = 2T(n/2) + n\log_2 n$$

3. (a) Differentiate between a Linked List and Circular Linked List with the help of a suitable example. Also, write a function to add a node at the end of a Circular Linked List. (6)

(b) Draw all possible Binary Search Trees for the three elements A, B and C. (4)

(c) Calculate the Run Time Complexity for the following given recurrence relation using the recursion tree method : (5)

$$T(n) = 2T(n/2) + cn$$

4. Consider the following sequence of elements: (15)

P, H, J, I, C, F, G, L, T, S, W, Y

(i) Construct the Binary Search Tree by inserting the elements one at a time in the order given above.

(ii) Write the Inorder traversal of the constructed tree.

(iii) Give the Height of the above tree. Also write its Postorder traversal.

(iv) Delete node with value 'H' from the tree and draw the resultant tree.