

B2.1-R4: DATA STRUCTURE THROUGH C++

NOTE:

IMPORTANT INSTRUCTIONS:

1. There are **TWO PARTS** in this Module/Paper. **PART ONE** contains **FOUR** questions and **PART TWO** contains **FIVE** questions.
2. **PART ONE** is to be answered in the **OMR ANSWER SHEET** only, supplied with the question paper, as per the instructions contained therein. **PART ONE** is **NOT** to be answered in the answer book.
3. Maximum time allotted for **PART ONE** is **ONE HOUR**. Answer book for **PART TWO** will be supplied at the table when the answer sheet for **PART ONE** is returned. However, candidates, who complete **PART ONE** earlier than one hour, can collect the answer book for **PART TWO** immediately after handing over the answer sheet for **PART ONE**.

TOTAL TIME: 3 HOURS

TOTAL MARKS: 100
(PART ONE – 40; PART TWO – 60)

PART ONE

(Answer all the questions)

1. **Each question below gives a multiple choice of answers. Choose the most appropriate one and enter in the “OMR” answer sheet supplied with the question paper, following instructions therein. (1x10)**
 - 1.1 A graph has M vertices and N edges. How many edges does a Minimum Spanning Tree has?
 - A) $(N + M) - 1$
 - B) $N - 1$
 - C) $M - 1$
 - D) $N \times M$
 - 1.2 A sequence of operations performed on the stack of the size 4 is as follows: push(5), push(3), pop, push(4), push(6), push(7), pop, push(8), pop, push(1) What is the condition of the stack after all these operations?
 - A) Stack overflow message is displayed.
 - B) Stack finally contains: 5, 3, 4, 6, 7, 8, 1
 - C) Stack finally contains: 5, 4, 6, 1
 - D) Stack finally contains: 5, 4, 8, 1
 - 1.3 Which is a linear data structure?
 - A) Binary search tree
 - B) Heap tree
 - C) Directed graph
 - D) None of the above
 - 1.4 Creation of BST using sequence: 15, 19, 25, 32, 37, 45, 66, 89, 90, 98 results in
 - A) Tree with root 15 and all other nodes in right subtree of it.
 - B) Tree with root 15 and all other nodes in left subtree of it.
 - C) Tree with root 15 and complete binary tree.
 - D) Tree with root 15 and height balanced tree.
 - 1.5 In C++, *cin* and *cout* are
 - A) Actually pre-defined functions of iostream header file
 - B) The objects or iostream class
 - C) The example of polymorphism in iostream
 - D) The member functions of iostream class

- 1.6 Which method of hashing guarantees no collision in any case?
- Folding method
 - Midsquare method
 - Digit Analysis method
 - None of the above
- 1.7 Which one is false for C++ language?
- A variable can be declared anywhere in the program before its use.
 - A class contains both data as well as functions.
 - Memory can't be allocated to an object at run time.
 - Address of an object can't be stored in a pointer.
- 1.8 A set of disjoint trees is known as
- Height Balanced
 - Null graph
 - Minimum Spanning Tree
 - Forest
- 1.9 What is the final answer after evaluation of the following postfix expression
 $2\ 3\ +\ 4\ *\ 5\ 3\ 2\ 4\ *\ +\ /\ +\ 5\ +$
- 25
 - 35
 - 100
 - Invalid postfix expression
- 1.10 In a linked list, each node contains two fields. One is data and another is next node's location. Which pointer represents next node's location?
- Pointer to character
 - Pointer to node
 - Pointer to class
 - Pointer to head
- 2. Each statement below is either TRUE or FALSE. Choose the most appropriate one and ENTER in the "OMR" answer sheet supplied with the question paper, following instructions therein. (1x10)**
- 2.1 Sparse array has high-density of non-zero elements.
- 2.2 Deque is a general representation of both stack and queue.
- 2.3 Replacing null pointer in the last node of a list with the address of its first node creates doubly linked list.
- 2.4 Evaluation of postfix expression is faster than evaluation of same infix expression.
- 2.5 A single leaf is also a null graph.
- 2.6 In C++, run time memory is allocated by *new* operator.
- 2.7 A collision occurs when two same data have the distinct hash value.
- 2.8 For any algorithm, if time complexity increases space complexity decreases.
- 2.9 The time complexity of binary search is $O(n \log n)$ in worst case.
- 2.10 Stack works in First in last out and Queue works in Last in First out.

3. Match words and phrases in column X with the closest related meaning/ word(s)/phrase(s) in column Y. Enter your selection in the “OMR” answer sheet supplied with the question paper, following instructions therein. (1x10)

X		Y	
3.1	An operator in C++ for new line	A.	stack
3.2	Data structure used in recursion	B.	F=0
3.3	No of comparison in selection sort for n items	C.	zero
3.4	In a binary tree, maximum in-degree of any leaf node	D.	$n \times (n-1)/2$
3.5	This searching technique needs data in sorted order	E.	queue
3.6	The value of TOP of stack with n size when a stack is underflow	F.	eight
3.7	No of leaf nodes in a complete binary tree with 15 nodes	G.	Deque
3.8	Breadth first search uses data structure	H.	one
3.9	Maximum difference between height of sub-trees in AVL tree	I.	endl
3.10	A condition for underflow of a circular queue i	J.	binary
		K.	two
		L.	four
		M.	F=R

4. Each statement below has a blank space to fit one of the word(s) or phrase(s) in the list below. Enter your choice in the “OMR” answer sheet supplied with the question paper, following instructions therein. (1x10)

A.	inorder	B.	$O(n \log_2 n)$	C.	$O(n^2)$
D.	One	E.	Three	F.	bubble
G.	linear probing	H.	Two	I.	zero
J.	$R \geq N$	K.	private	L.	loop
M.	division				

- 4.1 _____ traversal of a binary search tree visits nodes of a tree in ascending order.
 4.2 Quick sort in worst case complexity _____.
 4.3 Number of pointers required in a node of doubly linked list _____.
 4.4 The condition to make the simple queue overflow with size n is _____.
 4.5 _____ is one of the collision resolution techniques of hashing.
 4.6 _____ sort exchanges values whenever pairs of adjacent values are out of order.
 4.7 _____ is used to declared secure data members in a class in C++.
 4.8 An isolated node in a graph has out-degree _____.
 4.9 In adjacency matrix, all the elements of main diagonal contains zero, then that graph does not contain any _____.
 4.10 Average case time complexity of Two-way merge sort is _____.

PART TWO
(Answer **ANY FOUR** questions)

5.

- a) Define the terms: (i) Encapsulation, (ii) Polymorphism, (iii) Abstract Data Type (ADT).
- b) Arrange the given numbers in ascending order using radix sort.
9, 1106, 9099, 6, 7101, 990, 15, 99, 10, 909
- c) For given array $Z[-4:-1, 10:13, -1:1]$ find the total number of elements. Assume that the base address is 5001 and each element required 2 bytes. Find address of $Z[-2,12,0]$ element if it is stored in (a) row major order (b) column major.

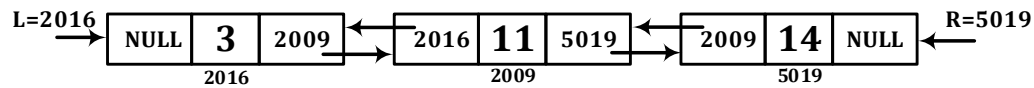
(6+4+5)

6.

- a) Convert the given expression into its equivalent postfix expression using stack. Show the contents of stack for each step.

$$(A + B) * D + E / (F + A * D * E) + C$$

- b) For the given doubly linked list perform the following operations such that it maintain ascending order of data. Draw linked list after each operation.

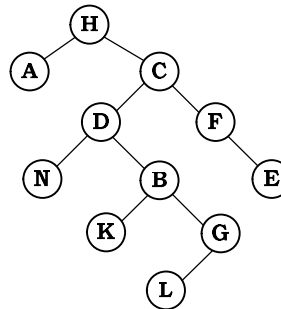


1. Insert 10 having node address 4092.
2. Insert 15 having node address 3014.
3. Delete 14.

(8+7)

7.

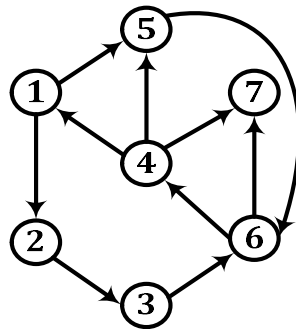
- a) Create Binary Search Tree using 90, 31, 75, 42, 67, 55, 62, 58, 65, 64. After creation of BST, delete 62 and draw final tree.
- b) Arrange the following data in the array using hash function $H(x)$. Calculate hash key and use linear probing to resolve collisions. Array size is 8 (index starts from 0)
Data: 214, 4563, 789, 2477, 1579, 4566, 1000, 6745
Hash function $H(x) = x \text{ modulo } 5 + 1$
- c) Traverse the following tree in Preorder, Inorder and Postorder.



(6+6+3)

8.

- a) Traverse the graph shown in the following figure using Breadth First Search strategy. Starting node is 1. Prepare adjacency matrix for the graph before traversal. Also show the step by step contents of the data structure used in traversal.



- b) Create AVL Tree for the given numbers. Show the tree after each operation.
7, 1, 6, 2, 5, 4, 3.

Apply Delete (1) and then Insert (9).

- c) What are the advantages of using linked list over array?

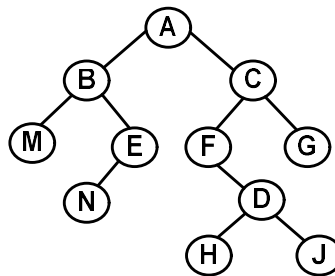
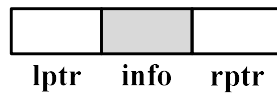
(6+6+3)

9.

- a) Prepare the max heap for the following data. Perform descending order sorting for the data using Max Heap Tree: 103, 11, 101, 110, 111, 119, 19, 91.

- b) Explain the priority queue concept.

- c) Assign the threads to the NULL links of given binary tree for **inorder** traversal. Redraw the tree with head node using following structure of a node.



(8+3+4)