C2-R3: DATA STRUCTURE THROUGH 'C' LANGUAGE

NOTE:

1. There are **TWO PARTS** in this Module/Paper. **PART ONE** contains **FOUR** questions and **PART TWO** contains **SIX** questions.

2. **PART ONE** is to be answered in the **TEAR-OFF ANSWER SHEET** only, attached to 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 "tear-off" answer sheet attached to the question paper, following instructions therein. (1 x 10)

1.1. Adjacency matrix for a digraph is

A) Unit Matrix
B) Symmetric Matrix
C) Asymmetric matrix
D) none of the above

1.2. Time complexity of insertion sort algorithm in the best case is

A) O(n)
B) $O(n \log_2 n)$
C) $O(n^2)$
D) none of the above

1.3. The prefix expression for the infix expression a * (b + c) /e - f is

A) /* a + bc – ef
B) -/ * + abc ef
C) - / * a + bcef
D) none of the above

1.4. In linked list representation, a node contains at least

A) node address field, data field
B) node number, data field
C) next address field, information field
D) none of the above

1.5. Number of nods in a complete binary tree of depth k is

A) 2^k
B) 2k
C) 2^k-1
D) none of the above

1.6. The smallest number of keys that will force a B-Tree of order 3 to have a height 3 is:

A) 12
B) 10
C) 7
D) none of the above

1.7. Given two sorted lists of size “m” and “n” respectively, the number of comparisons needed in the worst case by the merge sort will be:

A) m*n
B) max(m,n)
C) min(m.n)
D) m + n – 1

1.8 Any string of bits of length *n* represents a unique non-negative integer between

A) 0 and 2^{n-1} -1
B) 0 and 2^{n-1}
C) 1 and 2^{n-1}
D) none of the above

1.9 Which of the following expressions accesses the $(i,j)^{th}$ entry of a $m \times n$ matrix stored in a column major form ?

A) m * (i –1) + j
B) m * (j –1) + i
C) m * (n – j) + i
D) m * (m – i) + j

1.10 The following sequence of operation is performed on a stack
`push(1), push (2), pop, push(1), push(2), pop, pop, pop, push(2), pop.`
The sequences of popped out values are

A) 2, 2, 1, 2, 1
B) 2, 2, 1, 1, 2
C) 2, 1, 2, 2, 1
D) 2, 1, 2, 2, 2

2. Each statement below is either TRUE or FALSE. Choose the most appropriate one and ENTER in the "tear-off" sheet attached to the question paper, following instructions therein. (1 x 10)

2.1 All primitive recursive functions can be solved iteratively.
2.2 Breadth – first search algorithm can only be used for undirected graph.
2.3 De-referencing operator * has the same effect when it is applied to a pointer or to a structure.
2.4 Binary search performs efficiently on a linked list.
2.5 A symbol table can be constructed using binary tree.
2.6 A pre-order and post-order traversal sequence uniquely defines a tree.
2.7 If an undirected graph is of "n" vertices and "e" edges then the sum of degrees of all vertices is 2e.
2.8 The adjacency matrix corresponding to a graph consisting of "n" nodes but no edges is a unit matrix.
2.9 Recursion cannot be removed without using a stack.
2.10 Pointers are used for dynamically allocated memory.

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 "tear-off" answer sheet attached to the question paper, following instructions therein. (1 x 10)

X		Y	
3.1	Order notation	**A.**	Stack
3.2	Pattern matching	**B.**	Dynamic storage allocation
3.3	Dynamic data structure	**C.**	Two's compliment
3.4	Height Balanced Trees	**D.**	Primitive data type
3.5	Data structure for back tracking	**E.**	Finite automata
3.6	Circular list with "head" pointing to the last node	**F.**	Adjacency list
3.7	Suitable data structure for efficiently computing adjacent vartices of a given vertex	**G.**	Adjacency matrix
3.8	Partition exchange sort	**H.**	AVL Trees
3.9	Stable sorting Algorithm	**I.**	Adjacency multi list
3.10	Unique representation for zero	**J.**	Queue
		K.	Tree
		L.	Breadth – first
		M.	Depth – first
		N.	Quick sort
		O.	merge sort
		P.	Complexity measurement

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 "tear-off" answer sheet attached to the question paper, following instructions therein. (1 x 10)

A.	circular queue	**B.**	stack	**C.**	simple queue
D.	band	**E.**	sparse	**F.**	many-to-many
G.	one-to-many	**H.**	depth-first-search	**I.**	breadth-first search
J.	$\log_2 n$	**K.**	$n \log_2 n$	**L.**	E / (n + 1)
M.	Insertion	**N.**	$\text{Log}_2 n + 1$	**O.**	recursive
P.	Compaction	**Q.**	N		

4.1 Matrices in which non-zero entries tend to cluster around the middle of each row are called _____ matrix.

4.2 A graph represents _____ relationship between nodes.

4.3 _____ can be used to find the shortest distance between given two nodes in a graph.

4.4 Time complexity of inserting an element to a heap of "n" elements is of the order of _____.

4.5 If "E" is the total external length of a binary tree with "n" nodes then average number of comparisons for unsuccessful search is _____.

4.6 _____ data structure may give overflow error, even though the current number of elements in it is less than its size.

4.7 Conversion of infix arithmetic expression to prefix expression requires the use of _____.

4.8 The minimum number of edges in a connected cyclic graph of "n" vertices is_____.

4.9 Linked list is preferred over an array for _____ operation.

4.10 Recursion often provides elegant solution to programming task but _____ function chew up a lot of memory.

PART TWO

(Answer any **FOUR** questions**)**

5.

a) What are the limitations of array data structure?
b) Show with the help of an example, how the above limitations can be avoided by "linked" representation of lists?
c) Show the representation of linked lists using arrays.
d) What do you mean by linear list and generalized list?

(4+4+4+3)

6.

a) Write a "C" function to copy one stack to another assuming the stack is implemented using array.
b) Write an algorithm to evaluate Postfix expression with the help of a stack.

(8+7)

7.

a) Write a "C" function to implement the Knuth Mooris Pratt algorithm for string searching.
b) A function A(m,n) is defined as follows.

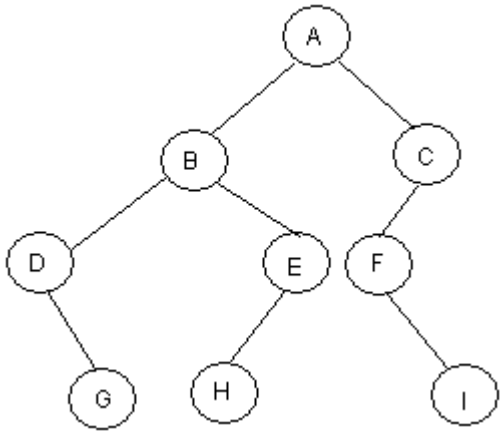
$$A(m,n) = \begin{cases} n + 1, & \text{if } m = 0 \\ A(m-1, 1), & \text{if } n = 0 \\ A(m-1, A(m, n-1)), & \text{otherwise} \end{cases}$$

write a recursive "C" function to compute A(m,n)

(8+7)

8.

a) Prove that the maximum number of nodes in a binary tree of height "k" is $2^{k+1}-1$.
b) Consider the following binary tree



Indicate the output in the following cases: -

i) When the tree is traversed in an "inorder" fashion.
ii) When the tree is traversed in an "preorder" fashion.
iii) When the tree is traversed in an "Postorder" fashion.

(6+[3×3])

9.

a) Show with the help of an example that the simple selection sort is not data sensitive.

b) Find the time complexity of the simple selection sort.

c) Write a recursive “C” function to implement binary search.

(6+3+6)

10.

a) Write a “C” function to compute the in-degree and out-degree of a vertex of a directed graph when the graph is represented by an adjacency list.

b) What is a spanning tree? What do you mean by minimal spanning tree?

(9+[3+3])