

GUJARAT TECHNOLOGICAL UNIVERSITY**MCA. Sem-II Remedial Examination December 2010****Subject code: 620001****Date: 16 /12 /2010****Subject Name: Data Structures****Time: 10.30 am – 01.00 pm****Total Marks: 70****Instructions:**

1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.

- Q.1 (a)** What is Heap? Describe Heap Sort Algorithm and write its pseudo code. You can assume that CREATE_HEAP(K, N) function is available **07**
- (b)** Given a vector K, consisting of N elements arranged in ascending order, write the pseudo code for the Iterative Binary Search Algorithm to search a given element X. For this algorithm, give the Order of Running Time and also the Order of Storage requirements. **07**

- Q.2 (a)** Differentiate between Stack and Queue. Convert the following Infix expression to the corresponding Reverse Polish expression: **07**

$$(a + b \uparrow c \uparrow d) * (e + f / d)$$

Give the trace of the steps including Stack Contents and Rank in tabular form.

- (b)** Write the pseudo code for the Algorithms for (i) Inserting a given element at the rear of a Circular Queue, and for (ii) Deleting and returning the last element from a Circular Queue. **07**

OR

- (b)** Write a short note on Array, and Storage Structure for Arrays. Consider a 3-dimensional array X whose subscripts limits are: **07**

$$1 \leq i \leq 3; 1 \leq j \leq 4; -1 \leq k \leq 0$$

Give the Addressing Function for the element $X[i, j, k]$ where the storage representation is in Row-major order. Justify your answer briefly.

- Q.3 (a)** Describe briefly Singly Linked List and a typical Node Structure used for it. What will be a Node Structure for a linked-list representation of a polynomial in three variables. Write the pseudo code of an Algorithm to Insert a Node at the end of a Singly Linked List. **07**

- (b)** Write pseudo code for the algorithm INSEND(X, FIRST) to insert a node at the end of the List. X is a new element, and FIRST is a pointer to the first element of a linked linear list whose typical node contains INFO (i.e. information field) and LINK (i.e. pointer) fields. **07**

OR

- Q.3 (a)** Describe briefly Doubly Linked List and a typical Node Structure used for it. Is it same as Circularly Linked Linear List? Justify. Write the pseudo code of an Algorithm to Delete a Node from Doubly Linked List whose address is specified. **07**

- (b)** Represent the following polynomial using the linked-list representation of a polynomial in three variables: **07**

$$f(X, Y, Z) = 3X^3$$

Represent first each of the following terms using the linked-list representation of a

polynomial and then give the (logical) steps to be followed if these terms are to be added one at a time at appropriate position in the latest $f(X, Y, Z)$ polynomial:

(i) $+ 2XY$, (ii) $+ 7X^2$, (iii) $+ XYZ$, (iv) $+ 4Y^2$

Rules for appropriateness of the position of a term:

- The priority of variable X is the highest while the priority of variable Z is the lowest. Thus,
- The terms containing X will appear first and in such terms, the power of X cannot be higher than that of the previous term. In case the power of X is same as or lower than in the previous term, the power of Y cannot be higher than that of the previous term. Similarly, if the power of X as well as the power of Y are either same or lower than the corresponding powers in the previous term, the power of Z cannot be higher than that of the previous term.

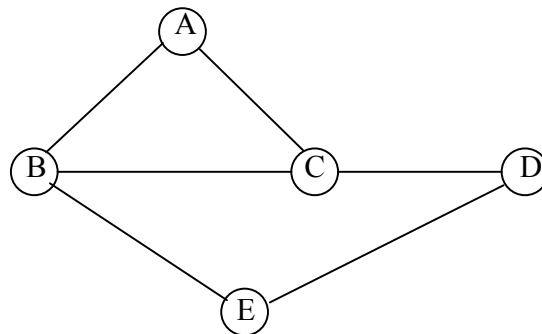
Q.4 (a) (i) Describe the basic characteristics of Linear and Non-linear Data Structures. Write the difference between the two types of structures. **03**

(ii) Construct an Expression Tree for the following expression: **04**

$(a + b) * c - d / (e + f) * g / h$

State the result of Pre-order, In-order, and Post-order Traversal of this tree.

(b) What are the (i) Node Table Directory Structure, and (ii) Edge Structure used in Breadth-First Search (BFS). Using these structures, display the storage representation of the following graph (Assume initial value of REACH to be false, that of DIST to be 0): **07**



Write the trace of the procedure BFS(INDEX) for the above graph taking INDEX = 1 (allotted to Node A).

The pseudo code for the procedure BFS(INDEX) is give below:

- [Initialize the 1st node's DIST number and place node in queue]
 $REACH[INDEX] \leftarrow true$
 $DIST[INDEX] \leftarrow 0$
 Call QINSERT(Queue, INDEX)
- [Repeat until all nodes have been examined]
 Repeat thru step 5 while queue is not empty
- [Remove current node to be examined from queue]
 Call QDELETE(Queue, INDEX)
- [Find all unlabeled nodes adjacent to current node]
 $LINK \leftarrow LISTPTR[INDEX]$
 Repeat step 5 while $LINK \neq NULL$
- [If this is an unlabeled node, label it and add it to the queue]
 If not $REACH[DESTIN(LINK)]$
 Then $DIST[DESTIN(LINK)] \leftarrow DIST[INDEX] + 1$
 $REACH[DESTIN(LINK)] \leftarrow true$
 Call QINSERT(Queue, DESTIN(LINK))

LINK ← EDGEPTR(LINK) (Move down edge list)

6. [Finished]

Return

The procedure QINSERT(Queue, INDEX) inserts the INDEX value at the end of Queue, while the procedure QDELETE(Queue, INDEX) deletes the 1st element of Queue and returns the INDEX value.

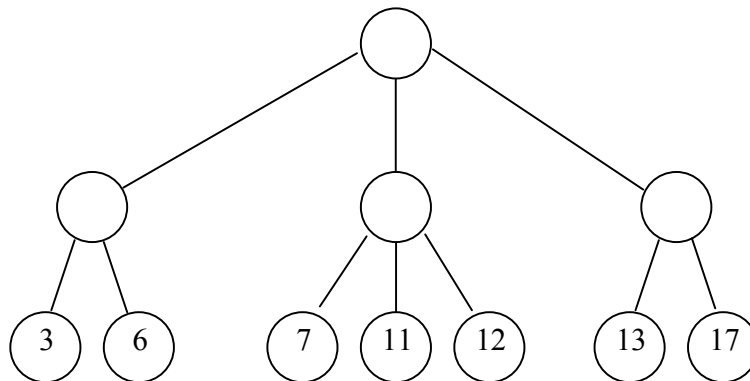
OR

- Q.4 (a)** (i) Define the asymptotic notation Big-Oh (O). State briefly its significance in Algorithm Analysis with an example. **03**
 (ii) What are the characteristics of Sparse Matrices? Is the following matrix a Sparse matrix? **04**

$$\begin{bmatrix} 0 & 0 & 0 & 4 & 0 & 0 & 1 & 0 \\ 9 & 0 & 0 & 0 & 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 6 & 7 & 8 & 0 & 0 & 0 & 0 \\ 20 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 5 & 7 & 0 & 0 & 0 & 13 & 0 & 0 \end{bmatrix}$$

Represent the above matrix using two types of sequential representations so as to store only non-zero elements.

- Q.4 (b)** Write the pseudo code for recursive algorithm of Depth-First Search (DFS). Give the trace of DFS (1, 0) for the first two (2) recursive calls of DFS for the graph shown in the Main part of Q. 4 (b) above. **07**
- Q.5 (a)** What is 2-3 Tree? Give the structure of Leaf Nodes and Non-leaf Nodes used in 2-3 Tree. Fill-up the contents in each non-leaf node in the following 2-3 Tree: **07**



How the above 2-3 Tree will be transformed if the number 10 is inserted as a new leaf node? Trace (write) each intermediate step with tree drawn for that step.

- (b)** What is the major application and utility of Hashing Function? What problem is encountered if the key space is large and how this problem is resolved? Describe the Division Method of Hashing Function. Using the Division Method with $m = 101$, obtain the hash values for the following set of two keys (Use Pre-conditioning of the keys by using 11, 12, ..., 36 for letters A, B, ..., Z): **07**
 (i) PAY (ii) AGE

OR

- Q.5 (a)** Describe briefly Trie Structure. List down all the three-letter sequences built from a, b, c beginning with the first letter as b. Take from this list of sequences one sequence at a time and tabulate a Trie structure in step-by-step manner. (Hint: Take Null character, i.e. ^ in addition to a, b, c.) **07**
- (b)** In a Height-Balanced Tree, what are the cases causing imbalance and how rebalancing is done in such cases. Write a step-by-step method (taking one number at a time) to develop Height-Balanced Tree (by rebalancing at each stage in case of imbalance) for the following sequence of numbers: **07**
- 12, 6, 9, 11, 10, 4, 7
