DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A,**
ANY TWO *from* **Group B** *and* ALL *from* **Group C.**

*All parts of a question (a,b,etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.*

Figures on the right-hand side margin indicate full marks.

Group A

1. (*a*) Write a C/C++ function to perform *linear search* on *n* data elements for a given key *k*. What are the worth case and average case time complexities ?　　10

   (*b*) Write a recursive C/C++ function to perform *binary search* on *n* data elements for a given key *k*. What are the worst case and the best case time complexities ?　10

2. (*a*) Suppose we store a two dimensional $n \times n$ array *A* in *row major* fashion in an one-dimensional *B* array of size n^2. What will be the index of the array element $A[i, j]$ in the array *B* ? If we store a two-dimensional $n \times n$ array *A* in *column major* fashion in an one-dimensional *B* array of size n^2, what will be the index of the array element $A[i, j]$ in the array *B* ? Explain your answer.　10

(b) Write a C/C++ function to *reverse* a singly linked list. You can traverse the list exactly once and can use only constant, *i.e.*, O(1) additional storage. Briefly explain your answer. 10

3. (a) Write a recursive C/C++ function to count the number of nodes in a linked list. 10

(b) Write a C/C++ function to insert a key at an appropriate place in a sorted linked list. Using this function, write the main or the driver function to sort n data elements. 10

4. (a) Write a C/C++ function to *insert* a key x to the right of a node pointed by p to a *doubly linked list*. 10

(b) Write a C/C++ function to *delete* a node pointed by p from a doubly linked list. 10

Group B

5. (a) Write a C/C++ function to *delete* an element from the root of a *binary search tree*. 10

(b) Write a C/C++ function to *delete* the smallest element from a *binary search tree*. 10

6. (a) Write a C/C++ function to count the leaves of a binary tree. 10

(b) Show the binary search tree that is obtained after inserting the key 8, 11, 5, 7, 9, 6, 10, 14, 12. Re-draw the tree after deleting the root. 10

7. (a) Construct a *binary search tree* whose post-order traversal is as follows :

$$5, 4, 9, 11, 12, 10, 13, 6$$ 10

(b) Can a *binary tree* be uniquely re-constructed given its pre-order and post-order traversals ? Justify your answer. 10

8. (a) Show the result of inserting 2, 1, 4, 5, 9, 3, 6, 7 into an initially empty AVL tree. Specify the type of rotation after each insertion. 10

(b) Delete the element 65 from the following 2-3 tree, *i.e.*, B-tree of order 3. 10
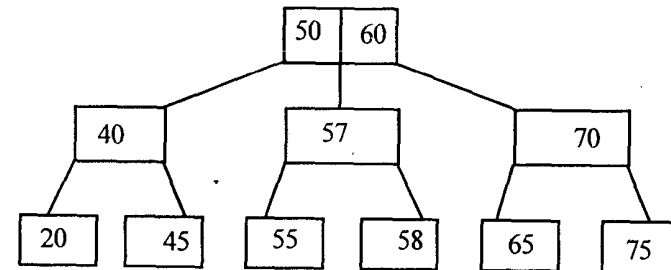
Group C

9. Choose the *correct* answer for the following : 10 × 2

(i) The equivalent postfix expression corresponding to the infix expression (A − B)*(D/E) is

(a) ABDE/*−
(b) AB − D*E/
(c) AB − DE/*
(d) ABD*− E/

(ii) Suppose a queue is implemented by a circular array QUEUE [0..11]. The number of elements in the queue, if FRONT = 10 and REAR = 3, will be

(a) 3
(b) 4
(c) 5
(d) 6

(*iii*) The number of null links in a binary tree with n nodes is

(*a*) $n + 1$

(*b*) $2n$

(*c*) $2n - 1$

(*d*) $n - 1$

(*iv*) Binary search can be performed, if data items are stored in an

(*a*) unordered array.

(*b*) ordered array.

(*c*) unordered linked list.

(*d*) ordered linked list.

(*v*) In an almost complete binary tree with 13 nodes, the number of leaves will be

(*a*) 5

(*b*) 6

(*c*) 7

(*d*) 8

(*vi*) Absence of a terminating condition in a recursive program generates

(*a*) null pointer access.

(*b*) index out of bounds.

(*c*) stack underflow.

(*d*) stack overflow.

(*vii*) Which of the following traversals are sufficient to reconstruct a binary tree uniquely ?

(*a*) pre-order

(*b*) in-order and post-order

(*c*) pre-order and post-order

(*d*) in-order

(*viii*) Assuming the height of a single node is zero, the maximum possible height of a binary tree with n nodes is

(*a*) $\lceil \log n \rceil$

(*b*) $\lfloor \log n \rfloor$

(*c*) $\lfloor \log (n + 1) \rfloor + 1$

(*d*) $n - 1$

(*ix*) The minimum number of nodes in an AVL tree height 10 is

(*a*) 150

(*b*) 200

(*c*) 235

(*d*) 232

(*x*) What is the minimum size of an array required to store an n node binary tree ?

(*a*) n

(*b*) $2n$

(*c*) $2^{n-1} - 1$

(*d*) $2^n - 1$

DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A,**
ANY TWO *from* **Group B** *and* ALL *from* **Group C.**

*All parts of a question (a,b,etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.*

Figures on the right-hand side margin indicate full marks.

Group A

1. (*a*) What is run time complexity of an algorithm ?
   Calculate the run time complexity of bubble sort.   2 + 4

   (*b*) What is asymptotic little O notation (o) ? What is big O
   notation ?   2 + 2

   (*c*) Design an O (n)-time algorithm that, given a real
   number x and a sorted array S of n numbers,
   determine whether or not there exist two elements in S
   whose sum is exactly x.   5

   (*d*) Suppose we have an array of numbers $a[1], ..., a[n]$ in
   which the first i numbers $a[1], ..., a[i]$ have been sorted
   into ascending order, and the remaining numbers
   $a[i+1], ..., a[n]$ have been sorted in descending order.

The aim is to sort the entire array in ascending order. Design an algorithm which merges the two sorted lists into a second array $b[1], ...; b[n]$ and copies the result back into a. Show that the algorithm takes time $O(n)$.   5

2. (a) Let A and B be two linked lists. Write a 'C' function to create a new linked list C that contains elements alternately from A and B beginning with the first element of A. If you run out of elements in one of the lists, then append the remaining elements of the other list to C.   6

(b) Let P be a linked list. Write a 'C' function called split to create two linked lists Q and R. Q should contain all elements in odd positions of P and R contains the remaining elements. Your function should not change list P. What is the complexity of your program ?   5 + 3

(c) What are the limitations of array data structure ? Show, with the help of an example, how the above limitations can be avoided by 'linked' lists. What do you mean by linear list and generalized list ?   2 + 2 + 2

3. (a) What is a circular list ? Write an algorithm for inserting a node at the front.   5

(b) Suppose you are given two polynomials. Represent the polynomial in a suitable data structure and write a 'C' function to add two polynomials.   6

(c) Write a 'C' program to implement a stack using a single array.   9

4. (a) Define a binary tree. What do you mean by tree traversal ? Write one traversal algorithm.   2 + 2 + 2

(b) Construct an AVL tree with the following key arriving in the given order : k, t, v, a, e 6

(c) Insert the following keys into a B-tree of order 3 and draw the final tree : 10, 24, 23, 11, 31, 16, 26, 35, 29, 20, 46, 28, 13, 27, 33, 21.   8

Group B

5. (a) Write a 'C' function to compute the degree of a given vertex of an undirected graph when the graph is represented by an adjacency matrix.   6

(b) What do you mean by graph traversal ? Define depth-first traversal (DFS) of a graph. Write an algorithm of non-recursive depth first traversal.   2 + 2 + 4

(c) Compare and contrast between DFS and BFS (Breadth First Search).   6

6. (a) What is hashing ? Give the characteristics of hash function.   2 + 4

(b) Name different hash functions with a brief description and analysis.   8

(c) What is collision resolution technique in hashing ? Explain.   6

7. (a) What are the differences between internal sorting and external sorting ? When is a sorting technique said to be stable ? Explain with an example.   3 + 2 + 2

(b) Write a 'C' program of sorting a set of numbers in descending order using selection sort. Analyse the algorithm complexity.   4 + 2

(c) Show the steps of sorting the following sequence in

ascending order using quick sort method. Show the snapshots after every interchange : 25, 57, 48, 37, 12, 92, 86   7

8. (a) Write a 'C' function to create a sequential file that is the same as file 1, except that all records on file 1 that are also on file 2 do not appear. Assume file 1 and file 2 records appear in sorted order based on the key field.   7

(b) Why do B-trees allow more flexible use of a disk than directories implemented using prime and overflow areas ?   6

(c) Suppose records of a file were stored in a hash table, with buckets for random access by key, and linked in a sequential order by pointers for sequential access. Compare the insertion and retrieval times of such an implementation with the use of B-trees.   7

Group C

9. Answer the following :   10 × 2

(i) 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.

Determine the sequences of popped out values.

(ii) Write the preorder expression of the following algebraic expression :

$$[a + (b - c)]^* [(d - e) / (f + g - h)].$$

(iii) What is the average and best case time complexity of insertion sort algorithm ?

(iv) Arrange   ting time in increasing order of
     magnitud

     $O(n^2)$, $O($), $O(1)$, $O(n)$, $O(n^3)$, $O(n \log n)$.

(v) Suppose   ; declared like

    union
    {
        flo
        cha
        int
    } num

    Assumin   requires 4 bytes, char requires 1
    byte an   es 2 bytes , how many memory
    space us   ariable num.

(vi) What is (llection ?

(vii) What is   conquer methodology ? Name
      one sorti   m where the divide and conquer
      methodo   l.

(viii) Which is   ructure give overflow error even
       though t   number of elements in it is less
       than its s   ; the solution to this problem ?

(ix) What is t   algorithm used in solving the 8
     Queens

(x) What is tne run time complexity of quick sort when the list is already sorted ?

S'12:4FN : CP403 (1448)

DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A,**
ANY TWO *from* **Group B** *and* ALL *from* **Group C.**

*All parts of a question (a, b, etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.*

Figures on the right-hand side margin indicate full marks.

Group A

1. (a) What is a sparse matrix ? Describe a data structure
for the efficient storage of a sparse matrix.   5

    (b) Write a recursive algorithm to compute the nth
Fibonacci number.   5

    (c) Define Big O notation and what is its utility in analysis
of algorithms ?   5

    (d) Describe briefly *three* types of structures used for
storing strings.   5

2. (a) Discuss the array representation of stacks. Write the
algorithms for push and pop stack operations.   4 + 3 + 3

    (b) Write a recursive algorithm to compute X^n, where X
is a floating point number and n, a positive integer.   4

(Turn Over)

(c) What is worst case and average case analysis ? 6

3. (a) For a $n \times n$ square matrix array, write a module to find the (i) number of non-zero elements (N); (ii) sum of elements (S) above the diagonal, and (iii) product of diagonal elements (P). 3 × 3

(b) Write a recursive algorithm to compute the factorial of a number n. 5

(c) Write algorithm for searching a link list when (i) list is sorted, and (ii) list is unsorted. 3 + 3

4. (a) Discuss array and linked representation of queue data structure. What are deques ? 3 + 3 + 3

(b) Write a recursive algorithm to reverse the order of items in a linked list. 5

(c) What is pattern matching ? Write a pattern matching algorithm to find the index of a string S in a string T. 2 + 4

Group B

5. (a) Describe briefly selection sort. Show different passes required to perform selection sort on the following list of numbers :

76, 32, 43, 10, 87, 21, 65, 54

Show the worst case and average case for selection sort. 3 + 5 + 2

(b) What is the Euler tour ? Find the Eulerion tour of the graph shown below : 2 + 4
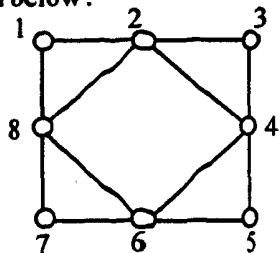
(c) What is the size of the smallest AVL tree with height 6 ? 4

6. (a) Sort the following list using merge sort : 8

8, 2, 4, 6, 9, 7, 10, 1, 5, 3

(b) Write an algorithm that finds both the smallest and the largest number from a list of n numbers. 4

(c) What are spanning trees and minimum cost spanning tree ? Write Kruskal's algorithm to identify minimum spanning tree of an undirected graph. 4 + 4

7. (a) Explain, with a suitable example, insertion sort. What are the worst case and average case time complexity of insertion sort ? 4 + 2

(b) What is graph traversal ? Write an algorithm for graph traversal. 3 + 4

(c) Write an algorithm to delete an element D from a binary search tree so that the tree remains a binary search tree. 7

8. (a) Write an algorithm for merge sort. 6

(b) Write short notes on following file organizations : 4 × 3

(i) Sequential organization

(ii) Random organization

(iii) Linked organization

(iv) Inverted organization.

(c) What is the adjacency matrix representation of graph ? 2

Group C

9. Answer the following in brief : 10 × 2

(i) What are the major data structures used in the following areas : RDBMS, network data model, and hierarchical data model ?

(ii) What is the minimum number of queues needed to implement the priority queue ? Explain.

(iii) What is the data structures used to perform recursion and why ?

(iv) Traverse the given tree using in-order, pre-order and post-order traversals.
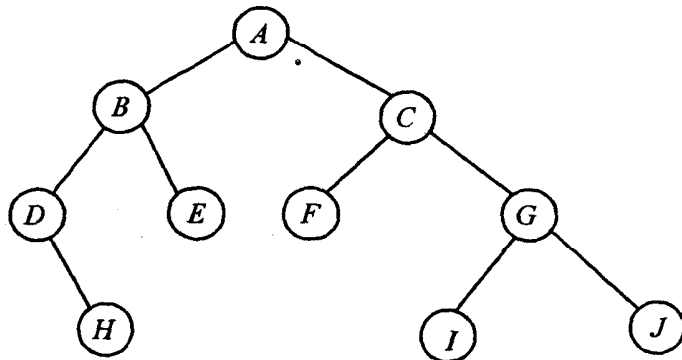
(v) There are 8, 15, 13, 14 nodes in four different trees. Which of them could have formed a full binary tree ?

(vi) What is a spanning tree ? Does the minimum spanning tree of a graph give the shortest distance between any two specified nodes ?

(vii) What is a doubly linked linear list ? State the advantage and disadvantage of such a list.

(viii) Determine the worst case time complexity of insertion sort and when does such situation arise ?

(ix) A two-dimensional integer array arr[6][8] is stored in row major order with base address 512. What is the address of arr[3][4] ?

(x) What is a B-tree ? What is a B-tree of order 3 referred to as ?

W'11:4FN : CP403 (1448)

DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A,**
ANY TWO *from* **Group B** *and* ALL *from* **Group C.**

*All parts of a question (a, b, etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.*

Figures on the right-hand side margin indicate full marks.

Group A

1. (a) What is meant by the 'stack overflow' condition ? Is it
applicable to the linked list method of implementation
of the stack ? Give reasons.   5

    (b) Show that $n \log(n) \in \theta(\log(n!))$.   5

    (c) Suppose you are given a two-dimensional array
$a[r, c]$. The base address is base $[a]$, the size of each
element is w, the lower limit of the row subscript is lbr
and the lower limit of the column is lbc. Find the
address of the element $a[k1, k2]$, if the array is stored
in (i) row major form; and (ii) column major form.   5

    (d) Write a recursive algorithm to reverse a string of any
length. You can use any number of functions but those
should be properly explained.   5

2. (a) How can you differentiate stack from an array ? Implement the push and pop operation of stack with testing for exceptional conditions. 2 + 4 + 4

(b) Design a method to simulate a stack using queues, considering as many queues as required. 10

3. (a) Design a data representation which sequentially map n data objects into an array $a[1, n]$, $n1$ of these data objects are stacks and the remaining $n2$, equal to $n-n1$, are queues. Write algorithms to add and delete elements from these objects. 10

(b) What are the advantages of circular link list over linear list ? Write a function which will concatenate two lists using a circular list. 10

4. (a) What are the problems of a circular link list ? Specify the appropriate data structure which can overcome the above problems. Using the above data structure, write a routine which inserts a node with information field x to the right of node p. 12

(b) Write a routine *addsame* to add two long integers of the same sign represented by doubly linked list. 8

Group B

5. (a) Write the algorithm for radix sort. 8

(b) Using the following traversals, construct the corresponding binary tree : 6

INORDER : H K D B I L E A F C M J G

PREORDER : A B D H K E I L C F G J M

(c) Use Kruskal's algorithm to extract the minimum spanning tree of the graph given in Fig. 1 : 6
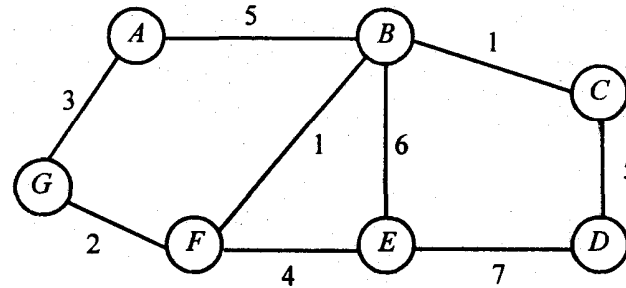

Fig. 1

6. (a) Write the algorithm for the creation of a binary search tree. 7

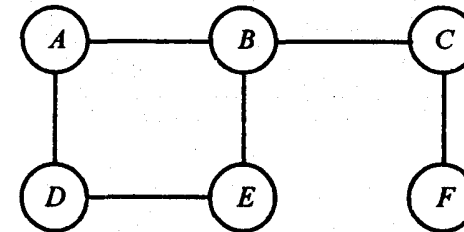(b) Determine the BFS and DFS traversals of the graph shown in Fig. 2 : 8

Fig. 2

(c) Name the most efficient search technique. Analyze its efficiency. 5

7. (a) Write procedures for adding and deleting a node from a balanced tree and leave the resulting tree balanced. 10

(b) Show the steps of quick sort on the following set of elements :

25, 57, 48, 37, 12, 92, 86, 33.

Assume the first element of the list to be the pivot element. 10

8. (a) Write an algorithm to delete an arbitrary record *m* from a multi list file with *n* keys. Assume that the order of records in individual list is irrelevent. Use the functions search (*x*) and update (*x*, *a*) to search and update an index. How many disks accesses are needed ?   9

(b) What is the maximum number of disk accesses made during a search of B-tree of order 2 *m*, if each node is 2 disk blocks and requires 2 disk accesses to retrieve ?   6

(c) What is the worst case complexity of heap sort technique ? Is the method stable ?   5

Group C

9. Answer the following :   10 × 2

(*i*) Algorithm *A* requires n^2 days and algorithm *B* requires n^3 sec to solve a problem. Which algorithm would you prefer for a problem instance with $n = 10^6$?

(*ii*) Assume the recurrence relation $T(N) = 2T(N/2) + N$, $N \geq 2$ with boundary condition $T(1) = 0$. What is the time complexity ?

(*iii*) The height balanced equivalent of the following binary search tree is :

(*a*)

(*b*)
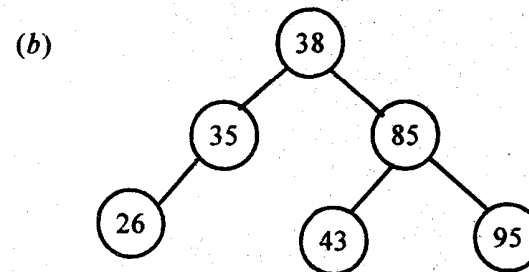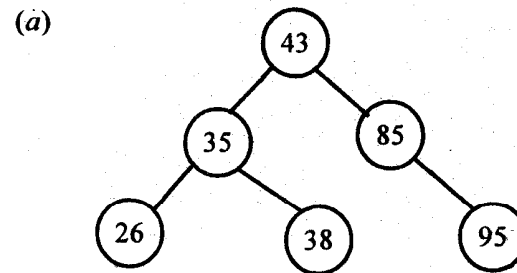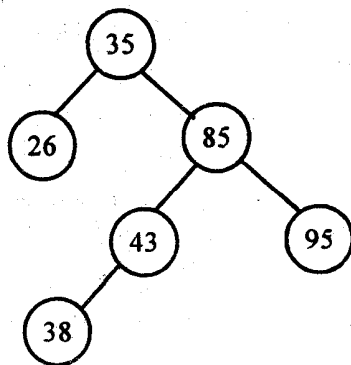
(*c*) The given tree is height-balanced.

(*d*) None of the above.

(*iv*) The prefix equivalent of the expression (3 + 5) * 7 / (8 * 9 + (10 * 11) % 12) is

   (*a*) * + 3 5 / 7 + * 8 9 % * 10 11 12

   (*b*) / * + 3 5 7 + * 8 9 % * 10 11 12

   (*c*) * + 3 5 + * / 7 8 9 % * 10 11 12

   (*d*) * + 3 5 + * / % 7 8 9 10 * 11 12

(*v*) Retrieval from a hash table with open addressing, linear probing and load factor λ requires _____ probes in the unsuccessful case.

(*vi*) The implication of the declaration : **int (*p(char*a))** **[10]**; is _____ .

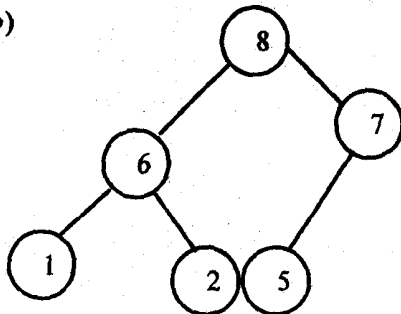(vii) Considering the following Max-Heap :
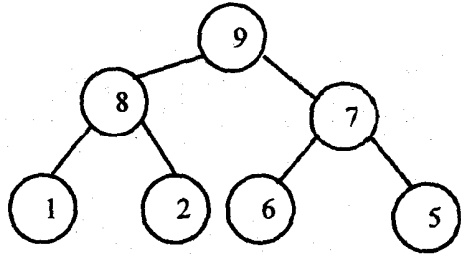
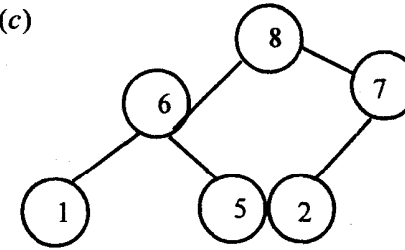If 9 be deleted, the resulting Max-Heap is :
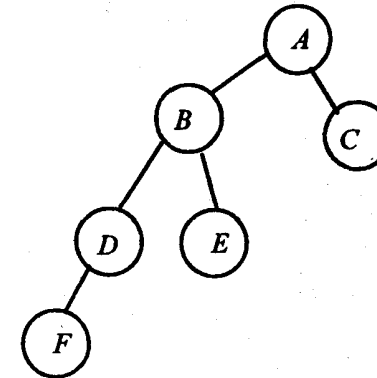
(a)
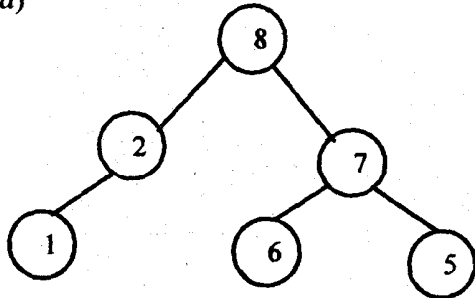
(b)

(c)

(d) None of the above.

(viii) Given the following unbalanced tree :

Draw the balanced tree.

(ix) For sorting contiguous list of records, quick sort may be preferred over merge sort because it

(a) requires less time always.

(b) does not require extra space for an auxiliary storage.

(c) requires more programming effort.

(d) takes help of the recursive procedure.

(x) To find maximum elements of a square matrix $n \times n$ takes quadratic time. Justify.

DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A**,
ANY TWO *from* **Group B** *and* ALL *from* **Group C**.

*All parts of a question (a, b, etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.*

Figures on the right-hand side margin indicate full marks .

Group A

1. (a) Suppose $T_1 (n)$ and $T_2 (n)$ are the time complexities
   of two program fragments P_1 and P_2, where
   $T_1 (n) = 0 (f (n))$ and $T_2 (n) = 0 (g (n))$. What is
   the time complexity of program fragment P_1 followed
   by P_2? 4

   (b) Suppose $P (n) = a_0 + a_1 n + a_2 n^2 + ... + a_m n^m$, that
   is, suppose degree $P (n) = m$. Prove that
   $P (n) = O (n^m)$. 4

   (c) Write a recursive algorithm to print the numbers 1
   to 10. 6

(*d*) Suppose multidimensional arrays *A* and *B* are declared using *A* (-2:2, 2:22) and *B* (1:8, -5:5, - 10:5), (*i*) find the length of each dimension and the number of elements in *A* and *B*; and (*ii*) consider the element, $B[3,3,3]$ in *B*. Find the effective indices E_1, E_2, E_3 and the address of the element, assuming Base (*B*) = 400 and there are *w* = 4 words per memory location. 6

2. (*a*) Suppose a linear linked list named LIST. Write an algorithm which deletes the last node from LIST. 5

(*b*) Suppose NAME1 is a list. Write an algorithm which copies NAME1 into a list NAME2. 5

(*c*) Discuss the advantages, if any, of a two-way list over a one-way list for each of the following operations : 5

(*i*) Traversing the list to process each node.

(*ii*) Deleting a node whose location LOC is given.

(*iii*) Searching an unsorted list for a given element ITEM.

(*iv*) Searching a sorted list for a given element ITEM.

(*v*) Inserting a node before the node with a given location LOC.

(*d*) Suppose LIST is the head node of a circular list. Write an algorithm which deletes the last node from LIST. 5

3. (*a*) Write an algorithm to transform an infix expression into a postfix expression. 8

(*b*) Write an algorithm that deletes and element from the head of a queue and assigns it to the variable ITEM. 4

(*c*) Consider the following queue of characters where QUEUE is a circular array which is allocated six memory cells :

FRONT = 2　　　REAR = 4,　　　QUEUE : —, A , C , D , —, — ("—" denotes empty memory cell).

Describe the queue as the following operations takes place : 8 x 1

(*i*) *F* is added to the queue.

(*ii*) Two letters are deleted.

(*iii*) *K*, *L* and *M* are added to queue.

(*iv*) Two letters are deleted.

(*v*) *R* is added to queue.

(*vi*) Two letters are deleted.

(*vii*) *S* is added to queue.

(*viii*) Two letters are deleted.

4. (*a*) Compare the advantages and disadvantages of using recursion as compared to iteration. 7
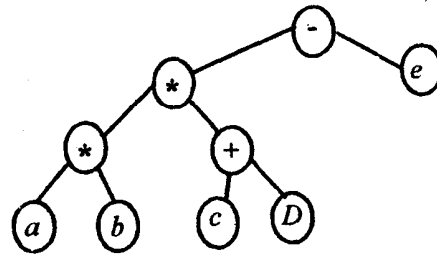
(b) Order the following function by growth rate :

$N, \sqrt{N}, N^{1.5}, N^2, N \log N, N \log\log N, N \log (N^2),$

$2/N, 2^N$ 37. Indicate if any functions grow at the same rate. **5**

(c) Given two sorted lists, L_1 and L_2. Write an algorithm to compute $L_1 \cup L_2$ using only the basic list operations. **8**

Group B

5. (a) Give the prefix, infix and postfix expressions corresponding to the tree as shown below : **6**

(b) Show that the maximum number of nodes in a binary tree of height h is $2^{h+1} - 1$. **5**

(c) Show the result of inserting 3, 1, 4, 6, 9, 2, 5, 7 into an initially empty binary search tree. Also, show the result of deleting the root. **7**

(d) Compute the height of tree as shown below : **2**
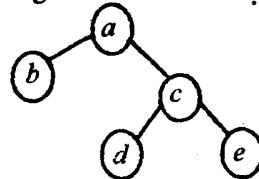
6. (a) Give an example of hash function. What are the advantages and disadvantages of various collision resolution strategies in hashing? What are the advantages and disadvantages of hashing based searching and sorting? **10**

(b) Show how the heapsort processes the input 142, 543, 123, 65, 453, 879, 572, 434, 111, 242, 811, 202. Show all steps. . **10**

7. (a) What is meant by a spanning tree of a graph? Give an algorithm to find a spanning tree. What is the complexity of your algorithm? **10**

(b) What do you mean by tree traversal? Give a recursive algorithm for tree traversal. Determine the complexity of your algorithm. **10**

8. (a) What is the difference between a sequential and direct access file? Write an algorithm for reading the records of a direct access file in which chaining with separate lists is used as the overflow technique. **10**

(b) Explain what do you mean by an inverted file using an example. What are the advantages of an inverted file over other file structures. **10**

Group C

9. Answer/choose the correct answer for the following : 10 x 2

(i) The lower bound is denoted as ——— for $T(n) = 3n^3 + 2n^2$.

(ii) Implementing a list as an array involves the following disadvantages :
(a) The size of the list is finite.

(b) The size of list cannot be extended.

(c) Insertion of an item requires subsequent elements from the positions to slide down.

(d) All of the above.

(iii) The header to a linked list is also called

(a) a base.

(b) an anchor.

(c) a null pointer.

(d) both (a) and (b) above.

(iv) What are the properties of M-ary tree ?

(v) Suppose the following list of numbers is inserted in order into an empty binary search tree : { 50, 60, 45, 35, 40, 90 }. Find the final tree T.

(vi) Write an algorithm to delete top element of STACK and assigns it to the variable item.

(vii) What is an input-restricted deque and an output-restricted deque ?

(viii) What do you mean by minimum spanning tree of an undirected graph ?

(ix) What are three types of structures for sorting strings ?

(x) Write the meaning of the statement given below :

PEEP (S, TOP, I)

DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A,**
ANY TWO *from* **Group B** *and* ALL *from* **Group C.**

*All parts of a question (a, b, etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.*

Figures on the right-hand side margin indicate full marks.

Group A

1. (*a*) Write C functions to perform push and pop operations in a stack. Using these functions, write a program converting a positive integer (in decimal) to its equivalent binary form.　　10

   (*b*) Consider that you are supplied with an ordered (ascending) singly linked list headed by the pointer head. The information contents of this list are integers. You are required to print the integers of the list in descending order of sequence. Write a C function to achieve the task.　　10

2. (a) Show that the function $f(n) = a_0 + a_1 n + a_2 n^2 + \ldots + a_m n^m \in O(n^m)$. **4**

(b) Suppose $f_1(n) \in O(t(n))$ and $f_2(n) \in O(t(n))$. Which one of the following is true ? In case an option is false, give reasons that why is it false (an example is sufficient). **6**

   (i) $f_1(n) + f_2(n) \in O(t(n))$

   (ii) $f_1(n) - f_2(n) \in O(t(n))$

   (iii) $f_1(n) / f_2(n) \in O(1)$

   (iv) $f_1(n) \in O(f_2(n))$.

(c) Consider the definition

   int $a[4][5][6]$;

Assume that the base address of the array a is 2000. Give an expression that computes the address of an element $a[i][j][k]$ of the array, where i, j and k are valid integers. **4**

(d) Write a recursive algorithm (C like) to compute the greatest common divisor (gcd) of two given positive integers. **6**

3. (a) What are the advantages and disadvantages of a singly-linked list over a contiguous list ? **4**

(b) Consider a string s consisting the characters '[', ']', '{', '}', '(' and ')' only. Write an algorithm to check if the string s is balanced in usual algebraic sense. **8**

(c) Given pointers $L1$ and $L2$ of two sorted singly-linked lists of integers. The lists may have some common elements. Write an algorithm to create a new sorted singly-linked list, L, consisting the elements of $L1 \cup L2(L1$ union $L2)$. **8**

4. (a) 'Recursion is better than iteration'. Critically comment on this statement. **6**

(b) Usually a queue is implemented using two pointers, viz., front and rear. Is it possible to implement a linked queue that requires only one pointer ? If yes, how to implement it ? **5**

(c) What is tail recursion ? How to remove it ? **5**

(d) 'Arrays are kept in row major order'. How is the statement relevant to a programmer ? **4**

Group B

5. (a) Write an algorithm for heapsort technique and explain it with a suitable example. What is its average case time complexity ? **10**

(b) What is a binary search tree (BST) ? Give the declarations required to implement one such BST. What type of difficulties a BST might have and how to improve the search performance by converting it to an AVL one ? **10**

6. (a) Define a binary tree. How is it different from an ordinary tree. **4**

(b) Form the binary tree for which the following are given : **6**

In-order traversal :   f   h   i   g   b   a   d   e   c
Pre-order traversal :   a   b   f   g   h   i   c   d   e

(c) A *full node* is a node with two children. Prove that the number of full nodes plus one is equal to number of leaves in a non-empty binary tree. **4**

(d) What is the concept of threading ? How is it implemented ? **6**

7. (a) Given input {3481, 2313, 7163, 9149, 4944, 9679, 1089} and a hash function $h(x) = x$ mod 10. Show the result of (with explanation) **3 × 3**

    (i) separate chaining.

    (ii) open addressing using linear probing.

    (iii) open addressing using quadratic probing.

(b) Critically compare different collision resolution techniques known to you. **5**

(c) Write an algorithm that accepts a pointer to a BST and deletes the smallest element from it. **6**

8. (a) What is a B-tree ? Generate a B-tree of order 5 with the alphabets (letters) arrive in the sequence as follows : **4 + 8**

    a g f b k d h m j e s i r x c l n t u p

(b) How is a B-tree different from B^+ -tree ? **4**

(c) Give at least *two* techniques of representing graphs in memory. **4**

Group C

9. Answer the following : **10 × 2**

(i) Take a queue containing numbers 10, 15, 5, 25, 30 in which 30 has been inserted first. After performing the following operations, what would be the contents of the queue ?

    (a) Delete two elements

    (b) Insert 7 and then 20

    (c) Delete an element

(ii) Assume that a Binary Search Tree is as follows :
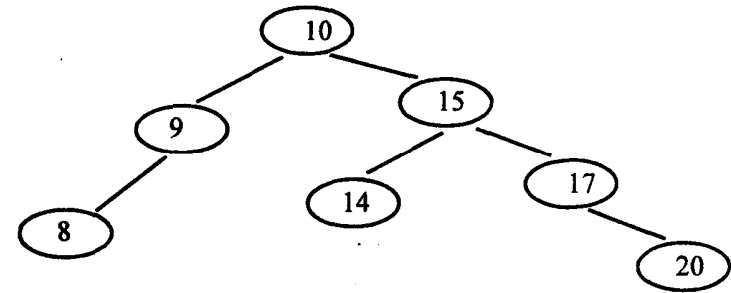
What will be the configuration of the BST if 10 is deleted from it ?

(iii) Translate the following infix expression into post-fix notation :

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

(iv) A stack is to be implemented using an array. The associated declarations are :

int   stack [100];
int   stacktop = 0; /* indicates stack is empty */

Which one of the statements below performs a PUSH operation ?

    (a) stack [stacktop ++] = value;
    (b) stack [++ stacktop] = value;
    (c) stack [stacktop --] = value;
    (d) stack [-- stacktop] = value;

(v) Choose the statement for performing POP operation on the stack described in (iv) above :

    (a) value = stack [stacktop ++];
    (b) value = stack [++ stacktop];
    (c) value = stack [stacktop --];
    (d) value = stack [--stacktop];

(vi) A list of names is stored as a binary tree in the lexicographic order. The left subtree of a node contains all names less than the name in the node. Likewise, the right subtree has all names greater than the names in the node. What tree traversal method would you use to print out the names in ascending order ?

(vii) Choose the statements to be inserted in the following procedure for insertion of an element into a list.

```
struct LISTELEM {
            int val;
            struct LISTELEM *next, *prev;
            };

void        lstinsert(elem, prev)
struct  LISTELEM   *elem, *prev;
{
    elem->next = prev->next;
    eiem->prev = prev;
    /* statements to be inserted here */
}
```

(a) prev-> prev = elem-> next -> next = elem;

(b) prev-> next = elem- > next -> next = elem;

(c) prev-> prev = elem-> next -> prev = elem;

(d) prev-> next = elem-> next -> prev = elem;

(viii) A complete binary tree is a binary tree in which there is one node at the root level, two nodes at level 2, four nodes at level 3, etc. How many nodes are there in a complete binary tree having 5 levels ?

(ix) Linked lists are preferred to other data structures when

(a) number of elements in the list is known beforehand.

(b) the elements are in ascending or descending order.

(c) no deletion of elements needs to be performed.

(d) insertions and deletions are frequent in a list of unknown size.

(x) A programmer wants to write a rudimentary scanner that would check unbalanced braces, brackets and square brackets in a C source file. He maintains a storage list, which would get all open braces, brackets and square brackets. When a closing brace, bracket or square bracket is seen, it is compared with contents of this list to issue a suitable diagnostic. The best possible structure of the storage list to accomplish this task is

(a) linked list

(b) struct

(c) union

(d) stack.

DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A,**
ANY TWO *from* **Group B** *and* ALL *from* **Group C.**

*All parts of a question (a,b,etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.*

Figures on the right-hand side margin indicate full marks.

Group A

1. (a) What is sparse matrix ? Write an algorithm to add to
sparse matrix and explain the assumed data structure
in support of your algorithm.   10

   (b) Write a recursive sub-algorithm to implement
following recursive function :

$$F(n) = \begin{cases} 1, \text{ if n} <=2 \\ F(n-1)+2*F(n-2), \text{ otherwise} \end{cases}$$

   Show how the designed algorithm will find F(5).   10

---

2. (a) Obtain the address calculation function for a two-dimensional array stored in column major order. Obtain location of element M(-1,2) for a matrix M[6,5], having row subscript range {-2 to 3} and column subscript range {0 to 4}. 10

(b) Write an algorithm to remove multiple occurrences of an element (on the basis of information) from a single linked list. Assume every node in list has one information (int type) and one link (pointer) field. 10

3. (a) Name parameters used for analyzing an algorithm. Find complexity of matrix multiplication and linear search algorithm. 10

(b) Suggest an appropriate data structure to represent a polynomial and write suitable algorithm to multiply two polynomials. Show how your algorithm will compute multiplication of P_1 and P_2, where 10

$P_1(x,y,z)$: $4X^9Y^3Z^4 + 2X^2Y^2Z - 3X^3YZ^5 - 5X^2Y^3Z^4 + 3$

$P_2(x,y,z)$: $3X^{10}Y^9Z^4 + 3X^9Y^3Z^4 - 5XYZ + 2X^3YZ^5 - 7$

4. (a) Write and explain insertion and deletion policy of priority queue ? Write an algorithm to implement priority queue with three levels of priority. Assume there can be at most six elements in each priority. 10

(b) What is the role of header node in circular linked list ? Write an algorithm to find weather the given list is empty or not. 10

Group B

5. (a) How many passes are required to sort a data set ?

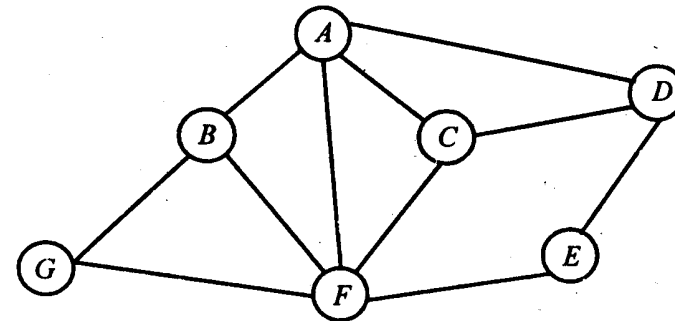Show how following data set can be sorted by radix sort :

28, 67, 45, 87, 49, 85, 45, 22, 455, 78,

124, 569, 48, 03, 557 10

(b) Write and explain algorithm to delete a node from above constructed BST. 10

6. (a) Represent following graph in adjacency list and explain how breadth first searching (BFS) algorithm is used to traverse the graph : 10

(b) What is the difference between a binary search tree (BST) and heap ? For a given sequence of numbers, construct a heap and a BST.

34, 23, 67, 45, 12, 54, 87, 43, 98, 75, 84, 93, 31 10

7. (a) What is the bottleneck of binary search tree (BST) ? Explain how it is minimized by B-tree and AVL tree. Construct BST, B-tree, AVL tree.

22, 45, 65, 67, 78, 89, 87, 65, 54, 43, 32, 57,
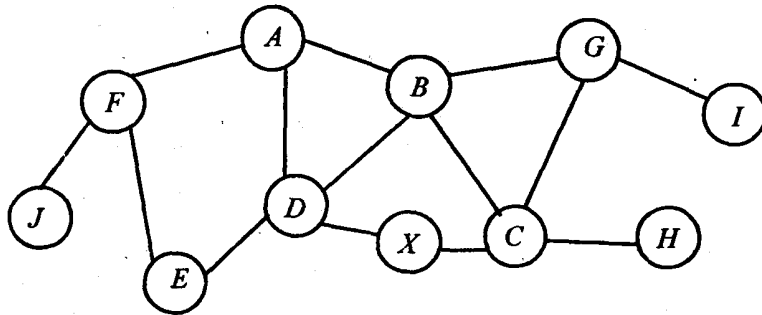
48, 39, 26, 37, 60, 40, 20 10

(b) Can you find a unique tree, if any *two* traversal are given ? Find all leaf nodes of the tree, where preorder and inorder are given below : 10

Preorder : A B C D E F

Inorder : B A D E C F

8. (a) What is spanning tree ? Obtain spanning tree of the given graph showing paths of shortest distance from a node (let X) to all other nodes. 10

(b) What is threaded tree ? How are threads represented ? Add threads to a given tree using inorder traversal and denote them by re-drawing the tree. 10

Group C

9. Answer the following questions by selecting one from given set of probable answers enclosed in braces{} : 2 x 10

(i) A node in graph, with zero degree, is called ____{pendent/isolated}

(ii) _____ is used in implementation recursion. {Array /Stack/Queue}

(iii) Policy of queue is _____. {Last In Last Out(LILO) /First In Last Out(FILO)}

(iv) Result of following post-fix expression is _____ {20/16/17/22} 234* + 621 +/+

(v) Tree is a non-linear data structures and _____represented using linear data structures like array {cannot be / can be}

(vi) Algorithm with complexity (n^2) is considered as _____ algorithm than of complexity (log n). {better/worse}

(vii) The result of DFS traversal in a graph results _____ {a graph / a spanning tree}

(viii) In AVL tree, difference of height in left sub-tree and right sub-tree for every node is _____ . { zero / one / atmost one / two / atmost two}

(ix) A minimal connected graph is called _____. {disconnected graph / unconnected graph / tree / null graph}

(x) In a tree of 18 nodes, there will be_____ edges.{9, 36, 17, 18}

DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A**,
ANY TWO *from* **Group B** *and* ALL *from* **Group C**.

*All parts of a question (a, b, etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.*

Figures on the right-hand side margin indicate full marks .

Group A

1. (*a*) Define the following terms in brief :   2 × 5

    (*i*)   Time complexity

    (*ii*) Space complexity

    (*iii*) Small o notation

    (*iv*) Big O notation

    (*v*)   Asymptotic notation.

    (*b*) Write a 'C' function that would print all prefixes of
    a given string of length less than 20.   10

2. (a) Explain the concept of hashing using division method of hashing. Store the following values in a hash table of size 11.

$$25, 45, 96, 101, 102, 162, 197, 201$$

Show the hash table after storing the values. 10

(b) Differentiate between row major and column major array index notation. How is index calculated in both? Explain your answer by using the example of an integer array int a [10][10] and the element a [2][5] is to be accessed. 10

3. (a) Write a recursive algorithm to count the number of elements in an array. Explain the advantages and disadvantages of using the recursive formulation of the algorithm compared to non-recursive formulation. 10

(b) What is a doubly linked list? What are its applications? Explain how an element can be deleted from the list using appropriate pseudo code. 10

4. (a) Explain, using appropriate pseudo code, how an element can be inserted and deleted in a circular queue. 10

(b) Explain how an arbitrary arithmetic expression can be evaluated using a stack. Explain your answer by considering the example $2 + 5 * 15 + 30 \div 3$. 10

Group B

5. (a) Write an algorithm to perform breadth first search (BFS). Compare the BFS and DFS search techniques. 10

(b) What is a CWL tree? How can a normal tree be converted into a balanced AWL tree? 10

6. (a) Explain the working of merge sort on the following data:

$$10, 15, 0, 17, 20, 25, 30, 16, 70, 6$$

Show all intermediate steps. Also, mention its time complexity. 10

(b) Discuss pre-order and post-order tree traversal techniques. Write the pseudo code for these two traversal methods. 10

7. (a) What are the applications of files in 'C' programming? Differentiate between multi-indexed and inverted files. 10

(b) Consider following binary tree:
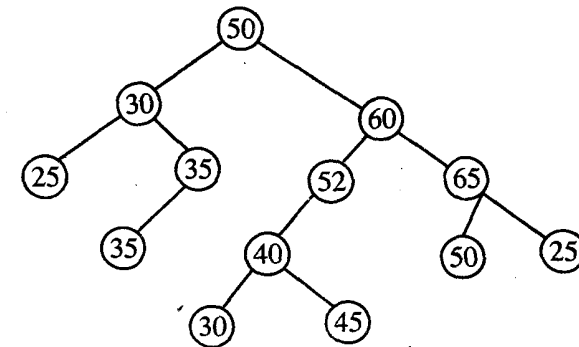
Show intermediate tree steps to add element 58 in this tree. 10

8. Write short notes on any *four* of the following : 5 × 4

   (*i*) Abstract data types

   (*ii*) Prim's algorithm

   (*iii*) Topological sort of a directed graph

   (*iv*) Application of queues

   (*v*) Multi-indexed files.

Group C

9. State *true* or *false* for each of the following statements : 2 × 10

   (*i*) Any graph can be converted into a tree by adding and deleting some of the edges.

   (*ii*) Dynamic memory allocation is done at run time.

   (*iii*) In a balanced tree, the maximum height difference between nodes can be n.

   (*iv*) The maximum number of edges in an undirected graph with n vertices are $n(n-1)/2$.

   (*v*) Expressions having unary operator cannot be represented using prefix or post-fix notation.

   (*vi*) Array of structure is a homogeneous data structure.

   (*vii*) Pointer is a data type which holds the address of a data element.

   (*viii*) The Kruskal's algorithm is used for generating the spanning tree of a graph.

   (*ix*) The length of a linked list may be varied at run time.

   (*x*) A circular queue may always be used in place of a doubly linked list.

DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A** ,
ANY TWO *from* **Group B** *and* ALL *from* **Group C.**

*All parts of a question (a, b, etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.*

Figures on the right-hand side margin indicate full marks .

Group A

1. (a) Write a recursive program to find sum of all even
        natural numbers below an input number.   10

    (b) Compare array, list and file data structures.   10

2. (a) Represent the following two polynomials in
        appropriate data structures. Write a suitable algorithm

to find the sum of these polynomials and compute the value of $R(3,2)$.

$$P(x,y): 12X^3Y^4 + 5X^2Y^3 - 8XY - 11$$

$$Q(x,y): 2X^2Y^3 + 7X^3Y - 3XY - 11$$

$$R(x,y) : ?$$

where $P + Q \rightarrow R$.   10

(b) What is a priority queue? How can it be implemented? Explain an application of priority queue.   10

3. (a) Write a recursive algorithm to check whether a given string is a palindrome or not. Write an analytical note to compare its equivalent iterative algorithm to recursive algorithm for finding palindrome.   10

(b) Write an algorithm to insert an element in an existing double linked list. Assume LEFT and RIGHT are two pointers keeping the address of two ends of the LIST. Explain working of your algorithm using an example. 10

4. (a) When more than one algorithms are available for a problem, how will you choose between them. State various parameters used to analyze and choose an algorithm. Write your comments on how to choose between the following alternative algorithms 'BUBBLE SORT' vs. 'MERGE SORT'.   10

(b) Write an algorithm to insert an element in a 'Circular Linked List', represented in sequential representation. Explain the working of your algorithm. Also, mention the condition of overflow. 10

Group B

5. (a) What are nonlinear data structures? Explain how a complete binary tree can be represented in an array. Show the equivalent binary tree that is represented in the following one-dimension array: 10

12	32	33	65	74	80	48

(b) What is hashing? Explain how it helps in faster accessing of the information? Write a method to handle the situation of collision in hashing. Explain your method.   10

6. (a) Find POSTFIX of the tree having INFIX and PREFIX as given below: 10

INFIX   : G C A D E B F
PREFIX : A C G B D E F
POSTFIX : ?

(b) Why is quick sort named as 'quick'? Show how following data set can be sorted by quick sort. Show each step in sorting.   10

78, 23, 11, 88, 43, 55, 67, 55.

7. (a) Write an algorithm to find the number of leaf nodes in given binary tree. Explain its working using an example? What is its time complexity?   10

(b) Draw the graph represented by given adjacency matrix. Also, find the depth first search (DFS) tree for the obtained graph. How can a graph be represented in a program ?   10

	a	b	c	d	e
a	1	0	1	0	1
b	0	0	1	1	0
c	1	1	0	0	0
d	0	1	0	0	1
e	1	0	0	1	0

8. (a) What is the difference between a heap and a binary search tree ? Obtain heap and binary tree for following data set :   10

45, 56, 78, 23, 11, 54, 88, 43, 55, 21, 67, 55

(b) Define a height balanced tree. Write methods to keep balanced height when a new node is inserted. Consider all situations of insertion of a new node into a height balanced tree.   10

Group C

9. State *true* or *false* for each of the following statements :   2 x 10

(i) A graph with all node having zero degree is called null graph.

(ii) Array of structures is a homogeneous data structure.

(iii) Stack follows the order FILO (first in last out).

(iv) A circular queue has better utilization of space than a linear queue.

(v) B-tree is a binary tree.

(vi) An algorithm with complexity (2^n) is considered to be a better algorithm than one having complexity $(\log n)$.

(vii) Non-linear data structures can be represented using linear data structures.

(viii) Expression, with Unary operators, cannot be represented using prefix or postfix notation.

(ix) A connected graph with unique path (only one path) between every pair of nodes is called a tree.

(x) Every recursive algorithm can be converted into a non-recursive algorithm by using an external stack.

DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A**,
ANY TWO *from* **Group B** *and* ALL *from* **Group C.**

*All parts of a question (a, b, etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answer may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.
Figures on the right-hand side margin indicate full marks.*

Group A

1. (a) What are the basic operations performed on an array ?
Give examples. 10

(b) Write algorithm and program to insert values to a
ordered cursor based list. 10

2. (a) What are the different types of implementations of
stack ? 10

(b) Write a procedure to find the presence of one string
in another string. 10

3. (*a*) Write a recursive algorithm to determine the sum of the contents of all nodes in binary tree.   10

    (*b*) Write an algorithm to merge two circular lists, *A* and *B*, to produce a resultant list *C*.   10

4. Explain any *four* of the following :   5×4

    (*a*) Application of queues.

    (*b*) Heap sort technique.

    (*c*) Linear search.

    (*d*) String operations.

    (*e*) Hashing.

Group B

5. (*a*) Define a binary tree. Represent the array $X = [A, B, C, D, E]$ in binary tree representation. 10

    (*b*) Draw the binary tree for following statements :   5×2

      (*i*) $(a + b) + c - d / (-e)$

      (*ii*) $e^{-at} \sin (bt + c)$.

6. (*a*) Explain about breadth first search in detail.   10

    (*b*) Explain about depth first search in detail.   10

7. Explain the following sorting methods :   10×2

    (*a*) Quicksort

    (*b*) Merge sort.

8. Write short notes on any *two* of the following :   10×2

    (*i*) Tail recursion

    (*ii*) Radix sort

    (*iii*) Indexed files.

Group C

9. (*A*) Choose the *correct* answer for the following :   2×4

    (*i*) A strict binary tree with *n* leaves will have

      (*a*) $2n + 1$

      (*b*) $2n - 1$

      (*c*) $2^n - 1$

      (*d*) $2^n + 1$.

    (*ii*) In an array representation of binary tree, the leaf child of root will be at the location of

      (*a*) 2

      (*b*) 0

      (*c*) 1

      (*d*) 3.

(*iii*) Name the tree which has only one node in every level.

    (*a*) strictly binary tree

    (*b*) complete binary tree

    (*c*) skew tree

    (*d*) almost complete binary tree.

(*iv*) In a digraph, the number of edges that are incident on it is called

    (*a*) out degree

    (*b*) in degree

    (*c*) weighted graph

    (*d*) directed graph.

(*B*) Answer the following :   2 x 6

(*i*) List out some string operations ?

(*ii*) What do you mean by recursion ?

(*iii*) Write a recursive algorithm for pre-order tree traversal.

(*iv*) What do you mean by file structure ?

(*v*) Write a short note on inverted files.

(*vi*) Give a suitable example each for queue and stack.

DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A**,
ANY TWO *from* **Group B** *and* ALL *from* **Group C**.

*All parts of a question (a, b, etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.*

Figures on the right-hand side margin indicate full marks .

Group A

1. (*a*) What are the operations that can be performed on
a static list. Write the procedure to insert an element
at specified position in a list. 10

(*b*) Discuss the heapsort method of sorting a list. What
is its time complexity. Show its operation on the list
containing elements 42, 32, 48, 20, 58, 53, 75,
53, 97. 10

2. (*a*) Write a procedure to find the presence of one string
in another string. 10

(*b*) Define the following in brief : 5 x 2

(*i*) Time complexity

(*ii*) Space complexity

(*iii*) Array representation of strings

(*iv*) Record

(*v*) Abstract data type.

3. (*a*) Consider the tree given below :   6 + 2 + 2
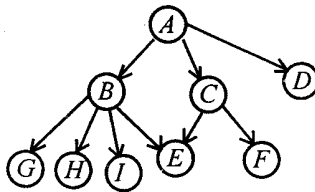
For this tree, find

(*i*) degree of each node of the tree ;

(*ii*) degree of tree ; and

(*iii*) level of each node of the tree.

(*b*) Write an algorithm to merge two circular lists, *A* and *B*, to produce a resultant list *C*.   10

4. Explain any *four* of the following :   5 x 4

(*i*) Applications of a queues

(*ii*) Bubble sort technique

(*iii*) Operations that can be performed on the string

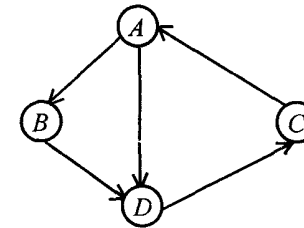(*iv*) Hashing

(*v*) Sequential search.

Group B

5. (*a*) Define a binary tree. Represent the array $X = [A, B, C, D, E]$ in binary tree representation.   10

(*b*) Consider a tree having pre-order and in-order traversal as follows. Draw the tree and write down its post-order traversal.   10

In-order   : *Z, A, Q, P, Y, X, C, B*
Pre-order : *Q, A, Z, Y, P, C, X, B*

6. (*a*) For following graph, obtain adjacency matrix as well as adjacency list representation :   10

(*b*) Explain various collision resolving techniques encountered in hashing function.   10

7. (*a*) What are binary plus trees ? Explain various applications of trees .   10

(b) A tape drive has following parameters:

    Bit density   :   2000 bits/cm
    Tape speed   :   8 cm/s
    Minimum time spent at : 3 min
        an inter-record gap
    Average record length :   4000 characters
    Estimate the time required to read a file of 100
    records.   10

8. Write short notes on any *four* of the following: 5 × 4

    (i) Indexed files;

    (ii) Hashed files;

    (iii) Radix sort;

    (iv) Threaded binary tree;

    (v) Tail recursion.

Group C

9. Choose the *correct* answer for the following: 2 × 10

    (i) In a linked list, there are no NULL links in

        (a) singly linked list

        (b) linear doubly linked list

        (c) circular linked list

        (d) None of the above.

(ii) In a stack command to access, the nth element from
     top of stack will be

     (a) $S[\text{TOP} - n]$

     (b) $S[\text{TOP} + n]$

     (c) $S[\text{TOP} - n - 1]$

     (d) None of the above.

(iii) If XXX and ZZZ are elements of a lexically ordered
      binary tree, then in pre-order traversal which node
      will be traversed first?

      (a) XXX

      (b) YYY

      (c) ZZZ

      (d) cannot be determined.

(iv) In a balanced binary tree, the height of two sub-trees
     of every node never differ by more than

     (a) 2

     (b) 1

     (c) 0

     (d) 3.

(*v*) The result of evaluating prefix expression
$* / b + - \ dacd$, where $d = 3$, $b = 6$, $c = 1$ and $d = 5$
is

(*a*) 8

(*b*) 5

(*c*) 10

(*d*) 15

(*vi*) In array representation of binary tree, the right child
of root will be at location of

(*a*) 2

(*b*) 5

(*c*) 3

(*d*) 0

(*vii*) A dummy header in linked test contains

(*a*) first record of actual data

(*b*) last record of actual data

(*c*) pointer to the last record of actual data

(*d*) None of the above.

(*viii*) Write the output of following program;
int a [ ] = {1, 2, 3}, $*p$;

(*b*) Junk value

(*c*) Run time error

(*d*) Address of third element.

(*ix*) If the out degree of every node is exactly equal to
m or 0, and the number of nodes at level k is m^{k-1},
then the tree is called as

(*a*) full m ary tree

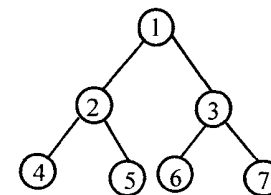(*b*) complete m ary tree

(*c*) positional m ary tree.

(*A*) only (*a*)

(*B*) only (*c*)

(*C*) Both (*a*) and (*c*) above

(*D*) Both (*b*) and (*c*) above.

(*x*) In the following tree, if post-order traversal generates
sequences $xy - zw * +$, then level of nodes 1 2 3 ....
will be

(a) $+, -, *, x, y, z, w$

(b) $x, -, y, +, z, *, w$

(c) $x, y, z, w, -, *, +$

(d) $-, x, y, +, *, z, w$

DATA STRUCTURES

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A**,
ANY TWO *from* **Group B** *and* ALL *from* **Group C**.

*All parts of a question (a, b, etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing or wrong data may be assumed suitably giving
proper justification.*

Figures on the right-hand side margin indicate full marks .

Group A

1. (a) Compare linked list and linear array representations
    of a data structure.   8

    (b) What do you mean by efficiency of an algorithm ?
    How can you compare the efficiency of two
    algorithms ? Explain the concept of best case, average
    case and worst case time complexity.   12

2. (a) Write an algorithm for quicksort. Also, apply the
    algorithm on the following sequence :

    25, 10, 30, 15, 20, 28, 18, 12

    Show all intermediate steps.   12

(b) Find all approachable nodes from source 2 using DFS.   8
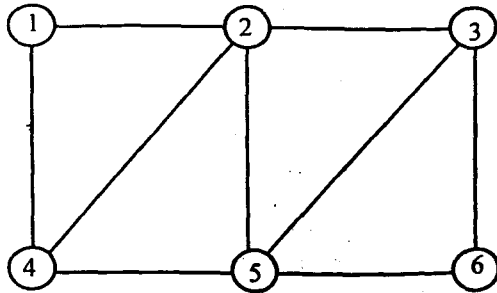
3. (a) How are stacks useful in expression evaluation? Give illustrative example to explain the concept.   12

(b) What is the purpose of keeping header node in a linked list?   4

(c) Given a matrix A of size $m \times n$, write a function that returns the sum of elements above the main diagonal, i.e., those elements a_{ij} for which $i < j$.   4

4. (a) Order the following functions by growth rates:

$$N, \sqrt{N}, N^{1.5}, N^2, N \, LogN, 2/N, 2^N,$$
$$N^2 \log N, N^3, N \log \log N.$$
   10

(b) Write a function using a stack that converts an expression from infix to postfix.   10

Group B

5. (a) What are height balanced trees? What is their advantage? Insert the following nodes one-by-one in the height balanced tree and balance the tree at every step:

$A, B, C, D, F, G, E, H, I, K, L, Q, I, F$ 15

(b) How are graphs represented in memory of a computer? Give relative merits and demerits of these representation schemes.   5

6. (a) For the following binary tree, answer the following: 2 × 8
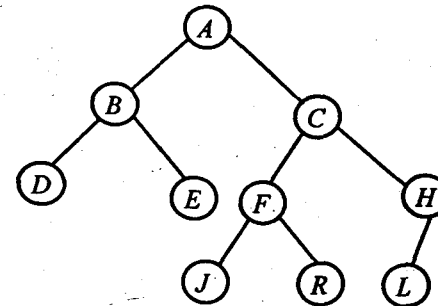
(i) Which nodes are leave nodes?

(ii) Which is root of the tree?

(iii) What is height of the tree?

(iv) Which are non-leave nodes?

(v) What is pre-order traversal?

(*vi*) What is in-order traversal?

(*vii*) What is post-order traversal?

(*viii*) Which are descendents of node C?

(*b*) Discuss applications of a graph.   4

7. (*a*) What are indexed files? How is a B^+ tree used to implement indexes? Explain with an example.   10

    (*b*) Compare sequential, relative and indexed file organizations. Also, give their applications with examples.   10

8. (*a*) How can a heap be represented using an array? Write an algorithm for heap-sort (using max-heap).   12

    (*b*) Compare DFS and BFS searching techniques.   8

Group C

9. Answer the following in brief: 2 x 10

(*i*) What are strings? How are strings represented in memory of a computer?

(*ii*) What is a queue? Is queue a linear or non-linear data structure?

(*iii*) Define a node.

(*iv*) What are ways to thread a binary tree? Explain briefly.

(*v*) Compare insertion and deletion in B-tree vs. B^+ tree.

(*vi*) What is hashing? What is its advantage?

(*vii*) What are common operations on data structures.

(*viii*) Distinguish between open and closed hashing.

(*ix*) Distinguish between a graph and a tree.

(*x*) Why is bubble sort algorithm so called?

DATA STRUCTURE

Time : Three hours

Maximum Marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A**,
ANY TWO *from* **Group B** and ALL *from* **Group C**.

*All parts of a question (a, b, etc.) should
be answered at one place.*

*Answer should be brief and to-the-point and be supplemented
with neat sketches. Unnecessary long answers may
result in loss of marks.*

*Any missing data or wrong data may be assumed suitably giving
proper justification.*
Figures on the right-hand side margin indicate full marks.

Group A

1. (*a*) Write the algorithm for matrix addition and find its
time complexity.   5

(*b*) Solve the following recurrence relation by expansion

$$T (n) = T (n /2) + 1$$
$$T (1) = 1$$

assume that n is an integer and $n \geq 1$.   10

(*c*) Write an algorithm to convert an infix expression
which include ',' '+', '-', '*' and '/' to postfix.   5

(Turn Over)

2. (a) Design the data structure that supports the following operations INSERT and MIN. The worst-case run time should be O(1) for each of these operations. 10

(b) Explain the concept of STACKS. Mention 3 applications of STACK. 5

(c) With the help of suitable example explain the concept of tail recursion. 5

3. (a) Describe an algorithm to evaluate postfix expression using STACK. 10

(b) What is the difference between circular linked list and doubly link list. Mention the applications of each type of list. 10

4. (a) What is a dequeue? Give an option between a linear array and circular array, which one will you choose to implement an array. Justify your answer. 10

(b) Consider two strings $X = x_1, x_2, \ldots x_n$ and $Y = y_1, y_2, \ldots y_n$ where x_i, $1 \leq i \leq n$ and y_j, $1 \leq j \leq m$ are members of finite set symbols. Write an algorithm to generate a string by taking 1 element from each list. When any one string is exhausted, the output string should store rest of the elements of other string. 10

· Group B

5. (a) What is the need of threaded binary tree? Describe its node structure. 10

(b) Consider following list:

$a[1, 10] = (310, 285, 179, 625, 351, 423, 861, 254, 450, 520)$

Sort the above list using merge sort. Show all intermediate steps. 10

6. (a) What do you understand by 'Garbage'. Explain how garbage collection method is used for allocating and freeing memory storage? 10

(b) Define the following in brief:

(i) AVL trees

(ii) B - tree

also, insert the keys below inorder to build an (I) AVL tree (II) B - tree of order 4
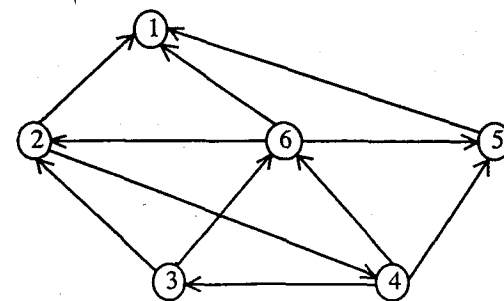
$a, g, f, b, k, d, h, m, j, e, s.$ 10

7. (a) For the digraph of figure below obtain:

(i) the indegree and outdegree of each vertex

(ii) its adjacency matrix representation

(iii) its adjacency list representation

(iv) its strongly connected component. 10

(b) Explain following with suitable example :   10

  (i)   Midsquare Hashing

  (ii)   Direct Hashing

  (iii)   Division remainder hashing

  (iv)   Fibonacci Hashing.

8. (a) Write an algorithm for depth first search.   4

   (b) Write an algorithm for Fibonacci search. Compare Fibonacci search with binary search.   10

   (c) Give an algorithm to count the number of leaf nodes in a binary tree t. What is its computing time and complexity ?   6

Group C

9. Answer the following questions :   2 × 10

  (i) How many real numbers are their between two floating point numbers ?

  (ii) Which sorting technique is preferred for sorting 10 integers, bubble sort or quick sort ?

  (iii) A strictly binary tree has n leaf nodes. How many nodes does it contain ?

  (iv) Draw the binary expression tree that represent the following postorder expression

$$ab + c / d *$$

(v) What is run time complexity of shellsort ?

(vi) List any four differences between tree and graph.

(vii) What do you understand by stability of a sorting alogorithm ?

(viii) What is loop invariant ? Explain with suitable example.

(ix) Mention two difference between inverted files, hashed file.

(x) Show that following equality is incorrect

$$n^2 / \log n = \theta (n^2).$$

W'05 : 4 FN : CP 403 (1448)

DATA STRUCTURES

Time : Three hours

Maximum marks : 100

Answer FIVE *questions, taking* ANY TWO *from* **Group A,**
ANY TWO *from* **Group B** *and* ALL *from* **Group C.**

All parts of a question (*a, b, etc*) *should be
answered at one place.*

*Answer should be brief and to-the-point and be supple-
mented with neat sketches. Unnecessary long answers
may result in loss of marks.*

*Any missing data or wrong data may be assumed suitably
giving proper justification.*

Figures on the right-hand side margin indicate full marks.

Group A

1. (*a*) Compare arrays and lists. State where they stand in
        context of speed, space, reliability, insertion,
        deletion and access operations.   10

    (*b*) Write two Different algorithms to implement a
        stack using array and using link list.   5 + 5

2. Write the implementation of a queue. What do you
    mean by priority queues ? Explain the difference. 10 + 5 + 5

---

3. Write a program using recursion for   10 + 10

   (a) Calculating the factorial of a given number

   (b) Calculating the greatest common divisor of a given number.

4. Why we always prefer worst case complexity of an algorithm? Why we do it priori instead of posterior analysis? Take the example of a sorting technique.   10 + 10

Group B

5. Write an algorithm / program for *any four* out of the following: 5 + 5 + 5 + 5

   (a) Bubble sort

   (b) Selection sort

   (c) Insertion sort

   (d) Quick sort

   (e) Merge sort.

6. What is a binary search tree? How it is different from a binary tree? Write an algorithm for the preorder, inorder and postorder traversal of a tree.   4 + 4 + 12

7. Explain *any four* out of following: 5 + 5 + 5 + 5

   (a) B-Trees

   (b) Inverted files

   (c) Hashing

   (d) Radix sort

   (e) Binary search.

8. Explain with example linear probing quadratic probing and bucket chaining techniques for collision avoidance in hashing.   7 + 7 + 6

Group C

9. Explain the following terms: 2 × 10

   (i) Dequeues

   (ii) Algorithm for doing reverse of a string

   (iii) Use of header and tail nodes in the linked list

   (iv) Where we should not use recursion

   (v) Sequential files

   (vi) Direct access files

   (vii) Height balances trees

   (viii) Five applications of tree data structure

   (ix) Breadth first search technique

   (x) Heap sort.