(i) An AVL Tree which is initially empty.
(ii) A red-black tree which is initially empty.
(iii)A 4-way search tree which is initially empty.

(4 + 4 + 4) = 12

7. (a) (i) Show the Depth First Search (DFS) Traversal Sequence of the following graph (Fig. 1), starting from the source vertex numbered 1, along with the discovery and finishing time of each vertex of the graph.

(ii) Draw the Depth First Tree (DFT) with respect to the DFS traversal of the graph (Fig. 1), from the source vertex 1 and classify all edges in the graph.

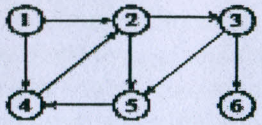

Fig. 1

(b) Show each step of insertion in a B-Tree of order 3, for the following sequence of numbers: 33, 4, 67, 12, 8, 9, 0, 34, 2, 7 and 21.

(3 + 4) + 5 = 12

Group - E

8. (a) Write a recursive function to implement binary search.

(b) Analyse the complexity of the binary search algorithm using recurrence relation.

(c) Can we apply binary search to search a key element in a single linked list? Support your answer with suitable reasons.

4 + 5 + 3 = 12

9. (a) Define the structure a maximum priority queue (max-heap) having two components: array of data elements and size. Write algorithm for dequeue function of this maximum priority queue, having appropriate underflow checking, as required.

(b) (i) What are the effective changes in an array, after partitioning it once, using one of its elements as pivot?

(ii) Write an algorithm for partitioning an array into two halves, using one of the array elements as pivot.

6 + (2 + 4) =12

10

B.TECH/CSE/3RD SEM/CSEN 2101/2016

DATA STRUCTURE & ALGORITHMS
(CSEN 2101)

Time Allotted : 3 hrs **Full Marks : 70**

Figures out of the right margin indicate full marks.

Candidates are required to answer Group A and
any 5 (five) from Group B to E, taking at least one from each group.

Candidates are required to give answer in their own words as far as practicable.

Group – A
(Multiple Choice Type Questions)

1. Choose the correct alternative for the following: **10 × 1 = 10**

(i) The number of edges in a complete undirected graph with n vertices is
(a) n(n-1) (b) n(n-1)/2 (c) n^2 (d) 2n-1.

(ii) Which of the following tree can be most effectively represented by an array?
(a) AVL Tree (b) Binary Search Tree
(c) Almost complete Binary Tree (d) Threaded Binary tree.

(iii) While doing a sequential search in an array, if the value being sought occurs only once in the array, the expected number of comparisons required is -
(a) (n -1)/2 (b) (n + 1)/2
(c) n - 1 (d) log$_2$ n.

(iv) The complexity of Bubble sort algorithm is
(a) O(n) (b) O(log n)
(c) O(n^2) (d) O(n log n).

(v) If a set of sorted integers is inserted in a Binary Search Tree then to search a certain item it's time complexity will be
(a) O(n) (b) O(log(log n))
(c) O(log n) (d) O(n^2).

(vi) Suppose we are sorting an array of eight integers using heap sort, and we have just finished some heapify (either maxheapify or minheapify) operations. The array now looks like this:
16 14 15 10 12 27 28
How many heapify operations have been performed on root of heap?
(a) 1 (b) 2 (c) 3 or 4 (d) 5 or 6.

(vii) Consider a situation where swap operation is very costly. Which of the following sorting algorithms should be preferred so that the numbers of swap operations are minimized in general?
(a) Heap Sort (b) Selection Sort
(c) Insertion Sort (d) Merge Sort.

(viii) The technique of linear probing for collision resolution can lead to
(a) overflow (b) clustering
(c) underflow (d) memory leakage.

(ix) Binary search algorithm cannot be applied to
(a) sorted linked list (b) sorted binary trees
(c) sorted linear array (d) pointer array.

(x) In case of an almost already sorted array as input, which of the following will be the fastest sorting technique
(a) Quick Sort (b) Insertion Sort
(c) Selection Sort (d) Cocktail Shaker sort.

Group - B

2. (a) What do you understand by the term "Abstract Data Type"?

(b) An array X [-15..........10, 15...............40] requires one byte of storage. If beginning location is 1500 determine the location of X [15][20].

(c) What is the complexity of the following code segment and why?

```
for (i = n; i > 0; i /= 2)
    printf ("Value of i = %d", i);
```

3 + 4 + (1 + 4) = 12

3. (a) Define appropriate structure for Doubly linked List.
Write an algorithm to perform the following operation of this Doubly Linked List: (i) insert at beginning, (ii) delete at end.

(b) Write an algorithm to reverse a singly linked list (strictly in a single left to right traversal of the list).

(3 + 3) + 6 = 12

Group - C

4. (a) Define a structure for circular queue using array and then write algorithm of the following circular queue operations : i) enqueue with overflow checking, ii) dequeue with underflow checking.

(b) Show the content of only operator stack, whenever push or pop occurs in the stack, while doing infix to postfix conversion of the following expression 7*9/(4+7) + (3*(7-3)).

(c) Write a recursive function for Tower of Hanoi problem.

(3 + 3) + 3 + 3 = 12

5. (a) Write a method duplicatestack(S) that returns a new stack containing the same elements and in the same order as the stack passed as parameter. The purpose of the method is to create a new stack and fill it with same data elements as of the given stack using the given stack and its standard functions. Before the method ends, it must restore the content of the original stack to its original state (same contents in same order). Beside the new stack that the method returns, the only additional data structure that can be used is a single queue.

(b) Write suitable functions to implement two stacks in a single array, where no stack overflow happens until the entire array space is exhausted?

(c) Given a stack S, sort it using recursion. Use of any loop constructs like while, for, etc. is not allowed. Only the following ADT functions can be used on Stack S

is_empty(S): Tests whether stack is empty or not.
push(S) : Adds new element to the stack.
pop(S) : Removes top element from the stack.
top(S) : Returns value of the top element. Note that this function does not remove element from the stack.

Example:

Input:	Output:
-3 <--- Top	30 <--- Top
14	18
18	14
-5	-3
30	-5

4 + 4 + 4 = 12

Group – D

6. Consider the following sequence of keys 40, 15, 65, 35, 55, 45, 75, 95, 85, 05, 30. Consider the insertion of the given keys in the order given into the following: