UNIT 5:
SEARCHING

1. Define searching.
2. Write a C function to implement linear search?
3. Write a C function to implement binary search?
4. Give the worst case and best case time complexity of binary search.
5. What is the precondition on a list of elements to apply binary search?
6. Differentiate between linear search and binary search.

SORTING

1. Define sorting.
2. Define internal sorting and external sorting with example.
3. What do you understand by stable sorting?
4. What do you understand by in-place sorting and out-place sorting?
5. Create a heap and sort the following elements using heap sort
   12, 8, 10, 6, 4, 10, 6, 11, 9, 8, 14, 1
6. Write an algorithm for sorting a set of integers using quick sort. What is the case average time complexity of the procedure?
7. Write an algorithm for insertion sort. Trace your algorithm on the following data to sort the list: 77, 33, 44, 11, 88, 22, 66, 55
8. Write an algorithm for merge sort. Using the algorithm, sort the following numbers in ascending order:
   i. 10, 25, 16, 5, 35, 48, 8
   ii. 30, 12, 38, 8, 5, 15, 1, 40
   iii. 75, 10, 20, 70, 80, 90, 100, 40, 30, 50
   iv. 15, 10, 5, 20, 25, 30, 40, 35
9. Write heap sort algorithm. Analyze the running time of your algorithm.
10. Use quick sort algorithm to sort the following numbers:
    i. 15, 22, 30, 10, 15, 64, 1, 3, 9, 52.
    ii. 12, 5, 14, 2, 56, 7, 85, 51, 18, 1, 75, 42, 1, 9.
    iii. 22, 55, 6, 7, 3, 66, 89, 56, 49, 65, 34, 67.
    iv. 68, 70, 75, 80, 84, 60, 50, 50, 45
    v. 38, 81, 22, 48, 13, 69, 93, 14, 45, 58, 79, 72
    Is it is stable sorting algorithm. Justify your answer.

11. Perform Heap sort on the following list of integers:

- **i.** 23, 5, 47, 58, 4, 52, 15, 48, 26, 3, 11, 4, 7
- **ii.** 25, 57, 48, 37, 12, 92, 86, 33

12. Using the bubble sort algorithm find the number of swaps (interchange) required to sort: 5, 10, 15, 55, 45, 35, 60, 75, 70

13. Which sorting algorithm is best if an array is already sorted? Justify your answer.

14. Write short on the following:

- **i.** Radix sort
- **ii.** Selection sort
- **iii.** Insertion sort

Binary Search Tree (BST)

1. What is a Binary Search Tree? Draw the Binary Search Tree when following keys are inserted in order in the initially empty binary search tree.

   5, 75, 19, 36, 8, 62, 49, 84, 12, 18, 25

   How can a Binary Search Tree (BST) be used for sorting of keys?
2. Construct a Binary Search Tree from the given values. Consider the first value as the root value. Values: 49, 22, 25, 90, 82, 7, 13, 47, 49, 63
3. How many maximum comparisons required in searching an element in a binary search tree?
4. What will happen if a binary search tree if left oriented or right oriented? Explain the problem and give the solution.
5. What will happen if a binary search if left oriented or right oriented?
6. Write an algorithm for inserting a node in Binary Search Tree. Suppose the following 10 members are inserted in order into an empty binary search tree T: 50, 48, 35, 44, 80, 70, 10, 55, 11, 85. Draw the tree T.
7. Define BST. Create BST for the following data, show all steps:

   20, 10, 25, 5, 15, 22, 30, 3, 14, 13
8. Explain Deletion in Binary Search Tree with example.
9. Write a C function to insert elements in a BST.
10. Write a C function to delete elements in a BST.
11. Write the important applications of binary search tree.
12. Explain different operations in a binary search tree.
13. Make a binary search tree for the following sequence of numbers, show all steps: 45, 32, 90, 34, 68, 72, 15, 24, 30, 66, 11, 50, 10

14. Draw the binary search tree that results from inserting into an initially empty tree records with keys given below in order.

E, A, S, Y, Q, U, E, S, T, I, O, N and then delete the Q

15. To determine whether two binary search trees on the same set of elements have identical tree structures, one could perform an in-order tree traversal on both and compare the output lists. Justify your answer.

16. Suppose that we have numbers between 1 and 1000 in a Binary Search Tree and want to search for the number 363. Which of the following sequences could not be the sequence of nodes examined? Explain your answer.

(i) 2, 252, 401, 398, 330, 344, 397, 363

(ii) 924, 220, 911, 244, 898, 258, 362, 363

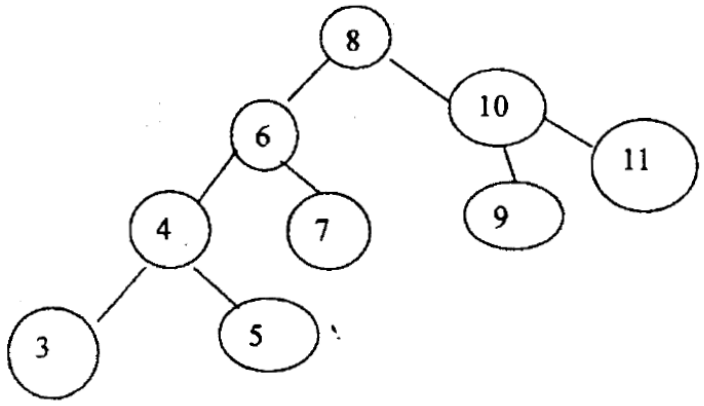
(iii) 925, 202, 911, 240, 912, 245, 363

(iv) 2, 399, 387, 219, 266, 382, 381, 278, 363

(v) 935, 278, 347, 621, 299, 392, 358, 363

AVL Tree

1. What is an AVL tree? Discuss the various kinds of rotations done for rebalancing the tree after insertion. Choose suitable example for illustration.
2. Explain all the unbalancing cases of AVL tree with examples showing balance factors of its node.
3. Take an example and show INSERTION and DELETION in an AVL tree.
4. Construct an AVL tree with the values 10 to 1 numbers into an initially empty tree.
5. What is the maximum height of any AVL tree with 7 nodes?
6. Consider the following AVL tree and insert 2, 12, 7, and 10 as new node. Show proper rotation to maintain he tree as AVL.



7. Construct the AVL tree with the following keys:
   - **i.** 35, 36, 80, 85, 67, 89, 25, 16, 10, 14
   - **ii.** 64, 1, 14, 26, 13, 110, 98, 85
   - **iii.** 8, 15, 1, 19, 16, 4, 25, 12, 23, 20, 17
8. Construct a height balanced Binary search tree by performing following operations:

   Step 1: Insert
   19, 16, 21, 11, 17, 25, 6, 13

   Step 2: Insert
   3

   Step 3: Delete
   16

B-Tree

1. Define B-tree. What do you understand by order of B-tree? Consider the following B-tree of order 3

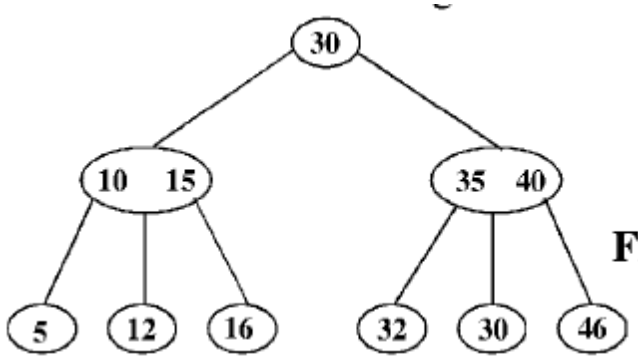


Fig. 2

Show the B-tree after the following operations: Insert 43, Insert 50 and delete 15.

2. What is m-way search tree? Construct B-tree from the following keys:
65, 71, 70, 66, 75, 68, 72, 77, 74, 69, 83, 73, 82, 88, 67, 76, 78, 84, 85, 80

3. What are the advantages of B+ tree over B tree?

4. Define the properties of B tree of order m.

5. Construct B tree of order 3 for the following elements:

- **i.** 92, 24, 6, 7, 11, 8, 22, 4, 5, 16, 19, 20, 78
- **ii.** 10, 20, 30, 40, 50, 60, 70, 80, 90

6. Obtain the minimum number of entries that can be made in a B-tree of order *m* and of level *l*.

7. Explain INSERTION and DELETION in B-tree.

8. What are the applications of B-tree?

9. Write short notes on B+ tree.

10. Compare and contrast the difference between B tee and B+ tree.

11. Draw a B-tree of order 4 by insertion of the following keys in order:
Z, U, A, I, W, L, P, X, C, J, D, M, T, B, Q, E, H, S, K, N, R, G, Y, F, O, V

HASHING

1. What is a hash function? Discuss various kinds of hash functions with examples. Further describe the approaches used to handle collisions.
2. What do you mean by Perfect Hash Function?
3. Discuss various methods used for resolving hash collisions.
4. What do you mean by hashing and collision? Discuss the advantages and disadvantages of hashing over other searching techniques.
5. What is a hash table? How using hash table is beneficial for us?
6. What are the types of Collision Resolution Techniques and the method used in each of type? Explain with suitable example.
7. Write the conditions when collision occurs in hashing.
8. What are the disadvantages of linear probing in hashing? Discuss how quadratic probing can be used to solve these problems.
9. Given input {4371, 1323, 6173, 4199, 4344, 9679, 1989} and a hash function h(X) = X (mod 10). Find memory location of each key and set them at memory from 1 to 10
   - **i.** Hash table with using linear probing and
   - **ii.** Hash table with quadratic probing.
   - **iii.** Hash table with second hash function h2 (x) = 7 − (x mod 7).

10. Write short notes on:
    - **i.** Linear probing
    - **ii.** Quadratic probing
    - **iii.** Double Hashing

STORAGE MANAGEMENT & COMPACTION

1. Write short notes on

- **i.** Storage Management
- **ii.** Garbage Collection
- **iii.** Compaction