Frequently Asked Questions:

a. What is a Binary Search Tree? its complexity?

Binary Search tree can be defined as a class of binary trees, in which the nodes are arranged in a specific order. ...

In a binary search tree, the value of all the nodes in the left sub-tree is less than the value of the root.

Similarly, value of all the nodes in the right sub-tree is greater than or equal to the value of the root.

Advantages of Binary Search Tree?

Searching become very efficient in a binary search tree since, we get a hint at each step, about which sub-tree contains the desired element.

The binary search tree is considered as efficient data structure in compare to arrays and linked lists. In searching process, it removes half sub-tree at every step. ...

It also speed up the insertion and deletion operations as compare to that in array and linked list.

c. What are the various operations that can be performed on Binary Search Tree?

Insert. Insert operation adds a new node in a binary search tree. The algorithm for the binary search tree insert operation is given below.

Delete. Delete operation deletes a node that matches the given key from BST. ...

Search. The search operation of BST searches for a particular item identified as â€œkeyâ€□ in the BST. ...

Traversals. We have already discussed the traversals for the binary tree. .

What are the difference between binary search tree and its variant binary trees

A Binary Tree follows one simple rule that each parent node has no more than two child nodes,

whereas a Binary Search Tree is just a variant of the binary tree which follows a relative order to how the nodes should be organized in a tree.

f. Applications of BST

Some of the major applications of BST are as follows:

BST is used to implement multilevel indexing in database applications.

BST is also used to implement constructs like a dictionary.

BST can be used to implement various efficient searching algorithms.

BST is also used in applications that require a sorted list as input like the online stores.

BSTs are also used to evaluate the expression using expression trees

EXPERIMENT 2

1. What is a tree? What are the properties of trees?

A tree is a hierarchical data structure defined as a collection of nodes. Nodes represent value and nodes are connected by edges.
A tree has the following properties:
The tree has one node called root. The tree originates from this, and hence it does not have any parent. Each node has one parent only but can have multiple children.

3. What are the rules to construct a binary tree?
In a Binary search tree, the value of left node must be smaller than the parent node, and the value of right node must be greater than the parent node.
This rule is applied recursively to the left and right subtrees of the root. Let's understand the concept of Binary search tree with an example.

5. Difference between recursive & Nonrecursive traversal?
A recursive function in general has an extremely high time complexity while a non-recursive one does not.
A recursive function generally has smaller code size whereas a non-recursive one is larger.

6. What are the rules to construct expression trees?
Following are the step to construct an expression tree: Read one symbol at a time from the postfix expression.
Check if the symbol is an operand or operator. Thus, An expression is created or constructed by reading the symbols or numbers from the left. If operand, create a node.

EXPERIMENT 3

a. What is traversing a tree
In computer science, tree traversal (also known as tree search and walking the tree) is a form of graph traversal and refers to the process of visiting

b. What is the various type of traversal and their advantages?
A binary tree is shown for the element 40, 56, 35, 48, 22, 65, 28.
Searching in Binary tree become faster.
Binary tree provides six traversals. Two of six traversals give sorted order of elements.

ADVANTAGE
Preorder traversal is used to create a copy of the binary tree.
It is used to get the prefix of an expression.
Binary tree traversals give quick searching, insertion, and deletion in cases where the tree is balanced.

c. What is mirror image of tree.
A tree is said to be a mirror image of itself if there exists an axis of symmetry through a root node that divides the tree into two same halves.

We use 2 copies of the binary tree say, b1 and b2 (to handle left and right subtree separately) and check if : root value of b1 and b2 are the same or not.

EXPERIMENT 4

a. What is a recursive algorithm?
A recursive algorithm is an algorithm which calls itself with "smaller (or simpler)" input values,
and which obtains the result for the current input by applying simple operations to the returned value for the smaller (or simpler) input.

b. What is stack?
In computer science, a stack is an abstract data type that serves as a collection of elements, with two main principal operations

c. Advantages of threaded Binary Tree?
Advantages of Threaded Binary Tree:
In threaded binary tree, linear and fast traversal of nodes in the tree so there is no requirement of stack.
It is more general as one can efficiently determine the successor and predecessor of any node .

d. What is the difference between binary trees and threaded binary trees?
B-TREE
BINARY TREE
Essential constraint A node can have at max M number of child nodes(where M is the order of the tree). A node can have at max 2 number of subtrees.
Used It is used when data is stored on disk.
It is used when records and data are stored in RAM.
Height of the tree logM N (where m is the order of the M-way tree)
log2 N
Application Code indexing data structure in many DBMS.
Code optimization, Huffman coding, etc.

EXPERIMENT 5

a. Can DFS be used to find shortest path?
DFS (Depth-First Search) will not necessarily give you the shortest path because it just explores one arbitrary path at a time - maybe you get lucky and that path is the shortest path, but generally it won't be.
It will give you the shortest path in a tree, but that's only because there's exactly one path to any given node

b. Why is BFS slower than DFS?
BFS stands for Breadth First Search is a vertex-based technique for finding the shortest path in the graph.

It uses a Queue data structure that follows first in first out.
In BFS, one vertex is selected at a time when it is visited and marked then its adjacent are visited and stored in the queue.
It is slower than DFS.

c. What is the time complexity of DFS
Complexity The time complexity of DFS is O (V + E) where V is the number of vertices and E is the number of edges.
This is because the algorithm explores each vertex and edge exactly once

d. What is DFS in coding?
A Distributed File System (DFS) as the name suggests, is a file system that is distributed on multiple file servers or multiple locations.
It allows programs to access or store isolated files as they do with the local ones, allowing programmers to access files from any network or computer

EXPERIMENT 6

What is Dijkstra algorithm example?
Dijkstra's algorithm (/Ëˆdaɪkstrəz/ DYKE-strəz) is an algorithm for finding the shortest paths between nodes in a graph

How is Dijkstra algorithm calculated?
Dijkstra's Algorithm works on the basis that any subpath B -> D of the shortest path A -> D between vertices A and D is also the shortest path between vertices B and D.
Each subpath is the shortest path Djikstra used this property in the opposite direction i.e we overestimate the distance of each vertex from the starting vertex

What is the complexity of Dijkstra algorithm
ijkstra's Algorithm Complexity. Time Complexity: O(E Log V) where, E is the number of edges and V is the number of vertices. Space Complexity: O(V)

) How do you solve Dijkstra's problem?
Below are the steps to be followed for solving using Dijkstraâ€™s algorithm: Convert any problem to its graph equivalent representation.
Maintain a list of unvisited vertices

EXPERIMENT 7

What is the time complexity of primâ€™s and Kruskalâ€™s algorithm
The time complexity of Primâ€™s algorithm is O(V 2). Conversely, Kruskalâ€™s algorithm runs in O(log V) time.
Conversely, Kruskalâ€™s algorithm runs in O(log V) time. In Primâ€™s algorithm, the adjacent vertices must be selected whereas Kruskalâ€™s algorithm does not have this type of restrictions on selection criteria.

Which algorithm (primâ€™s and Kruskalâ€™s) is better and why?
The advantage of Primâ€™s algorithm is its complexity, which is better than Kruskalâ€™s algorithm.
Therefore, Primâ€™s algorithm is helpful when dealing with dense graphs that have lots of edges.
However, Primâ€™s algorithm doesnâ€™t allow us much control over the chosen edges when multiple edges with the same weight occuR.

Applications of Minimum Spanning Tree.
The applications of Minimum Spanning Tree are:
Minimum Spanning tree is used to describe/ design a network.
Taxonomy.
Cluster analysis: clustering points in the plane, single-linkage clustering (a method of hierarchical clustering), graph-theoretic clustering and clustering gene expression data.
Constructing trees for broadcasting in computer networks

EXPERIMENT 8

What are the advantages of Sequential File Organization?
Advantages
The sequential file organization is efficient and process faster for the large volume of data.
It is a simple file organization compared to other available file organization methods.
This method can be implemented using cheaper storage devices such as magnetic tapes.
It requires fewer efforts to store and maintain data elements.
The sequential file organization technique is useful for report generation and statistical computation process.

What are serial and sequential files and how are they used in organization?
There are four methods of organizing files on a storage media.
This include: sequential, random, serial and indexed-sequential
Sequential file organization Records are stored and accessed in a particular order sorted using a key field.

Compare and contrast sequential file and random access file organization?
Sequential Access to a data file means that the computer system reads or writes information to the file sequentially, starting from the beginning of the file and proceeding step by step.
On the other hand, Random Access to a file means that the computer system can read or write information anywhere in the data file.

What are the primitive operations sequential files?
Storing and sorting in contiguous block within files on tape or disk is called as sequential access file organization.

In sequential access file organization, all records are stored in a sequential order. The records are arranged in the ascending or descending order of a key field.

EXPERIMENT 10

Define File? What are the factors affecting the file organization?
File Organization refers to the logical relationships among various records that constitute the file, particularly with respect to the means of identification and access to any specific record.
In simple terms, Storing the files in certain order is called file Organization.

In choosing a file organization for a particular file in a database, we should consider seven important factors:
Fast data retrieval.
High amount of work for processing data input & maintenance transaction.
Efficient use of storage space.
Protection from failures or data loss.
Minimizing need for reorganization.
Accommodating growth.
Security from unauthorized use.

Explain the different File opening modes in C++?
Different file opening modes in C++ are as follows:
Modes: Description: ios::in: Opens a file for reading. New contents cannot be written in the file. ios::out: Opens a file for writing.

Write a note on Direct Access File?
A filed-length logical record that allows the program to read and write record rapidly. in no particular order.
The direct access is based on the disk model of a file since disk allows random access to any file block'

What are the advantages of of direct access file.
Advantages:
This supports direct access to the blocks occupied by the file and therefore provides fast access to the file blocks.
It overcomes the problem of external fragmentation

EXPERIMENT 11

1. What is AVL tree?
In computer science, an AVL tree (named after inventors Adelson-Velsky and Landis) is a self-balancing binary search tree (BST).
It was the first such data structure to be invented. In an AVL tree, the heights of the two child subtrees of any node differ by at most one;
if at any time they differ by more than one, rebalancing is done to restore this property

In an AVL tree, at what condition the balancing is to be done
VL tree is a height-balanced binary search tree. That means, an AVL tree is also a binary search tree but it is a balanced tree.
A binary tree is said to be balanced if, the difference between the heights of left and right subtrees of every node in the tree is either -1, 0 or +1.

When would one want to use a balance binary search tree (AVL) rather than an array data
structure
The AVL trees are more balanced compared to Red-Black Trees, but they may cause more rotations during insertion and deletion.
So if your application involves many frequent insertions and deletions, then Red Black trees should be preferred.

EXPERIMENT 12

1. Why do we need Hashing
Importance of Hashing Hashing gives a more secure and adjustable method of retrieving data compared to any other data structure. It is quicker than searching for lists and arrays.

2. What is Hashing technique in DBMS?
Hashing is a DBMS technique for searching for needed data on the disc without utilising an index structure.
The hashing method is basically used to index items and retrieve them in a DB since searching for a specific item using a shorter hashed key rather than the original value is faster

3. What is the difference between HashMap and HashTable?

HashMap
Hashtable
1) HashMap is non synchronized. It is not-thread safe and can't be shared between many threads without proper synchronization code.
Hashtable is synchronized. It is thread-safe and can be shared with many threads.
2) HashMap allows one null key and multiple null values.
Hashtable doesn't allow any null key or value.
3) HashMap is a new class introduced in JDK 1.2.
Hashtable is a legacy class.
4) HashMap is fast.
Hashtable is slow.
5) We can make the HashMap as synchronized by calling this code
Map m = Collections.synchronizedMap(hashMap);
Hashtable is internally synchronized and can't be unsynchronized.
6) HashMap is traversed by Iterator.
Hashtable is traversed by Enumerator and Iterator.
7) Iterator in HashMap is fail-fast.
Enumerator in Hashtable is not fail-fast.

8) HashMap inherits AbstractMap class.
Hashtable inherits Dictionary class.

Explain Types of Hashing Techniques.
There are mainly two types of SQL hashing methods/techniques:
Static Hashing : In the static hashing, the resultant data bucket address will always remain the same
Dynamic Hashing : Dynamic hashing offers a mechanism in which data buckets are added and removed dynamically and on demand.
In this hashing, the hash function helps you to create a large number of values

Write down difference between Ordered Indexing and Hashing
In order Indexing addresses in the memory are sorted according to a critical value while in hashing addresses are always generated using a hash function on the key value.
Hash collision is a state when the resultant hashes from two or more data in the data set, wrongly map the same place in the hash table

EXPERIMENT 13

What is Binary Heap?
A binary heap is a heap data structure that takes the form of a binary tree. Binary heaps are a common way of implementing priority queues

Why is Binary Heap Preferred over BST for Priority Queue
Since Binary Heap is implemented using arrays, there is always better locality of reference and operations are more cache friendly.
Although operations are of same time complexity, constants in Binary Search Tree are higher.
We can build a Binary Heap in O(n) time. Self Balancing BSTs require O(nLogn) time to construct.
Binary Heap doesnâ€™t require extra space for pointers.
Binary Heap is easier to implement.

What is Binomial Heap?
A Binomial Heap is a set of Binomial Trees.
A Binomial Tree must be represented in a way that allows sequential access to all siblings, starting from the leftmost sibling (We need this in and extractMin () and delete ())

What is Fibonacci Heap?
In computer science, a Fibonacci heap is a data structure for priority queue operations, consisting of a collection of heap-ordered trees.
It has a better amortized running time than many other priority queue data structures including the binary heap and binomial heap

What is Heap Sort?
In computer science, heapsort is a comparison-based sorting algorithm.
Heapsort can be thought of as an improved selection sort: like selection sort,

heapsort divides its input into a sorted and an unsorted region, and it iteratively shrinks the unsorted region by extracting the largest element from it and inserting it into the sorted region.

What is Kth largest Element in an array
The index of kth Largest element = k-1 (zero-based indexing) The array can also be sorted in ascending order. The index of kth Smallest element = k-1 (zero based indexing) The idea is to construct a max-heap of elements