

1. What is the distinction between a list and an array?

a)

i) The main difference between list and array is that list is a collection of items which may contain elements of multiple datatypes where array has only homogeneous elements.

ii) list supports negative indexing whereas array does not support negative indexing.

2. What are the qualities of a binary tree?

a)

i) In binary tree each node can have at most 2 children.

ii) The maximum number of nodes present at level  $L$  of a binary tree is  $2^L$ .

iii) The maximum number of nodes in a binary tree of height  $H$  is  $2^{H-1}$ .

3) What is the best way to combine two balanced binary search trees?

a)

Method1:

Take all elements of first BST one by one, and insert them into the second BST. Inserting an element to a self balancing BST takes  $\log n$  time where  $n$  is size of the BST. So time complexity of this method is  $\log(n) + \log(n+1) + \dots + \log(m+n-1)$ . The value of this expression will be between  $m \log n$  and  $m \log(m+n-1)$ .

Method2:

1) Do inorder traversal of first tree and store the traversal in one temp array `arr1`. This step takes  $O(m)$  time.

2) Do inorder traversal of second tree and store the traversal in another temp array `arr2`. This step takes  $O(n)$  time.

3) The arrays created in step 1 and 2 are sorted arrays. Merge the two sorted arrays into one array of size  $m + n$ . This step takes  $O(m+n)$  time.

4) Construct a balanced tree from the merged array, This step takes  $O(m+n)$  time.

Time complexity of this method is  $O(m+n)$  which is better than method 1. This method takes  $O(m+n)$  time even if the input BSTs are not balanced.

4) How would you describe Heap in detail?

a)

A Heap is a special Tree-based data structure in which the tree is a complete binary tree. Generally, Heaps can be of two types:

Max-Heap: In a Max-Heap the key present at the root node must be greatest among the keys present at all of its children. The same property must be recursively true for all sub-trees in that Binary Tree.

Min-Heap: In a Min-Heap the key present at the root node must be minimum among the keys present at all of its children. The same property must be recursively true for all sub-trees in that Binary Tree.

5) In terms of data structure, what is a HashMap?

a)

Hash map is a widely used efficient data structure that used to store data which can be searched in constant time  $O(1)$ . It is also referred as hash table, unordered map. This data structure is implemented over an array that maps keys to values. Hence, hash map can be seen as a set of key value pairs. Each key is a number in the range of 0 to the array size  $- 1$ , generated by a hash function.

A good example for hash map is phone book. A phone book has names and phone numbers. In this case, the names are the keys, the phone numbers are the values.

6) How do you explain the complexities of time and space?

a)

The time complexity of an algorithm determines the amount of time taken by an algorithm to run as a function of the length of the input. The time to run is a function of the length of the input and not the actual execution time of the machine on which the algorithm is running on.

Space Complexity of an algorithm is the total space taken by the algorithm with respect to the input size.

7) How do you recursively sort a stack?

a)

We can recursively pop out each of the elements of the stack and then call a recursive function to insert the elements again in the stack in sorted order.