# Easy Questions:

1. Merge two sorted Arrays of integers into Single Array in O(n) time.
2. Second Largest number in an Array of integers in O(n) time.
3. String Reversal in O(n) time.
4. Check if String and list are same like: list = ["I","Like","Apple"] and str = "I Like Apple" are same in O(n) time.
5. Print nodes of Left view of tree and right view of binary tree.
6. Find duplicates in an array in O(n) time.
7. Find the occurrence of an integer in the array in linear time.
8. Write a program to sort the given array in nlog(n) time.
9. Add two numbers represented using Linked List and store the output in linked list.
10. Check if Singly Linked List is cyclic or not.
11. Calculate Height of Binary Tree.
12. Check Tree is Perfect Binary Tree or not. Perfect Binary tree is one where every node has either 0 or 2 children.
13. Check if tree is complete Binary tree or not.
14. Print the no. occurrences of alphabets in a string. Using arrays and time complexity should be O(n).

Input: s = "abcabcbb"

Output: {a:2, b:4, c:2}

Input: s = "bbbbb"

Output: {b:4}

Explanation: The answer is "b", with the length of 1.

1. Given a linked list of size N. The task is to reverse every k node (where k is an input to the function) in the linked list. If the number of nodes is not a multiple of k, then left-out nodes, in the end, should be considered as a group and must be reversed (See Example 2 for clarification).

Input:

LinkedList: 1->2->2->4->5->6->7->8

K = 4

Output: 4 2 2 1 8 7 6 5

Explanation:

The first 4 elements 1,2,2,4 is reversed first

and then the next 4 elements 5,6,7,8. Hence, the

resultant linked list is 4->2->2->1->8->7->6->5.

1. Find the very first non-repeating character in a string array

Input: String [] strArr={“apple”,”aappllee”,””,”aabbcde”,”kettle”};

Output: Character [] chArr={‘a’,’0’,’\u0000’,’c’,’k’};

# Medium Questions:

1. Given a list of lists that contains arrival and departure of N number of trains e.g.: {{A1,D1},{A2,D2}...{An,Dn}},  
   Calculate min number of platforms needed so that there are no conflicts.
2. Count Number of complete Nodes in a Binary Tree. Complete node is those who has both right and left child.
3. Find the missing number in an integer array of N-1 numbers such that it contains only distinct integers from 1 to N.  
   N=5, list --> {1,2,5,3}  
   Output: 4  
   N=10, {6,1,2,8,3,4,7,10,5}  
   Output: 9
4. Given an array of integers where each element represents the max number of steps that can be made forward from that element. Write a function to return the minimum number of jumps to reach the end of the array (starting from the first element). If an element is 0, they cannot move through that element. If the end isn’t reachable, return -1.  
   Input: arr[] = {1, 3, 5, 8, 9, 2, 6, 7, 6, 8, 9}  
   Output: 3 (1-> 3 -> 9 -> 9)  
   Explanation: Jump from 1st element to 2nd element as there is only 1 step, now there are three options 5, 8 or 9. If 8 or 9 is chosen then the end node 9 can be reached. So, 3 jumps are made.
5. Given an array of N integers, and an integer K, find the number of pairs of elements in the array whose sum is equal to K.
6. Check if Binary tree is Balanced Binary Tree or not.
7. Given a string s, find the length of the longest substring without repeating characters.

Example 1:

Input: s = "abcabcbb"

Output: 3

Explanation: The answer is "abc", with the length of 3.

Example 2:

Input: s = "bbbbb"

Output: 1

Explanation: The answer is "b", with the length of 1.

Example 3:

Input: s = "pwwkew"

Output: 3

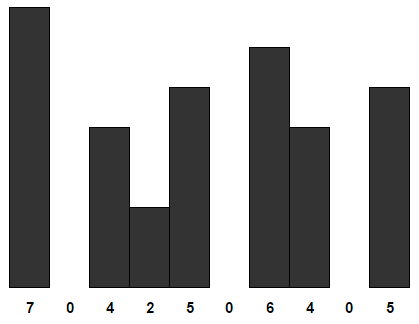
Explanation: The answer is "wke", with the length of 3.

Note that the answer must be a substring, "pwke" is a subsequence and not a substring.

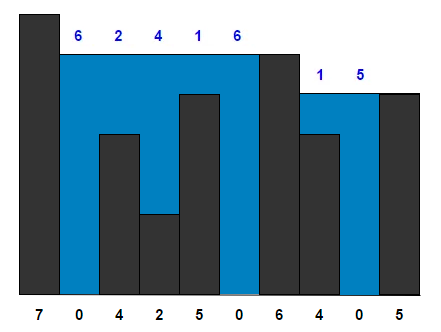
Example 4:

Input: s = ""

Output: 0

1. Find the maximum amount of water that can be trapped within a given set of bars where each bar’s width is 1 unit.  
   For example,  
   Input: An array containing height of bars {7, 0, 4, 2, 5, 0, 6, 4, 0, 5}

The maximum amount of water that can be trapped is 25, as shown below (blue).



# Algorithmic Questions:

1. Given an input string, write a function that returns the Run Length Encoded string for the input string.

For example,

if the input string is “wwwwaaadexxxxxx”,

then the function should

return “w4a3d1e1x6”

Expected Time Complexity: O(n)

1. Let 1 represent ‘A’, 2 represents ‘B’, and so on. Given a digit sequence, count the number of possible decoding’s of the given digit sequence.

Input:

digits[] = "121"

Output: 3

The possible decoding's are

"ABA", "AU", "LA"

Input:

digits[] = "1234"

Output: 3

The possible decoding's are

"ABCD", "LCD", "AWD"

Expected Time Complexity: O(n)

1. Given an array of strings strings, group the anagrams together. You can return the answer in any order.

An Anagram is a word or phrase formed by rearranging the letters of a different word or phrase, typically using all the original letters exactly once.

Input:

strs = ["eat","tea","tan","ate",

"nat","bat"]

Output:

[["bat"],["nat","tan"],

["ate","eat","tea"]]

Expected Time Complexity: O(n)

1. Given a file containing data of student name and marks scored by him/her in 3 subjects. The task is to find the list of students having the maximum average score.

If more than one student has the maximum average score, print them as per the order in the file.

Input:

file[] = {“Shrikanth”, “20”, “30”,

“10”, “Ram”, “100”, “50”, “10”}

Output:

Ram 53

Average scores of Shrikanth, Ram are 20 and 53 respectively. So Ram has the maximum average score of 53.

Expected Time Complexity: O(n)

1. Given a linked list, write a function to reverse every k nodes (where k is an input to the function).

Input:

1->2->3->4->5->6->7->8->NULL, K = 3

Output:

3->2->1->6->5->4->8->7->NULL

Expected Time Complexity: O(n)

Auxiliary Space: O(n/k).

1. Given a Binary Search Tree (BST) and a positive integer k, find the k’th largest element in the Binary Search Tree.
2. Given an unsorted array arr[0..n-1] of size n, find the minimum length subarray arr[s..e] such that sorting this subarray makes the whole array sorted.

Example: If the input array is [10, 12, 20, 30, 25, 40, 32, 31, 35, 50, 60], your program should be able to find that the subarray lies between indexes 3 and 8.

1. Given a binary array, in which, moving an element from index i to index j requires abs(i – j) cost. The task is to find the cost to move all 1s to each index of the given array.

Input:

arr[] = {0, 1, 0, 1}

Output:

4 2 2 2

Explanation:

Moving elements from index 1, index 3 to index 0 requires abs(0 – 1) + abs(0 – 3) = 4.

Moving elements from index 1, index 3 to index 1 requires abs(1 – 1) + abs(1 – 3) = 2.

Moving elements from index 1, index 2 to index 2 requires abs(2 – 1) + abs(2 – 3) = 2.

Moving elements from index 1, index 2 to index 3 requires abs(3 – 1) + abs(3 – 3) = 2.

Therefore, the required output is 4 2 2 2.

1. Problem of placing N chess queens on an N×N chessboard so that no two queens attack each other.

The expected output is a binary matrix which has 1s for the blocks where queens are placed.

For example, following is the output matrix for above 4 queen solution.

{ 0, 1, 0, 0}

{ 0, 0, 0, 1}

{ 1, 0, 0, 0}

{ 0, 0, 1, 0}

Solution: Use Recursion and Backtracking

1. Given an array of random numbers, push all the zeroes of a given array to the end of the array. For example, if the given array is {1, 9, 8, 4, 0, 0, 2, 7, 0, 6, 0}, it should be changed to {1, 9, 8, 4, 2, 7, 6, 0, 0, 0, 0}. The order of all other elements should be same.

Expected time complexity is O(n) and extra space is O(1).

1. Arrange the number in an array in the form of sine wave

ie alternate number have to be in increasing then decreasing order

Sample case:

1. 4,12,17,19,0,3,8  
     
   Ans  
   4 17 12 19 0 8 3  
     
     
   4 is smaller than 17  
   17 is bigger than 12  
   12 is smaller than 19  
   19 is bigger than 0  
   and so on an alternate sequence
2. Draw and explain LLD of a vending machine