

HASHING QUESTIONS

Question 1 :

Bottom View of a Binary Tree

The top view of a binary tree is the set of nodes visible when the tree is viewed from the top. Given a binary tree, print the top view of it. The output nodes can be printed in any order.

Sample Input :

```
      20
     /  \
    8    22
   / \   \
  5  3   25
   / \
  10 14
```

Sample Output : 5 10 3 14 25

Hint : Use the concept of Vertical Order

Question 2 :

Two Sum

Given an array of integers `arr[]` and an integer `target`, return indices of the two numbers such that they add up to `target`.

You may assume that each input would have exactly one solution, and you may not use the same element twice.

You can return the answer in any order.

Sample Input 1 : `arr = [2, 7, 11, 15]`, `target = 9`

Sample Output 1 : `[0, 1]`

As `arr[0] + arr[1] == 9`, we return `[0, 1]`.

Sample Input 2 : `arr = [3,2,4]`, `target = 6`

Sample Output 2 : `[1, 2]`

Question 3 :

Sort by Frequency

Given a string *s*, sort it in decreasing order based on the frequency of the characters. The frequency of a character is the number of times it appears in the string.
Return the sorted string. If there are multiple answers, return any of them.

Sample Input 1 : *s* = "cccaaa"

Sample Output 1 : "aaaccc"

Both 'c' and 'a' appear three times, so both "cccaaa" and "aaaccc" are valid answers.
Note that "cacaca" is incorrect, as the same characters must be together.

Sample Input 2 : *s* = "tree"

Sample Output 2 : "eert"

'e' appears twice while 'r' and 't' both appear once.

So 'e' must appear before both 'r' and 't'. Therefore "eetr" is also a valid answer.

BONUS (LRU Cache) IMPORTANT

Please go on the platform and solve this question : <https://leetcode.com/problems/lru-cache/>