

CWF FSE Screening Questions

What to ask from this Question Bank.

Coding Question - Question No 1

- One Coding question is **mandatory** to solve covering most of the edge cases. The code should be written in Java.

Concept Question - Question No (2 to 5)

- For questions 2 to 5 please ask any 2 questions from each section.

Guidance

- If the candidate finishes the question early and is doing well, please ask follow-up questions or additional questions to fill the remaining time and gather more information about their skills.

Q1 - Algorithm/Coding in Java - Ask any 1

Question

Given a sorted and rotated array `arr[]` and a key, write a function to find the key in the array where the array index starts from 0. **The expected time complexity is $O(\log n)$.**

Example:

Input: `arr[] = {5, 6, 7, 8, 9, 10, 1, 2, 3}`, `key = 3`

Output: Found at index 8

Input: `arr[] = {5, 6, 7, 8, 9, 10, 1, 2, 3}`, `key = 30`

Output: Not found

Input: `arr[] = {30, 40, 50, 10, 20}`, `key = 10`

Output: Found at index 3

```
{
```

```
    public static int search(int[] arr, int key){
```

```
        int low = 0;
```

```
        int high = arr.length-1;
```

```

while(low <=high){

    int mid = low + (low +high)/2;

    if(arr[mid]==key){
        return mid;
    }
    // left
    if(arr[low] <= arr[mid]){

        if(key >=arr[low] && key <=arr[mid]){
            high = mid-1;
        }
        else{
            low = mid +1;
        }
    }
    // right
    else{

        if(key > arr[mid] && key<=arr[high]){
            low = mid+1;
        }
        else{
            high = mid -1;
        }

    }

}

// not found the key case
return -1;
}

public static void main(String[] args){

    int[] arr = {5, 6, 7, 8, 9, 10, 1, 2, 3};

    int key = 3;

    int index = search(arr, key);

    if(index != -1){
        System.out.println("found at index"+index);
    }
    else{
        System.out.println("not found at index");
    }

}

}

```

Question

Given an array of distinct integers and a sum value. Find the count of triplets with a sum smaller than the given sum value. **The Expected Time Complexity is $O(n^2)$.**

Example:

Input: arr[] = {-2, 0, 1, 3}
sum = 2.

Output: 2

Explanation: Below are triplets with sum less than 2
(-2, 0, 1) and (-2, 0, 3)

Input: arr[] = {5, 1, 3, 4, 7}
sum = 12.

Output: 4

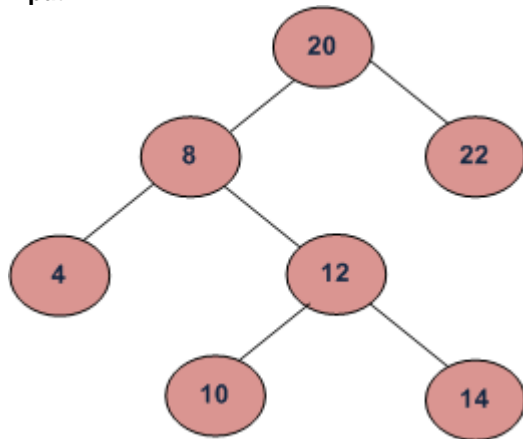
Explanation: Below are triplets with a sum less than 12
(1, 3, 4), (1, 3, 5), (1, 3, 7), and
(1, 4, 5)

Question

Given the root of a binary tree, the task is to print level order traversal in a way that nodes of all levels are printed in separate lines.

Example:

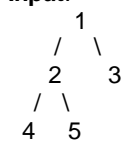
Input:



Output:

20
8 22
4 12
10 14

Input:



Output:

1
2 3
4 5

Question

Given a sorted array `arr[]` and a number `x`, write a function that counts the occurrences of `x` in `arr[]`. **Expected time complexity is $O(\log n)$**

Example:

Input: `arr[] = {1, 1, 2, 2, 2, 2, 3,}`, `x = 2`

Output: 4 // `x` (or 2) occurs 4 times in `arr[]`

Input: `arr[] = {1, 1, 2, 2, 2, 2, 3,}`, `x = 3`

Output: 1

Input: `arr[] = {1, 1, 2, 2, 2, 2, 3,}`, `x = 1`

Output: 2

Input: `arr[] = {1, 1, 2, 2, 2, 2, 3,}`, `x = 4`

Output: -1 // 4 doesn't occur in `arr[]`

Q2 - Java Concepts - Ask any 2

- Why is Java called the 'Platform Independent Programming Language'?
- Why is the main method static in java?
- Differences between HashMap and Hashtable in Java.

// both are using Map Interface

// hashmap : this is not thread safe if we have to use this then we can use manually synchronized or ConcurrentHashMap

// hashtable : This is thread-safe it is synchronized one and hashtable is faster than HashMap

- Why is method overloading not possible by changing the return type in java?
- Difference Between ConcurrentHashMap, Hashtable, and Synchronized Map in Java
- Explain JVM, JRE, and JDK?

// Java Virtual Machine

Take care for java byte code execution and take care for platform independency ,
memory management

// Java Run time Environment

It is a combined package of softwares and take care of JVM , some code library and support some file that help in code execution

// Java Development Kit : it is a software package i can say full featured software package where we can say get support java code till execution involve JRE

Q3 - Web API Concepts - Ask any 2

- What are the HTTP Methods?

// GET – retrieving data from the server as request

Example GET /username/ id

// POST – sending data to server as response

// PUT – Updating the existing data

// DELETE – Deleting the data

// PATCH – partial modification for data

- What is the main difference between POST, PUT and PATCH?
- What are the key components of an HTTP Request?
- What are HTTP Status codes?

// 100 – informational purpose

// 100 – continue

// 101 – switch protocols

// 200 - success case

// 200 ok status

// 201 created new resource

// 202 accepted now resource

// 300 – redirect page/ webpage purpose

// 301 – multiple choices redirection

// 301 moved permanently

// 302 – found resource

//400 - client side error status

//500 - server side error status

- What is REST API & Difference Between REST API and RPC API

Q4 - SQL Question - Ask any 2

- The Table name is Employee with columns - id, name, salary. Delete duplicate names keeping 1st name not deleted.

// DELETE FROM Employee

```
WHERE id NOT IN (  
    SELECT Min(id)  
    FROM employee  
    GROUP BY name  
)
```

- How to find Nth highest salary from a table

// let suppose table name is salary

```
SELECT Salary
```

```
FROM Employees E1
```

```
WHERE (N) =(
```

```
SELECT COUNT (DISTINCT E2.salary)
```

```
FROM employees E2
```

WHERE E2.salary >= E1.salary

)

- There is a table which contains two columns Student and Marks, you need to find all the students, whose marks are greater than average marks i.e., list of above average students.
- SQL to display the Department Name Having Highest Average Salary. The table name is Company and has id, name, department, and salary as columns.
- SQL to Print the Name and Salary of the Person Having the Least Salary in each Department. The table name is Company and has id, name, department, and salary as columns.

Q5 - JavaScript concepts (Full Stack Engineers) - Ask any 2

- Difference between “==” and “===” operators in JavaScript.
- Explain Closures in JavaScript.

// when a function enclose with internal child function and where we can see the child is having access of parent function scope , means can access argument and other scoping stuff that concept known as “Closures “ in javascript

//use case data privacy

- What are callbacks in JavaScript?
- What is the use of promises in JavaScript?
- Explain call(), apply() and bind() methods.

// call : used to invoke / run the function where with given “this “ and takes parameters separately

// apply : used to invoke / run the function where with given “this and take parameters as array

// bind : it does do function borrowing with given this scope and does not invoke/ run the function
unlike call , apply