|  |
| --- |
| **Anuj Arora (May 3rd, 2024)**  **Detailed explanation of each tool you have worked on and also differences between alternative tools.**  **Java questions on JVM memory, final keyword, multithreading.**  **Write java code for making immutable class and explain full functionality of each keyword in the code.**  **Write java code for deep copy and shallow copy and explain it briefly** |

**Anuj Arora (May 17the, 2024) - Round -1**

Design a data structure with following features: ( Coder pad)  
1. push an element at the end of the data structure (O(1) time complexity)  
2. Pop the element from the begining of the data structure( pop the earliest element added, O(1) TC)  
3. Seek an element at index i at any given state of the data structure. (O(1) TC)

**Sushanth Sharma & Seena (May 20the, 2024) - Round -2**

1.how kafka make sure the messages are published successfully.

 2.Publisher vs Consumer

 3. Consumer group and it's usage

 4. how to retry in case of failure and till when u will retry  (REST / Async  )

 5. When to use SQL Or  NoSQL ( difference and scaling and data type support)

 6. Horizontal & vertical scaling in SQL and NoSQL

 7. Cap theorem.

8. What are the different component of a eCommerce websites (Order , Auth ,Product,Cart, Profile, Payment etc)

9. In case of payment failure what are the services impacted.

10. Which services you consider for REST or for Event driven.

11. Will you make Payment services as Event driven or Rest.

12. Write  method/endpoint for you're services (/orders , /orders/:id etc)

13. How to Scale and How much horizontal scale need in SQL DB.

14. Kafka overview and use case.

15. Spring vs Springboot

**Syed Namdar Hussain(3rd May)**

**Round 1**

**1) Detailed questions related to my previous work from resume.**

**Some Example:**

**A. What is Event Driven System?**

**B. Why we use Event Driven System? (Pro and cons)**

**C. How do we maintain Idempotency? (with example)**

**2) What is singleton class?**

**3) Write down your own customize singleton class. (Take care all the scenario like serialization clone etc).**

**4) Some Java basic questions like:**

**A. Immutable class**

**B Some spring boot related questions. ( Beans Scope)**

**Round 2**

**1) All question was scenario based on my last project,**

**That was E-commerce platform.**

**a) First question was to explain design of microservice you have used.**

**b) Full explanation of event driven architecture.**

**c) When we have to use this type of architecture (Pros and Cons).**

**And lots of cross questions.**

**2) Questions on spring boot**

**a) Some scenario-based questions how to handle transaction.**

**b) Cascading related question**

**c) Mapping, how you manage OneToMany mapping in jpa.**

**d) Lazy and Eager uses in ORM tools.**

**YAWAR KHAN( May 07th, 2024)**

Questions from the previous project  
reverse string without replacing special character   
Hash set internal architecture  
make immutable class for employee with department class which we don't have the access  
arraylist and linkedlist difference  
Concurrent set and concurrent map  
which method will be called if we create object parameter and String parameter in two methods and will pass string value  
Exception heirarchy  
bean scopes  
default scope and how to change  
exception handling spring boot  
different autowired  
prototype bean vs singleton scope  
idempotent law  
solid  
design pattern strategy

Seena and Sushant(May,09th, 2024)  
  
Questions from the previous project  
Use AWS S3 bucket  
resiliency/scability maintains of mongodb  
Singleton Design Pattern  
Spring boot vs other framework  
mySQL and mongodb use cases  
Use cases for Redis  
HTML, Rest API, Protocol etc.

Interviewer: Mayank Jain (May,16th, 2024)

Candidate: Mohammad Rafique

Questions from previous project

2 coding question only asked from and asked in detail how the code works and debug it with each iteration and asked space and time complexity for each code. i.e.

Coding1:

Find the longest unique substring from a given String.

Eg: String input1 = "abcdabbe"

String output1 = "abcde"

String input2 = "abcdabbeff"

String output2 = "abcdef"

Coding 2:

Create a new array by replacing each integer value with next greatest number and if not present than set it with default value i.e. -1.

int[] input1 = [5,10,6,8,6]

int[] output1 = [10,-1,8,-1,-1]

Iteration 1:

int[] input1 = [5,10,6,8,6]

int[] output1 = [**10**,10,6,8,6]

explanation:  next greatest number for 5 is 10.

Iteration 2:

int[] input1 = [5,10,6,8,6]

int[] output1 = [10,**-1**,6,8,6]

explanation:  next greatest number than 10 is not present in the arrays so set it as **-1**.

Iteration 3:

int[] input1 = [5,10,6,8,6]

int[] output1 = [10,-1,**8**,8,6]

explanation:  next greatest number for 6 is 8.

Iteration 4:

int[] input1 = [5,10,6,8,6]

int[] output1 = [10,-1,8,**-1**,-1]

explanation:  next greatest number than 10 is not present in the arrays so set it as **-1**.

Iteration 5:

int[] input1 = [5,10,6,8,6]

int[] output1 = [10,-1,8,-1,**-1**]

explanation:  next greatest number than 6 is not present in the arrays so set it as **-1**.

Interviewer: Seena and Sushant (May 17th, 2024)  
Candidate: Mohammad Rafique  
  
Questions from the previous project in details with which service I have worked on.  
How Databases used till now in career  
All the Design Pattern used till now in career  
How you have used spring Boot, Microservice Kafka and Mongo in your project.  
Fault tolerance of the system  
How did you resolve Production issues till now in career (log management)  
Different types of testing. different tools used for this testing.  
HTML, Rest API, Protocol etc. how to handle post, put and patch in detail.  
How to create a Restful API from the scratch.

Interviewer: Mayank Jain (May,24th, 2024)

Candidate: Veerababu  
  
Questions from previous project.  
  
1,  Move all zeroes to start/end of array   
  
input : = {0,1,1,1,0,0,0,1,1}  
result = {0,0,0,0,1,1,1,1,1}  
result = [1,1,1,1,0,0,0,0]

2,  Design a data structure  to perform get() , delete() and  insert() in O(1) time.

3. Time complexity questions on O(1),O(n) and O(log n).

Interviewer: Mayank Jain (31st May 2024)

* Self-Introduction
* Coding Question: Explain approach, including complexity analysis.
  + input: arr1[] = {2,5,8,10,15}, arr2[] = {1,4,7,9,13}
  + output: result [] = {1,2,4,5,7,8,9,10,13,15}
  + Input: nums1 = [1,2,3,0,0,0] and nums2 = [2,5,6].  
    output: nums1= [1,2,2,3,5,6]
  + String input1 = "abcdae"  
    String output1 = "bcdae"
  + Given two sorted arrays, merge them to single sorted array.
  + Given Two sorted arrays and merge them into single sorted array without using extra space. he first array, nums1, is sized to fit m initialized elements and n additional elements (initialized to 0), where m and n are the lengths of nums1 and nums2, respectively. The challenge is to merge these arrays in-place in nums1.
  + Find the longest unique substring from a given String.
* Find first non- repeating character in a string .  Ex : input : "abaccd"  , output : "b"
* Given an input string . Find the longest repeating subsequence's first index and length of the subsequence. Ex : input : "aabbbc" ; output : index : 2, length : 3 ("bbb")

Interview: 13 June 2024 4:00 PM

Candidate: Ashish Pratap  
Round 1:

1. Given a string **str**, the task is to convert the given string into the number without using any inbuilt function.

**1.1.1. Problem Description**

You are given a string str that may contain a mixture of characters including digits, spaces, alphabetic characters, and signs ('+' or '-'). Your task is to convert the string into an integer following these specific rules:

* 1. **Leading Alphabets**: If the string starts with alphabetic characters, the function should return -100.
  2. **Leading Spaces**: Any leading spaces in the string should be ignored.
  3. **Sign Handling**: The string may contain a single '+' or '-' sign that indicates the sign of the resulting number. This sign should be considered only if it appears before any digits.
  4. **Digit Conversion**: Convert consecutive digits into the corresponding integer value.
  5. **Out of Range Handling**: If the resulting integer is outside the range of a 32-bit signed integer:
     + Return Integer.MAX\_VALUE (2,147,483,647) if the number is too large.
     + Return Integer.MIN\_VALUE (-2,147,483,648) if the number is too small.
  6. **Alphabets After Digits**: If any alphabetic characters appear after the digits, stop processing further characters and return the integer formed up to that point.

**1.1.2. Additional Constraints:**

* 1. You are not allowed to use any inbuilt functions for parsing or converting the string into an integer.
  2. You must handle edge cases explicitly as mentioned above.

**1.1.3. Example Cases:**

* 1. Input: " -42"
     + Output: -42
  2. Input: "4193 with words"
     + Output: 4193
  3. Input: "words and 987"
     + Output: -100
  4. Input: "-91283472332"
     + Output: Integer.MIN\_VALUE
  5. Input: "3.14159"
     + Output: 3
  6. Input: " +0 123"
     + Output: 0

1. You are given a binary tree with integer values, and your task is to determine if it is possible to split the tree into two separate trees by cutting one edge such that the sum of the nodes' values in both resulting trees is equal.  
   Input: [1, 2, 3]  
   Output: true

Interview: 14 June 2024 4:00 PM

Candidate: Ashish Pratap

Round 2:

1. List all hands-on software design patterns
2. Write Singleton design pattern
3. Write factory design pattern
4. how we handle transactions in spring
5. Write a method using @Transactional to handle credit/debit on accounts.
6. The behaviour of @Transactional when nested or same class also means of Propagation property.
7. CAP Theorem
8. List all SQL/No-SQL DB used
9. Describe each or divide it based on CAP
10. What is kafka? How does it function?
11. What is a topic? and partition?
12. Candidate: RajKumar Gautam
13. Round 1: Taken By Kuldip
14. Ques 1: Best Time to Buy and Sell Stock - LeetCode (You need to find index also)
15. Ques 2: Longest Substring Without Repeating Characters - LeetCode   
    At place of Character assume a list of tuple(userid, userName)   
    Test case: [{1,user1}, {2, user2}, {3, user3}, {4,user1}, {5,user3}]  
    output : 3
16. ({1,user1}, {2, user2}, {3, user3}) (This is the longest Subarray without repeating userName)
17. Round 2: Taken By Seena and Sushant
18. Past project discussion  
    Singleton Design Pattern  
    Question around mongo DB (Indexing, Query Optimization)  
    Write a simple spring boot for POST method (Controller, Service, Repository)  
    Write a method using @Transactional to handle credit/debit on accounts.
19. Candidate - Sahil Ayush  
    Round 1 :  
    Interviewer - Ketan Verma
20. Q1. Find the missing number in a sorted array containing numbers from 1 to N. (Binary Search)  
    Q2. Architecture discussion on Single Sign On, Authentication flows (Resume based)  
    Q3. Generic tech stack discussion to understand what all things I have worked on till now.  
    Q4. Modification of Trapping rain water question - <https://leetcode.com/problems/trapping-rain-water/description/>
21. Round 2 :   
    Interviewer - Seena & Sushant
22. Q1. Basic Intro.  
    Q2. Write a Post-Mapping in a controller class for saving an employee name and address.   
    Q3. @Autowired what is it and use?   
    Q4. Make the above rest call as a Async call.  
    Q5. Which architecture should we use in order to handle async calls better in your opinion?   
    Q6. Have you worked on Kafka? (I haven't, I worked on pub-sub events of GCP so a bit discussion about that)  
    Q7. I had passed an Employee Entity/POJO object in the request body so they asked how would this object be mapped to an object in FrontEnd?  
    Q8. Write Singleton class?  
    Q9. What is the use of volatile keyword and where does java store these variables marked as volatile, in which memory?   
    Q10. What all kinds of DB's you have used till now?   
    Q11. How do you secure a rest API call?   
    Q12. How do you handle transaction related processes in Spring?   
    Q13. Can @transactional annotation be used at class level as well as method level at the same time? If yes then how would this work?  
    Q14. What are the Propagation levels in @Transactional and what is the behavior in default case.
23. Candidate → Uday Singh1
24. **Round 1** →  (Coderpad)
25. Interviewer (Anuj Arora)  
    -------------------------------------------------------  
    Design a data structure with following features:  
    1. push an element at the end of the data structure (O(1) time complexity)  
    2. Pop the element from the begining of the data structure( pop the earliest element added, O(1) TC)  
    3. Seek an element at index i at any given state of the data structure. (O(1) TC)  
    1,2,3,4,5  
    pop() -> 2,3,4,5  
    pop() -> 3,4,5  
    Seek(0) -> 3 seek(1)-> 4
26. **-----------------------------------------------------------------------------------------------------------------------------------------------------------------------**
27. **Round 2** ->
28. Interviewer (Eti Mishra & Zohaib Arbab)  
    --------------------------------------------------------  
    1. Can you write a code for a singleton class?  
    2. Can you write a code for an immutable class?  
    3. Can you write the code to find the area of a square using a functional interface and a lambda expression?  
    4. Can you explain the architecture of Kafka?  
    5. What is a partition in Kafka?  
    6. What is the use of a consumer group in Kafka?  
    7. What is the difference between SQL and NoSQL databases?  
    8. If you want to retrieve the records of employees with a salary greater than 25000, where the salary is not a primary key, how would you do it in a NoSQL database?  
    9. What is compound indexing in NoSQL?  
    10. What is multithreading?  
    11. What is the use of a circuit breaker and how would you implement it?  
    12. Can you explain event-driven architecture?  
    13. How does the monitor point work in a singleton class?  
    14. Write a code to find the next largest element for each element in an array and calculate the distance between them. For example, given the input `{ 71, 72, 76, 69, 70, 50, 78, 80 }`, the output should be `{1, 1, 4, 1, 2, 1, 1, 0}`. Here, the next largest element for 71 is 72, so the distance is 1. For 72, the next largest element is 76, so the distance is 1, and so on.

Candidate → Taj Mohammad

1. **Round 1** → (Coder pad)

Interviewer (Animesh. Ananmay)  
-------------------------------------------------------  
Design a data structure with following features:  
1. Given an input string. Find the longest uniform subsequence's first index and length of the subsequence. Example:

* input: "aabbbc"
* output: index: 2, length: 3 ("bbb")

2.  Given two arrays, merge them to single sorted array. For Example:

* input: arr1[] = {1,3,5,7}, arr2[] = {0,2,4,6}
* output: result [] = {0, 1, 2, 3, 4, 5, 6, 7}

Candidate: Sneha Shivanna

Round 1: Taken By Kuldip

Ques 1: Best Time to Buy and Sell Stock - LeetCode (You need to find index also)

Ques 2: Given a string str, the task is to convert the given string into the number without using any inbuilt function

You are given a string str that may contain a mixture of characters including digits, spaces, alphabetic characters, and signs ('+' or '-'). Your task is to convert the string into an integer following these specific rules:

1. **Leading Alphabets**: If the string starts with alphabetic characters, the function should return -100.
2. **Leading Spaces**: Any leading spaces in the string should be ignored.
3. **Sign Handling**: The string may contain a single '+' or '-' sign that indicates the sign of the resulting number. This sign should be considered only if it appears before any digits.
4. **Digit Conversion**: Convert consecutive digits into the corresponding integer value.
5. **Out of Range Handling**: If the resulting integer is outside the range of a 32-bit signed integer:
   * Return Integer.MAX\_VALUE (2,147,483,647) if the number is too large.
   * Return Integer.MIN\_VALUE (-2,147,483,648) if the number is too small.
6. **Alphabets After Digits**: If any alphabetic characters appear after the digits, stop processing further characters and return the integer formed up to that point.

**Additional Constraints:**

* You are not allowed to use any inbuilt functions for parsing or converting the string into an integer.
* You must handle edge cases explicitly as mentioned above.

**Example Cases:**

* 1. Input: " -42"
     + Output: -42
  2. Input: "4193 with words"
     + Output: 4193
  3. Input: "words and 987"
     + Output: -100
  4. Input: "-91283472332"
     + Output: Integer.MIN\_VALUE
  5. Input: "3.14159"
     + Output: 3
  6. Input: " +0 123"
     + Output: 0

Ques 3: Internal implementation of HashMap and how tree is balanced(explain about red/black tree height balance).

Round 2: Zohaib Arbab & Chandan Mandal

1. Can you write a code for a singleton class?
2. Explain about class level lock and object level lock. What happens when there is class level lock is acquired by a thread.
3. Can you write the code to find the area of a square using a functional interface and a lambda expression?
4. Explain architecture of Kafka
5. Given an int array return the random array . for Ex int[] arr = {1,2,3,4} output = {2,3,4,1} - for every execution the array elements should be randomly arranged.
6. Swap 2 numbers in a single statement.
7. Internal implementation of HashMap.
8. Candidate: Shashank Tiwari
9. **Round 1: Taken By Anuj Arora (15 July, 2024)**
10. Ques 1: Design a data structure with following features:  
      
    1. Push an element at the end of the data structure (O(1) time complexity) - void push(int element)  
    2. Pop the element from the beginning of the data structure( pop the earliest element added, O(1) time complexity) - int pop()  
    3. Seek an element at index i at any given state of the data structure. (O(1) time complexity) - int seek(int index)
11. Elements can be duplicate too.
12. Solution : Use a HashMap for this. Key will be index and Value will be the element (which is Integer in this case) .  
    Have a startIndex and endIndex variable to know which index to add/remove.  
    Here is the solution :
13. public class Main {
14. public static void main(String[] args) {
15. MyCustomizedDataStructure ds = new MyCustomizedDataStructure();
16. ds.push(1);
17. ds.push(2);
18. ds.push(3);
19. System.out.println(ds);
20. ds.pop();
21. System.out.println(ds);
22. ds.pop();
23. ds.pop();
24. System.out.println(ds);
25. ds.push(4);
26. System.out.println(ds.seek(0));
28. }
29. }
30. class MyCustomizedDataStructure {
31. private final Map<Integer, Integer> map = new HashMap<>();
32. private int headIndex = -1;
33. private int endIndex = -1;
34. public void push(int num) {
35. // When first element is added
36. if (map.isEmpty()) {
37. headIndex = 0;
38. endIndex = 0;
39. map.put(endIndex, num);
40. return;
41. }
42. map.put(++endIndex, num);
43. }
44. public int pop() {
45. if (map.isEmpty()) {
46. throw new RuntimeException("Nothing to remove as no elements present");
47. }
48. int val = map.remove(headIndex++);
49. // Reset the indices if map is empty
50. if (map.isEmpty()) {
51. endIndex = -1;
52. headIndex = -1;
53. }
54. return val;
55. }
56. public int seek(int index) {
57. if (map.isEmpty()) {
58. throw new RuntimeException("Map is empty and there is nothing to seek");
59. }
60. int actualIndex = headIndex + index;
61. if (actualIndex > endIndex) {
62. throw new RuntimeException("Index sought is out of bound for our DataStructure");
63. }
64. return map.get(actualIndex);
65. }
66. @Override
67. public String toString() {
68. return map.toString();
69. }
70. }
71. **Round 2: Taken By Sunil Kadadevarmath AND  Seena Edavalakandiyil (16 July, 2024)**
72. Questions By Seena:
73. a) Describe architecture diagram of last project you worked on. Also, list the REST endpoints for it and explain.
74. b) Explain Singleton Pattern
75. c) Explain strategy pattern
76. Questions By Sunil:
77. a) How you measured performance of the APIs of the project you worked on.
78. b) How you authenticated your Rest APIs
79. Candidate: Girish Kolisetti
80. **Round 1: Taken By Kuldip Kumar (18 July, 2024)**
81. Qsn1: Find all possible paths of a binary tree from root node to leaf node  
      
          5
82. 10     6
83. 7   8  
      
    OP: [[5,10], [5,6,7],[ 5,6,8]]
84. Qsn2: Convert String into int without using any inbuilt methods or lib  
      
    Conditions:  
    ip: "123"
85. op: 123
86. ip: "-123"
87. op: -123
88. ip: "12sbdjsh123"
89. op: 12  
    ip: "   127shdg  "
90. op: 127
91. ip:"  - 12shstd "
92. op: Exception
93. ip:"ajsdjs"  
    op: Exception
94. Qsn3: Hashmap internal working and time complexities for get and put operations
95. Candidate: Srikakula Sai Srinivas
96. **Round 1: Taken By Ketan Verma (17 July, 2024)**
97. 1.Find the missing number in a sorted array containing numbers from 1 to N. (Binary Search)  
    2.Discussion on previous project's architecture and migration challenges etc.
98. **Round 2: Taken By Seena and Sunil (18 July, 2024)**
99. Intro  
    Explain different layers in a Spring Boot application (like controller, service etc.)  
    Kafka architecture and real time use cases  
    How does Kafka ensure message is message is stored/delivered (basically what changes we need to ensure from producer side and consumer side)  
    what is dead letter topic  
    Write Singleton design pattern.  
    Explain factory design pattern.  
    Write a code for Get Mapping for product details for particular product.  
    Spring vs Spring Boot  
    What are changes needs to be done to change the default tomcat server in spring boot  
    what all different Generation Strategies in hibernate / spring data jpa  
    Question on transactional annotation like class level and method level  
    What is difference between primary key vs unique key  
    Horizontal scaling vs vertical scaling  
    What is prepared statement.  
    Java 17 features  
    Question on previous projects
100. Candidate: Manchirevula Prasanna Kumari
101. **Round 1: Taken By Kuldip Kumar (23 July, 2024)**
102. List Java 8 features?
103. How Streams work internally?
104. List terminal and intermediate functions in stream?
105. Implement code to show usage and difference between map and flatMap?
106. LLD of stackOverFlow website(Create the classes and methods) and flow up question on how to make sure that user can access only valid data?
107. **Round 2: Taken By Anuj Arora and Zohaib Arbab (24 July, 2024)**
108. Design three classes Person, PhoneNumber class as immutable and Address Class as mutable and explain the each keyword and why to use it?(While Implementing use java 8 features)  
     Design Singleton class (Eager, Lazy and thread safe)?  
     what is HashMap?
109. Space and TimeComplexity of hashmap functions?  
     What is Bucket in HashMap?  
     Difference between Fail fast and Fail safe Iterator?

Candidate: Vinay Maurya

**Round 1: Taken By Ketan Verma (23 July, 2024)**

Discussion over software architecture and practices pertaining to the following problem:  
Communication between two services

* Sync vs Async communication paradigm
* Using kafka
* Multithreading
* Fixing issues that pop up in production after deploying some change

**Round 2: Taken By Seena and Sunil (25 July, 2024)**

* Discussion over past project
* Singleton Design Pattern
* Factory Design Pattern
* Java 8 features
* Lazy loading in ORM
* Kafka - architecture, dead letter topic
* Advantage of Java 8 lambdas in cloud environment
* Class vs Interface
* Candidate: Rithish Chandra Bathula
* **Round 1: Taken By Ketan Verma (18 July, 2024)**
* Discussion on previous project's architecture, tools I have worked on and migration challenges etc.
* **Round 2:  Taken By Eti Mishra and Sunil (24 July, 2024)**
* Intro
* Shallow copy and deep copy explanation with a sample code
* Internal implementation of HashSet
* What is Multithreading? What is ExecutorService and its internal implementation
* How to make REST application secure
* Basic Authentication and JWT authentication
* How you measured performance of the APIs of the project you worked on.
* Explain about Load Testing and tool used for load testing
* Splunk monitoring and write a sample sample splunk query (Resume based question)
* What is Synchronization
* Explain about Time Complexity
* Explain about Kafka working experience
* Explain about Agile methodology and how do you prioritize between sprint works and a production bug?
* What is the process you follow to resolve the production issue? Like how you find the exact issue.
* Coderpad question to implement a thread-safe Queue. Consider a scenario that we have 2 threads i.e., producerThread and consumerThread. Here, ProducerThread can add the element into the Queue if queue is available and queue is not full. ConsumerThread will read the values from the queue only if queue is not empty. Write logic to add element and fetch element from Queue with a thread-safety
* Difference between Queue and PriorityQueue
* Scenario based question on PriorityQueue like if I add elements like 3, 2, 5 into my PriorityQueue and if I call queue.poll() method, which element will be removed.
* Write a code to find the next largest element for each element in an array and calculate the distance between them. For example, given the input `{ 71, 72, 76, 69, 70, 50, 78, 80 }`, the output should be `{1, 1, 4, 1, 2, 1, 1, 0}`. Here, the next largest element for 71 is 72, so the distance is 1. For 72, the next largest element is 76, so the distance is 1, and so on.
* ***Candidate***: Pavan Kumar Marri  
  ***Date***: 13 Aug 2024
* **Coderpad Round:**
* ***Problem 1:***
* Given a list of student test scores, find the best average grade. Each student may have more than one test score in the list. Complete the bestAverageGrade function in the editor below. It has one parameter, scores, which is an array of student test scores.  
  Each element in the array is a two-element array of the form [student name, test score]  
  e.g. [ "Bobby", "87" ].  
  Test scores may be positive or negative integers.  
  If you end up with an average grade that is not an integer, you should use a floor function to return the largest integer less than or equal to the average. Return 0 for an empty input.  
  Example:  
  Input:  
  [ [ "Bobby", "87" ],  
  [ "Charles", "100" ],  
  [ "Eric", "64" ],  
  [ "Charles", "22" ] ].  
  Expected output: 87  
  Explanation: The average scores are 87, 61, and 64 for Bobby, Charles, and Eric,  
  respectively. 87 is the highest.
* ***Problem 2:***
* You are an avid rock collector who lives in southern California. Some rare and desirable rocks just became available in New York, so you are planning a cross-country road trip. There are several other rare rocks that you could  
  pick up along the way. You have been given a grid filled with numbers, representing the number of rare rocks available in various cities across the country. Your objective is to find the optimal path from So\_Cal to New\_York that would allow you to accumulate the most rocks along the way.
* Note:  
  1) You can only travel either north (up) or east (right).  
  2) Consider adding some additional tests in doTestsPass().  
  3) Implement optimalPath() correctly.  
  4) Here is an example:  
  ^  
  {{0, 0, 0, 0, 5}, New\_York (finish) N  
  {0, 1, 1, 1, 0}, < W E >
* So\_Cal (start) {2, 0, 0, 0, 0}} S  
  v
* The total for this example would be 10 (2 + 0 + 1 + 1 + 1 + 0 + 5).  
  Signature:  
  public static Integer optimalPath(Integer[][] grid) {  
  }
* ***Problem 3:***Convert String to Integer without using any built in functions.

LEAD DEVELOPER QUESTIONS:

* **There were questions regarding candidate experience and related to it theoretical questions from time to time.**
* **Questions:**
* **1) Decode the given input String and expected output.  
  input = a3[b2[cd]];  
  output=abcdcdbcdcdbcdcd**
* **input =ab2[c2[b]]e3[a]z  
  output=abcbbcbbeaaaz**
* **2) Find the Pair of elements in array equal to the given number.  
  Example:  
  arr={1,2,3,4,5}, number=6**
* **output:  
  {2,4}  
  {1,5}**
* **(Added on 01/30/2024)**
* **3) Find all the distinct triplets in an array with sum equals to 0.  
  Example:  
  arr = {0, -1, 2, -3, 1}, Output =  (0 -1 1), (2 -3 1)**
* **4)Given n pairs of parentheses, write a function to generate all combinations of well-formed parentheses  
  Example:  
  n=1, Output = ["()"]**
* **n=2, Output = ["()()". "(())"]**
* **n=3, Output = ["((()))", "(()())", "(())()", "()(())", "()()()"]**

**Round 1:  two coding questions in java**

1. find list of duplicate numbers in an array in most optimized way  
   example - Input = [2,5,3,1,7,4,5,1,8], output = [5,1]
2. decode string (input string has number in square brackets and we have to decode it by transforming number into characters before square bracket)  
   example -  Input = ab[2]c[3]de, output = abbcccde

**Round 2: One coding question in java which was using two pojo classes**

1. There are 2 pojo classes Input.java and status.java
2. Input has 3 fields - int Id (primary key), String Name, String Divison
3. Status has 4 fields - int Id (primary key), String Name, String Divison, boolean isActive
4. There is an inputList of Input data type having 10 records (it could be n records)
5. There is an statusList of Status data type having 5 records (it could be n records)

UseCase -

1. If Input record is already there in status table and isActive is true - do nothing (we will compare both table's Id)
2. If Input record is already there in status table and isActive is false - update status record as isActive = true;
3. If Input record is not there in status table - add input record in status with isActive = true
4. If input record is there in status table but divison has been changed - update status record divison and set isActive = true
5. If status has an record which is not there in inputList - update status record as isActive = false;

**Round 1:**

  Problem 1: Design a system like Splitwise(Expense sharing app).

* Expense can be split equally or percentage wise.
* Final Settlement should show How much amount owes by which member?

**Round 2:**

Problem 1: Write a program to generate Random Number without using sdk methods or any library.

**Round 1:**  
    problem 1: encode aaabbc to a3b2c1  
    problem 2: decode a3b2c1 to aaabbc  
    problem 3: decode [ab]3[cd]2 to abababcdcd  
handle null and empty string scenarios  
  
**Round 2:**  
   problem: Two list of employees i.e. presentList and masterList  
     Employee has fields id,divison and status  
     PresentEmployeeList has Employee with status true.  
     MasterEmployeeList have older employees.  
     Update MasterList with following conditions:  
      1) if presentList employee is found in masterList then update divison and status according to present employee  
      2) if presentList employee is not found in masterList then create new employee in masterList with properties of present employee  
      3) if masterList employee is not found in presentList then update status to false

**Round 1:**

  Problem 1: given an integer array that satisfies the following conditions

* Each array may or may not contain a pivot element.
* An element is a pivot element if all the numbers after it are sorted in descending order and if all the numbers before it are sorted in ascending order.
* An element is a pivot element if all the numbers after it are sorted in ascending order and if all the numbers before it are sorted in descending order.

       The problem is to find the pivot element or else return -1 if array does not contain a pivot element.

       array = [1,2,3,4,5] ans= -1,  array =[5,4,3,2,1] ans = -1,  array = [5,4,1,2,3] ans=1,  array = [1,2,5,4,3] ans=5

     Problem 2: find mean mode and median of an integer array (take care of overflow conditions in case of mean).

**Round 2:**

   Problem 1: Design a data structure that can perform get(), delete(), insert() and getRandom() in O(1) time.

**Round 1:**

**Problem 1:**

There are n children standing in a line. Each child is assigned a rating value given in the integer array ratings.

You are giving candies to these children subjected to the following requirements:

* Each child must have at least one candy.
* Children with a higher rating get more candies than their neighbors.

Return *the minimum number of candies you need to have to distribute the candies to the children*.

**Example 1:**

**Input:** ratings = [1,0,2]

**Output:** 5

**Explanation:** You can allocate to the first, second and third child with 2, 1, 2 candies respectively.

**Example 2:**

**Input:** ratings = [1,2,2]

**Output:** 4

**Explanation:** You can allocate to the first, second and third child with 1, 2, 1 candies respectively.

The third child gets 1 candy because it satisfies the above two conditions.

**Round 1(interviewer - Awinash singh):**

**Problem 1:**

Write a program or function that takes this list as input and returns the name and value of the pair with the maximum average numerical value.  
handle negative number, empty list.

input -  {{"Mohit",82},{"Mohit",78},{"Awinash",75},{"Awinash",79}}  
output -   ["Mohit", 80]  
  
  **Problem 2:**

encode aabbccccddd => abc4d3, limit =2  
Write a program or function that takes a string and a limit as input, and returns the encoded string based on the specified rules.

**Problem 3:**

Write a program or function that takes two integers as input, performs the division, and returns the result as a string. If the result has repeating decimals, enclose the repeating part within square brackets in the string representation.

input -  1 , 3  
1/3 = 0.33333 => 0.[3]

**Round 2(interviewer - Sruthi Santhosh):**

**Problem 1:**

The input consists of an unsorted list of pairs. Merge all pairs that intersect with each other. Please note that these pairs can be visualized as lines on a scale.

Example input: {(1,4),(2,5),(7,9),(6,10)}

Desired output: {(1,5),(6,10)}

**Problem 2:**

Array of integer numbers (including positive, negative, and zero): Find the maximum sum of a contiguous subarray.

Input: [5,-1,-8,6,8,1] Desired Output: 15

1) How to create Infinite loop in java

2) Given an Integer n, I have to find the nth Fibonacci number

    Input: n= 3 Output: 1

3) Find the smallest number in rotated sorted array

    Input: arr=[3,4,5,1,2] Output: 1

   Follow-up question: If there are duplicates in rotated sorted array, find the smallest number

4) Find if the number n is the power of 10 or not

5) Find the maximum grade

   Input:{ "Bobby","87"}, {"Charles", "100"}, {"Jessy","45"}, {"Charles", "22"}

   Output : 87

6. Find shortest path and calculation distance between two coordinates. (DSA - Medium)

      a. Define the formula and steps to calculate shortest path between two coordinates [(x1, y1), (x2, y2)].

      b. Calculate the distance between two points. e.g. we have below points and how we find distance between A to H.

          //A, B, C, D, E

          //F, G, H, I, J,

          //K, L, M, N, O

      c. completes the below function and return distance.

            public int findShortestPath(int x1, int y1, int x2, int y2 ){ //return distance}

2. Previous project related questions like workflow and technology stack used.

     a. Your contribution in project like in which area you work.

     b. Roles and responsibilities in last project.

1. Find the minimum distance between the words of a string.

Input:

          S="the quick the brown quick brown the frog"

          w1="quick"

          w2="frog"

Output: 2

1. About project and modules worked on  
2. Detailed functionalities of modules worked on in previous project  
3. Find substring of 2 given strings, return boolean  
4. Find the longest substring of 3 given input strings, return maxLength of substring  
5. Given 2 integer arrays with same length, find dotProduct of arrays  
6. In a given integer array, If length is less than 2 just return -1

1. Find the second smallest number
2. If array contains positive and negative numbers, find the absolute second smallest number
3. modify the logic to find the third smallest number

7. Lets given m,n array,Given two points  
source = M (m-1,0)  
Destination = R (m-1,n-1)  
Find the possible number of paths from source to destination  
constraints:  
posible ways = (i,j+1),(i-1,j+1),(i+1,j+1)

Ex given

A B C D E F   
G H I J K L   
M N O P Q R

1. Introduction
2. Last project which you worked on.
3. Find the min and max given an Array.
4. Find Peaks and Troughs present in an Array. Peak is defined as arr[i-1] < arr[i] && arr[i+1] < arr[i]. Similarly Troughs are defined as arr[i-1] > arr[i] && arr[i+1] > arr[i]
5. Find Drawdowns in a given Array which represents the prices of a stock. Drawdowns are defined as the Difference in stock prices between a Peak and the next immediate Trough (if it exists) utilize the previous method to compute peak and troughs.
6. 1) Given a String "aabbbbccd".  Return the length of the longest substring along with the starting index.  
   2) Given an array of non-negative integers representing the elevations from the vertical cross-section of a range of hills, determine how many units of snow could be captured between the hills.
7. **Example:**
8. See the example array and elevation map below.
9. [A graph of numbers and a line

   Description automatically generated with medium confidence](https://www.csestack.org/wp-content/uploads/2022/01/snow-between-hills.png)
10. Output: In this example, 13 units of snow (\*) could be captured.
11. **There were questions regarding candidate experience and related to it theoretical questions from time to time.**
12. **Questions:**
13. **1) Decode the given input String and expected output.  
    input = a3[b2[cd]];  
    output=abcdcdbcdcdbcdcd**
14. **input =ab2[c2[b]]e3[a]z  
    output=abcbbcbbeaaaz**
15. **2) Find the Pair of elements in array equal to the given number.  
    Example:  
    arr={1,2,3,4,5}, number=6**
16. **output:  
    {2,4}  
    {1,5}**
17. **(Added on 01/30/2024)**
18. **3) Find all the distinct triplets in an array with sum equals to 0.  
    Example:  
    arr = {0, -1, 2, -3, 1}, Output =  (0 -1 1), (2 -3 1)**
19. **4)Given n pairs of parentheses, write a function to generate all combinations of well-formed parentheses  
    Example:  
    n=1, Output = ["()"]**
20. **n=2, Output = ["()()". "(())"]**
21. **n=3, Output = ["((()))", "(()())", "(())()", "()(())", "()()()"]**

**Round 1:  two coding questions in java**

1. find list of duplicate numbers in an array in most optimized way  
   example - Input = [2,5,3,1,7,4,5,1,8], output = [5,1]
2. decode string (input string has number in square brackets and we have to decode it by transforming number into characters before square bracket)  
   example -  Input = ab[2]c[3]de, output = abbcccde

**Round 2: One coding question in java which was using two pojo classes**

1. There are 2 pojo classes Input.java and status.java
2. Input has 3 fields - int Id (primary key), String Name, String Divison
3. Status has 4 fields - int Id (primary key), String Name, String Divison, boolean isActive
4. There is an inputList of Input data type having 10 records (it could be n records)
5. There is an statusList of Status data type having 5 records (it could be n records)

UseCase -

1. If Input record is already there in status table and isActive is true - do nothing (we will compare both table's Id)
2. If Input record is already there in status table and isActive is false - update status record as isActive = true;
3. If Input record is not there in status table - add input record in status with isActive = true
4. If input record is there in status table but divison has been changed - update status record divison and set isActive = true
5. If status has an record which is not there in inputList - update status record as isActive = false;

**Round 1:**

  Problem 1: Design a system like Splitwise(Expense sharing app).

* Expense can be split equally or percentage wise.
* Final Settlement should show How much amount owes by which member?

**Round 2:**

Problem 1: Write a program to generate Random Number without using sdk methods or any library.

**Round 1:**  
    problem 1: encode aaabbc to a3b2c1  
    problem 2: decode a3b2c1 to aaabbc  
    problem 3: decode [ab]3[cd]2 to abababcdcd  
handle null and empty string scenarios  
  
**Round 2:**  
   problem: Two list of employees i.e. presentList and masterList  
     Employee has fields id,divison and status  
     PresentEmployeeList has Employee with status true.  
     MasterEmployeeList have older employees.  
     Update MasterList with following conditions:  
      1) if presentList employee is found in masterList then update divison and status according to present employee  
      2) if presentList employee is not found in masterList then create new employee in masterList with properties of present employee  
      3) if masterList employee is not found in presentList then update status to false

**Round 1:**

  Problem 1: given an integer array that satisfies the following conditions

* Each array may or may not contain a pivot element.
* An element is a pivot element if all the numbers after it are sorted in descending order and if all the numbers before it are sorted in ascending order.
* An element is a pivot element if all the numbers after it are sorted in ascending order and if all the numbers before it are sorted in descending order.

       The problem is to find the pivot element or else return -1 if array does not contain a pivot element.

       array = [1,2,3,4,5] ans= -1,  array =[5,4,3,2,1] ans = -1,  array = [5,4,1,2,3] ans=1,  array = [1,2,5,4,3] ans=5

     Problem 2: find mean mode and median of an integer array (take care of overflow conditions in case of mean).

**Round 2:**

   Problem 1: Design a data structure that can perform get(), delete(), insert() and getRandom() in O(1) time.

**Round 1:**

**Problem 1:**

There are n children standing in a line. Each child is assigned a rating value given in the integer array ratings.

You are giving candies to these children subjected to the following requirements:

* Each child must have at least one candy.
* Children with a higher rating get more candies than their neighbors.

Return *the minimum number of candies you need to have to distribute the candies to the children*.

**Example 1:**

**Input:** ratings = [1,0,2]

**Output:** 5

**Explanation:** You can allocate to the first, second and third child with 2, 1, 2 candies respectively.

**Example 2:**

**Input:** ratings = [1,2,2]

**Output:** 4

**Explanation:** You can allocate to the first, second and third child with 1, 2, 1 candies respectively.

The third child gets 1 candy because it satisfies the above two conditions.

**Normal Java interview questions**

1. Tell me about your past company experience

2. Tell me about your Last projects with tech stack, role and responsibilities.

2. Tell me about your Last projects with tech stack, role and responsibilities.  
3. Previous Project Architecture with flow diagram?  
    1.1 Project framework with each block explanation? why it is used?  
    1.2 Application security mechanism?  
    1.3. Data flow/persist flow.  
    1.4 Tech stack purpose and its functionality?  
4. What you independently develop in this project/Individual contribution?

5. New Java 8 features?  
6. Types of Design patterns? Explain factory design pattern with your project example.  
7. Junit testing tool annotations?  
8. How to review code, which things to follow during the code review?  
9. Challenges you faced during development or in production or in task and how you resolved it.

Java Programs:  
1. write a code to start two thread T1 and T2, T1 thread will print only odd numbers, T2 will print only event   
numbers. Print the numbers is in sequence as 1,2,3,4...till 50?

2. write a code to sort string "abc" as per the pattern "bac"(according to pattern "bac" you have to sort string "abc")?   
define its time complexity?

**Interview experience**

1.First half and hour she took for project related questions only based on our profile and technology mentioned on it.  
we need to be very strong to explain about working and past project details.  
2.Explain how do you create your own hashmap?  
3.Find the count of character stored in a file?  
    
public class FildReaderWithCount {  
    public static void main(String[] args) throws IOException {  
        File file = new File("C:/Users/Raja\_G/Documents/file.txt");  
        FileInputStream input = new FileInputStream(file);  
        int i = 0;  
        Map<String, Integer> map = new HashMap<>();  
        while ((i = input.read()) != -1) {  
            char s = (char) i;  
            String s1 = s + "";  
            if (map.containsKey(s1)) {  
                map.put(s1, map.get(s1) + 1);  
            } else {  
                map.put(s1, 1);  
            }  
        }  
        map.entrySet().stream().forEach(a -> System.out.println(a.getKey() + "  " + a.getValue()));  
        input.close();  
    }  
}

4.explain about spring and spring boot framework  
5.Java 8 features  
6.time complexity of hashmap and others  
7.working of binarry tree  
8.exception handling in java.  
9.Explain the architecture and flow of the application

1. \*\*\*coding and situation  
     
   Given 2 ascending sorted arrays, merge the arrays into a single array.   
   Follow Up :
   1. Modify the code to sort the array in descending order.
   2. Time complexity and space complexity
2. Given an array and sum k, find all the pairs where sum is equal to k.

Follow Up:

1. Provide brute force and optimized approach both
2. Based on the approach asked about the time and space complexity.
3. And Indepth HashMap internals

Implement a method that checks if a given integer is a power of 10. Add test cases.   
2. Find a bug in a method atoi (a method that converts String to int). Need also to implement a method that tests atoi() method and come up with possible test cases, including invalid input cases.   
For the both tasks the interviewer asks time complexity and test cases that may break the code.   
   
Also before the coding I was asked 3-4 behavioral questions. For example: imagine a situation when your team member that works from abroad (huge time difference) delivers something that is completely wrong. You need to complete the task by tomorrow but it is already 12 a.m. at your colleague's location. What would you do?

1. Find the most frequent IP qaddress from the logs (string array)

["10.0.0.1 - GET 2020-08-24", "10.0.0.1 - GET 2020-08-24", "10.0.0.2 - GET 2020-08-20"]

2. Having student name and test score (array), count best average grade. Student may have more than one test score.

1. Find first non-repeating character in a String
2. Having student name and test score (array), count best average grade. Student may have more than one test score
3. Given an array and startindex, each element of the array is pointing to another index in the array and so on   
   Find if a loop exists and if yes return the length else –1

1. Write a program or function that takes two integers as input, performs the division, and returns the result as a string. If the result has repeating decimals, enclose the repeating part within brackets in the string representation.

input -  1 , 3   
1/3 = 0.33333 => 0.(3)

1 / 2= 0.5

1/0=-1

behavioural questions:    
What will you do if your work credit is claimed by someone else / stolen by anyone without giving any credit to you.

1.Best average grade in string 2d array.

Input:

[["Bobby","87"],

["Charles","100"],

["Eric","64"],

["Charles","22"]].

Expectedoutput:87

Explanation:Theaveragescoresare87,61,and64forBobby,Charles,andEric,

respectively.87isthehighest.

2.countLengthOfCycle int array(hare and tortoise algorithm)

 int arry[] ={0,1,0} startindex 0 output is 1

1. Implement put, contains and inorderTraversal of BST.
2. Reverse string
3. You are an avid rock collector who lives in southern california. Some rare and desirable rocks just became available in new york, so you are planning a cross-country road trip. There are several other rare rocks that you could pick up along the way. You have been given a grid filled with numbers, representing the number of rare rocks available in various cities across the country. Your objective is to find the optimal path from so\_cal to new\_york that would allow you to accumulate the most rocks along the way. Note: you can only go up or right.                  example: {{0,0,0,0,5},

                  {0,1,1,1,0},

                  {2,0,0,0,0}}; ans = 2+0+1+1+1+0+5

1. Introduction and what all API you have worked on in your project?

2.Deque implementation.

Methods to implement: addFirst(),addLast(),removeFirst(),removeLast(),peekFirst(),peekLast(),size()

3.Rock Collector Question.

1. Find first non-repeating character in String
2. Trapping rainwater problem <https://leetcode.com/problems/trapping-rain-water/>

Java memory areas, garbage collection

1. About past project: how to create API and all.
2. Spring Boot application: how does it work, annotations and all
3. A robot is standing at a position (0,0) and a string input is given. Based on the string input, a robot can either go up, down, left or right when the string contains 'U', 'D','L', 'R' characters respectively. All other characters can be ignored. For the given string, we need to find out the final position of the robot.
4. Most frequent IP address from the logs.
5. What is the impact of testing in software development ? How do we achieve it ?
6. Deployment plan and strategy in software development.
7. If you notice something is hard-coded in your codebase, what will you do about it?
8. Behavioral questions:
   1. If you are stuck with one problem, what will you do to get it done ?
   2. If you bany issue while writing code, will you google it ?
9. Intro – Asked basic questions about my projects in the past and my tech stack.
10. DSA 1: First non-repeating character with approach, code and complexities.
11. DSA 2: Given a string **str** and an integer **K,** the task is to reduce the string by applying the following operation any number of times until it is no longer possible.    
    Eg - K = 2, str = “geeksforgeeks” ,  Output - gksforgks   
    K = 3, str = “geeksforgeeks” ,  Output - geeksforgeeks   
       
    Approach and time complexities only. (No code)
12. Spring vs Springboot
13. Framework to run and compile springboot applications that I used-  I told gradle and maven. Seems like he was expecting that only.
14. Few questions about maven dependencies that I am familiar with.    
    I mentioned few names including springboot-actuator. So, he asked the use of actuator dependency.
15. Behavioral question -   
    What will you do if you are stuck with some technical issues while development ?   
    I told the steps of resolution -> End to end debugging of issues -> check In-house User guides, Developer guides (If functional issue due to lack of product knowledge) -> Ask lead to arrange for some KT  -> If some tech(Java, DSA) related  issue, look on stackoverflow. If not found, ask from teammates.

**DSA**

1. Dot multiplication of 2 arrays. If both are of same length, dot product cannot be taken. Return –1.   
   Array 1 - [a1, a2, …., aN] Array 2 - [b1, b2, …, bN]   
   Dot product - (a1 \* b1) + (a2 \* b2) + … + (aN \* bN)
2. (Interviewer mistakenly pasted solution, so changed the question) Given a string, there are some repeating characters and some are unique. Return first occurring unique character.   
   Input – abacbef   
   Output – Unique characters are c, e and f. 'c' is occurring first so answer is 'c'.
3. Implement a run length encoding function. For a string input the function returns output encoded as follows:   
   "a" -> "a1"   
   "aa" -> "a2"   
   "aabbb" -> "a2b3"   
   "aabbbaa" -> "a2b3a2"   
   "" -> ""

**Behavioral**

What will you do in a situation, when your lead is showing favoritism for an employee.

**DSA**

1. First non-repeating character with approach, code and complexities. test cases other than listed.
2. Fine the best average grade in string 2d array, scores might be positive and negative integers.

Input:

[["Bobby","87"],

["Charles","100"],

["Eric","64"],

["Charles","22"]].

Expected output: 87

Explanation: The average scores are 87, 61, and 64 for Bobby, Charles, and Eric, respectively. 87 is the highest.

with approach, code and complexities. test cases other than listed.

1. Find the most frequent IP address from the logs (string array)
2. You are given an integer array of size N.  Every element of the array is greater than or equal to 0. Starting from arr[startIndex], follow each element to the index it points to. Continue to do this until you find a cycle. Return the length of the cycle. If no cycle is found return -1
3. Past Projects.
4. Spring-Boot Bean annotation, hibernate, jpa.
5. Find the most frequent IP address from the logs (string array)
6. Fine the best average grade in string 2d array, scores might be positive and negative integers.

Input:

[["Bobby","87"],

["Charles","100"],

["Eric","64"],

["Charles","22"]].

Expected output: 87

Explanation: The average scores are 87, 61, and 64 for Bobby, Charles, and Eric, respectively. 87 is the highest.

with approach, code and complexities. test cases other than listed.

1. Past Project.

**DSA**

1. Find first non-repeating character in a string

(Repeated question from the page)

2. Find the length of the cycle in an integer array of size N.

Input:

* ([1, 0], 0), Output: 2
* ([1, 2, 0], 0) Output: 3

**Behavioral**

Suppose you have done something very wrong in the project. How will you deal and will you take accountability of the same.

There were some follow-up questions as well.

1. Program to check whether given integer number as input is palindrome (Avoid using String/ other internal APIs). Basically logic implementation was required.
2. Represent Human and Tiger using java oops principle.
3. Given an array of integers and an integer k. Print all the pairs whose sum is equal to k.
4. Given an algorithm for length of longest repeating subsequence. Explain the code lines.    
   [Longest Repeating Subsequence - GeeksforGeeks](https://www.geeksforgeeks.org/longest-repeating-subsequence/)
5. Brief intro and asked about previous projects
6. 1.Median of 2 sorted arrays
7. [Median of Two Sorted Arrays - LeetCode](https://leetcode.com/problems/median-of-two-sorted-arrays/)
8. 2.Longest Uniform Substring
9. Given an input string . Find the longest repeating substring first index and length of the subsequence. Ex : input : "aabbbc" ; output : index : 2, length : 3 ("bbb")
10. **DSA**
11. **First non repeating character in a String**
12. **Return the highest average grade from a 2d string array**
13. with approach, code and complexities. test cases other than listed.

**DSA**

1. Returns the smallest number in array that has been rotated

           For example - Array {3,4,5,6,1,2} returns 1

          Input array was originally sorted in increasing orders

          FindMinInArray must have O(log n) runtime

         Input array does not have any duplicates .

  2.   Given an array of non-negative integers representing the elevations

  from the vertical cross section of a range of hills, determine how

  many units of snow could be captured between the hills.

  See the example array and elevation map below.

                                \_\_\_

            \_\_\_                |   |        \_\_\_

           |   |        \_\_\_    |   |\_\_\_    |   |

 \*        \_\_\_|   |    \_\_\_|   |   |   |   |   |   |

    \_\_\_|\_\_\_|\_\_\_|\_\_\_|\_\_\_|\_\_\_|\_\_\_|\_\_\_|\_\_\_|\_\_\_|\_\_\_|\_\_\_

   { 0,  1,  3,  0,  1,  2,  0,  4,  2,  0,  3,  0 }

**LeetCode -  Decode String**

From <<https://leetcode.com/problems/decode-string/description/>>

FRONT END:

**ReactJS**

1. Find second smallest number in an array

2. How react working on the browser

3. How react update browser dom when a component updates

4. What is hooks ? How hooks working in react, how it's updating in browser?

5. How to render some components of external website to our react application? With using cors

6. Explain cors? 7. What is redux, what is the use of it? how redux working on the browser?

8. How javascript working inside the browser, how js update a element in browser?

9. Difference between normal dom update and react update the dom? How it's updating on the browser?

10.  Redux? Explain flow , why we use middlewares?

11. React Portals - its use?

12.Refs?

13. React Fibre?

* 1. Coding question
  2. Flatmap
  3. ParkingLot