1. LEARNING CODING:- FROM LEETCODE & INTERVIEWBIT ADITYA FOR DP (NOT EVERYTHING PLACEMENT TAK 2 BAAR IB KARNA DON'T GO FOR HARD QUESTION)
2. HR QUESTION:- https://docs.google.com/document/d/1mJOlCYdoZWxxbZ5nMyRbjoppS95CwBztf8M_msDNsr0/edit?usp=sharing
3. GFG PUZZELS:- https://www.geeksforgeeks.org/puzzles/
4. GFG MUST DO:- https://www.geeksforgeeks.org/must-do-coding-questions-for-companies-like-amazon-microsoft-adobe/
5. SDE SHEET OF (*TAKE U FORWARD YT CHANNEL)*:-

For banks and quant roles
50 problems in probability,
https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwiAruWS5_frAhVZfd4KHTTiDusQFjAAegQIARAB&url=http%3A%2F%2Fwww.mbapreponline.files.wordpress.com%2F2013%2F07%2Ffifty_challenging_problems_in_2.pdf&usg=AOvVaw1NkMVVY3DAHWNiFji9BEnT

6. Expectation value - https://www.codechef.com/wiki/tutorial-expectation

SDE SHEET

Day1: (Arrays)

1. **Sort an array of 0's 1's 2's without using extra space or sorting algo**
   https://www.youtube.com/watch?v=oaVa-9wmpns&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=2 (Problem link in description)
   Basic solution(counting sort)
   https://medium.com/enjoy-algorithm/sort-an-array-of-0s-1s-and-2s-b4e61533e750

```
public class sortcolor {
    public static void main(String[] args) {
       int arr[]= {0, 1, 2, 0, 1, 2};
        int nof0=0,nof1=0,nof2 =0;
        while(i<arr.length){
            if(arr[i]==0) nof0++;
            if(arr[i]==1) nof1++;
            if(arr[i]==2) nof2++;
            i++;
        }
       int j=0;
       while(nof0>0){
           arr[j]=0;
           j++;
           nof0--;
       }
       while(nof1>0){
           arr[j]=1;
           j++;
           nof1--;
       }
       while(nof2>0){
           arr[j] = 2;
           j++;
           nof2--;
       }
        int k=0;
        while(k<arr.length){
           System.out.print(" "+arr[k]);
            k++;
        }
    }
}
```

2. **Repeat and Missing Number**
   https://www.youtube.com/watch?v=5nMGY4VUoRY&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=3 (Problem link in description)

https://www.geeksforgeeks.org/find-a-repeating-and-a-missing-number/
Don't go for the XOR solution
Just do the easy one
Basic solution

```
import java.util.Arrays;
class ArrayNumber
{
public static void main(String[] args)
 {
  int [] arr = new int [] {3,1,3};
  // sorting input array
  Arrays.sort(arr);   //this might be something which the interviewer says to do it
  int n = arr.length;
  int miss = 0, repeat = 0;
  // finding missing and repeating elements
  for (int i = 0; i < n; ++i)
    {
        if(i == n-1)
             break;

        if(arr[i] == arr[i+1])
            repeat = arr[i];

        if((arr[i+1] - arr[i])!= 1)
            miss = i+1;
    }
  System.out.println("Missing: "+miss+" Repeating: "+repeat);
 }
}
```

and **for the optimal one we can use the hash map**
https://youtu.be/70qy6_gw1Hc **(basics of hash maps)**

```
   int[] arr = { 4, 3, 6, 2, 1, 1 };

        HashMap<Integer, Boolean> numberMap = new HashMap<>();

        int max = arr.length;

        for (Integer i : arr) {

            if (numberMap.get(i) == null) {
                numberMap.put(i, true);
            }
            else {
                System.out.println("Repeating = " + i);
            }
        }
        for (int i = 1; i <= max; i++) {
            if (numberMap.get(i) == null) {
```

```
                System.out.println("Missing = " + i);
            }
        }
```

And good thing is we can use mainly the library for sorting so we need not to write the merge sort algorithm over there
But in case the interviewer says to write it please learn know how to implement **merge sort**

3. **Merge two sorted Arrays without extra space**
   https://www.youtube.com/watch?v=hVl2b3bLzBw&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=4 (Problem link in description)

   Efficient solution is the GAP algorithm
   https://medium.com/swlh/merge-two-sorted-arrays-without-extra-space-efficiently-o-1-gap-method-detailed-simplified-57a336146601
   Very good explanation

```
public class mergearray {

    private static int nextGap(int gap)
    {
        if (gap <= 1)
            return 0;
        return (gap / 2) + (gap % 2);
    }

    private static void merge(int[] arr1,
                              int[] arr2, int n,
                              int m)
    {
        int i, j, gap = n + m;
        for (gap = nextGap(gap); gap > 0;
             gap = nextGap(gap))
        {
            // comparing elements in the first
            // array.
            for (i = 0; i + gap < n; i++)
                if (arr1[i] > arr1[i + gap])
                {
                    int temp = arr1[i];
                    arr1[i] = arr1[i + gap];
                    arr1[i + gap] = temp;
                }

            // comparing elements in both arrays.
```

```
            for (j = gap > n ? gap - n : 0;
                 i < n && j < m;
                 i++, j++)
                if (arr1[i] > arr2[j])
                {
                    int temp = arr1[i];
                    arr1[i] = arr2[j];
                    arr2[j] = temp;
                }

            if (j < m)
            {
                // comparing elements in the
                // second array.
                for (j = 0; j + gap < m; j++)
                    if (arr2[j] > arr2[j + gap])
                    {
                        int temp = arr2[j];
                        arr2[j] = arr2[j + gap];
                        arr2[j + gap] = temp;
                    }
            }
        }
    }

    // Driver Code
    public static void main(String[] args)
    {
        int[] a1 = { 10, 27, 38, 43, 82 };
        int[] a2 = { 3, 9 };

        // Function Call
        merge(a1, a2, a1.length, a2.length);

        System.out.print("First Array: ");
        for (int i = 0; i < a1.length; i++) {
            System.out.print(a1[i] + " ");
        }

        System.out.println();

        System.out.print("Second Array: ");
        for (int i = 0; i < a2.length; i++) {
            System.out.print(a2[i] + " ");
        }
    }
}
```

4. Kadane's Algorithm
   https://www.youtube.com/watch?v=w_KEocd__20&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=5

5. Merge Overlapping Subintervals
   https://www.youtube.com/watch?v=2JzRBPFYbKE&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=6

6. Find the duplicate in an array of N+1 integers.
   (Ignore the video quality, as this was the first video which I recorded)
   https://www.youtube.com/watch?v=32Ll35mhWg0&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=1

Day2: (Arrays)

1. Set Matrix Zeros
   (https://www.youtube.com/watch?v=M65xBewcqcI&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=7)

2. Pascal Triangle
   https://www.youtube.com/watch?v=6FLvhQjZqvM&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=8

3. Next Permutation
   https://www.youtube.com/watch?v=LuLCLgMElus&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=9

4. Inversion of Array (Using Merge Sort)
   https://www.youtube.com/watch?v=kQ1mJlwW-c0&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=10
5. Stock Buy and Sell
   https://www.youtube.com/watch?v=eMSfBgbiEjk&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=11

6. Rotate Matrix
   https://www.youtube.com/watch?v=Y72QeX0Efxw&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=12

Day3: (Arrays/maths)

1. Search in a 2D matrix
   https://www.youtube.com/watch?v=ZYpYur0znng&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=13

2. Pow(X,n)
   https://www.youtube.com/watch?v=l0YC3876qxg&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=14

3. Majority Element (>N/2 times)
   https://www.youtube.com/watch?v=AoX3BPWNnoE&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=15

4. Majority Element (>N/3 times)
   https://www.youtube.com/watch?v=yDbkQd9t2ig&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=16

5. Grid Unique Paths
   https://www.youtube.com/watch?v=t_f0nwwdg5o&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=17

6. Reverse Pairs (Leetcode)
   https://www.youtube.com/watch?v=S6rsAlj_iB4&list=PLgUwDviBIf0rPG3Ictpu74YWBQ1CaBkm2&index=18

7. Go through Puzzles from GFG (Search on own)

Day4: (Hashing)

1. 2 Sum problem
   https://www.youtube.com/watch?v=dRUpbt8vHpo&list=PLgUwDviBIf0rVwua0kKYlsS_ik_1lyVK_&index=1

2. 4 Sum problem
   https://www.youtube.com/watch?v=4ggF3tXlAp0&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=20

3. Longest Consecutive Sequence
   https://www.youtube.com/watch?v=qgizvmgeyUM&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=21

4. Largest Subarray with 0 sum
   https://www.youtube.com/watch?v=xmguZ6GbatA&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=22

5. Count number of subarrays with given XOR(this clears a lot of problems)
   https://www.youtube.com/watch?v=lO9R5CaGRPY&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=23

6. Longest substring without repeat
   https://www.youtube.com/watch?v=qtVh-XEpsJo&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=25

Day5: (LinkedList)

1. Reverse a LinkedList
   https://www.youtube.com/watch?v=iRtLEoL-r-g&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=26

2. Find the middle of LinkedList
   https://www.youtube.com/watch?v=sGdwSH8RK-o&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=27

3. Merge two sorted Linked List
   https://www.youtube.com/watch?v=Xb4slcp1U38&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=28

4. Remove N-th node from the back of LinkedList
   https://www.youtube.com/watch?v=Lhu3MsXZy-Q&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=29

5. Delete a given Node when a node is given. (0(1) solution)
   https://www.youtube.com/watch?v=icnp4FJdZ_c&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=30

6. Add two numbers as LinkedList
   https://www.youtube.com/watch?v=LBVsXSMOIk4&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=31

Day6:

1. Find intersection point of Y LinkedList
   https://www.youtube.com/watch?v=u4FWXfgS8jw&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=32

2. Detect a cycle in Linked List
   https://www.youtube.com/watch?v=354J83hX7RI&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=33

3. Reverse a LinkedList in groups of size k.
   https://www.youtube.com/watch?v=Of0HPkk3JgI&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=33

4. Check if a LinkedList is a palindrome or not.
   https://www.youtube.com/watch?v=-DtNInqFUXs&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=35

5. Find the starting point of the Loop of LinkedList
   https://www.youtube.com/watch?v=QfbOhn0WZ88&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=36

6. Flattening of a LinkedList
   https://www.youtube.com/watch?v=ysytSSXpAI0&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=37

7. Rotate a LinkedList
   https://www.youtube.com/watch?v=9VPm6nEbVPA&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=38

Day7: (2-pointer)

1. Clone a Linked List with random and next pointer
   https://www.youtube.com/watch?v=VNf6VynfpdM&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=39

2. 3 sum
   https://www.youtube.com/watch?v=onLoX6Nhvmg&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=40

3. Trapping rainwater
   https://www.youtube.com/watch?v=m18Hntz4go8&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=41

4. Remove Duplicate from Sorted array
   https://www.youtube.com/watch?v=Fm_p9lJ4Z_8&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=42

5. Max consecutive ones
   https://www.youtube.com/watch?v=Mo33MjjMlyA&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=43

Day8: (Greedy)

1. N meeting in one room
   https://www.youtube.com/watch?v=II6ziNnub1Q&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=44

2. Minimum number of platforms required for a railway
   https://www.youtube.com/watch?v=dxVcMDI7vyI&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=45

3. Job sequencing Problem
   https://www.youtube.com/watch?v=LjPx4wQaRIs&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=46

4. Fractional Knapsack Problem
   https://www.youtube.com/watch?v=F_DDzYnxO14&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=48

5. Greedy algorithm to find minimum number of coins
   https://www.youtube.com/watch?v=mVg9CfJvayM&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=47

6. Activity Selection (it is same as N meeting in one room)
   https://www.youtube.com/watch?v=II6ziNnub1Q&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=44

Day9 (Recursion):

1. Subset Sums
   https://www.youtube.com/watch?v=rYkfBRtMJr8&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=52

2. Subset-II
   https://www.youtube.com/watch?v=RIn3gOkbhQE&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=53

3. Combination sum-1
   https://www.youtube.com/watch?v=OyZFFqQtu98&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=49

4. Combination sum-2
   https://www.youtube.com/watch?v=G1fRTGRxXU8&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=50

5. Palindrome Partitioning
   https://www.youtube.com/watch?v=WBgsABoClE0&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=51

6. K-th permutation Sequence
   https://www.youtube.com/watch?v=wT7gcXLYoao&list=PLgUwDviBIf0p4ozDR_kJJkONnb1wdx2Ma&index=55

Day10: (Recursion and Backtracking)

1. Print all Permutations of a string/array
2. N queens Problem
3. Sudoku
4. M coloring Problem (Graph prob)
5. Rat in a Maze
6. Word Break (print all ways)

-

Day11: (Divide and Conquer)

1. 1/N-th root of an integer (use binary search) (square root, cube root, ..)
2. Matrix Median

3. Find the element that appears once in sorted array, and rest element appears twice (Binary search)
4. Search element in a sorted and rotated array/ find pivot where it is rotated
5. Median of 2 sorted arrays
6. K-th element of two sorted arrays

Day12: (Bits) (Optional, very rare topic in interviews, but if you have time left, someone might ask)

1. Check if a number if a power of 2 or not in O(1)
2. Count total set bits
3. Divide Integers without / operator
4. Power Set (this is very important)
5. Find MSB in o(1)
6. Find square of a number without using multiplication or division operators.

Day13: (Stack and Queue)

1. Implement Stack / Implement Queue
2. +
3. Implement Stack using Queue
4. Implement Queue using Stack
5. Check for balanced parentheses
6. Next Greater Element

Day14:

1. Next Smaller Element
2. LRU cache (vvvv. imp)
3. Largest rectangle in histogram
4. Sliding Window maximum
5. Implement Min Stack
6. Rotten Orange (Using BFS)

Day15: (String)

1. Reverse Words in a String

2. Longest Palindrome in a string
3. Roman Number to Integer and vice versa
4. Implement ATOI/STRSTR
5. Longest Common Prefix
6. Rabin Karp

Day16: (String)

1. Prefix Function/Z-Function
2. KMP algo
3. Min
4. Check for Anagrams
5. Count and Say
6. Compare version numbers

Day17: (Binary Tree)

1. Inorder Traversal (with recursion and without recursion)
2. Preorder Traversal (with recursion and without recursion)
3. Postorder Traversal (with recursion and without recursion)
4. LeftView Of Binary Tree
5. Bottom View of Binary Tree
6. Top View of Binary Tree

Day18: (Binary Tree)

1. Level order Traversal / Level order traversal in spiral form
2. Height of a Binary Tree
3. Diameter of Binary Tree
4. Check if Binary tree is height balanced or not
5. LCA in Binary Tree
6. Check if two trees are identical or not

Day 19: (Binary Tree)

1. Maximum path sum
2. Construct Binary Tree from inorder and preorder
3. Construct Binary Tree from Inorder and Postorder

4. Symmetric Binary Tree
5. Flatten Binary Tree to LinkedList
6. Check if Binary Tree is mirror of itself or not

Day 20: (Binary Search Tree)

1. Populate Next Right pointers of Tree
2. Search given Key in BST
3. Construct BST from given keys.
4. Check is a BT is BST or not
5. Find LCA of two nodes in BST
6. Find the inorder predecessor/successor of a given Key in BST.

Day21: (BinarySearchTree)

1. Floor and Ceil in a BST
2. Find K-th smallest and K-th largest element in BST (2 different Questions)
3. Find a pair with a given sum in BST
4. BST iterator
5. Size of the largest BST in a Binary Tree
6. Serialize and deserialize Binary Tree

Day22: (Mixed Questions)

1. Binary Tree to Double Linked List
2. Find median in a stream of running integers.
3. K-th largest element in a stream.
4. Distinct numbers in Window.
5. K-th largest element in an unsorted array.
6. Flood-fill Algorithm

Day23: (Graph)

1. Clone a graph (Not that easy as it looks)
2. DFS
3. BFS
4. Detect A cycle in Undirected Graph/Directed Graph
5. Topo Sort

6. Number of islands (Do in Grid and Graph both)
7. Bipartite Check

Day24: (Graph)

1. SCC(using KosaRaju's algo)
2. Djisktra's Algorithm
3. Bellman Ford Algo
4. Floyd Warshall Algorithm
5. MST using Prim's Algo
6. MST using Kruskal's Algo

Day25: (Dynamic Programming)

1. Max Product Subarray
2. Longest Increasing Subsequence
3. Longest Common Subsequence
4. 0-1 Knapsack
5. Edit Distance
6. Maximum sum increasing subsequence
7. Matrix Chain Multiplication

Day26: (DP)

1. Maximum sum path in the matrix, (count paths and similar type do, also backtrack to find the maximum path)
2. Coin change
3. Subset Sum
4. Rod Cutting
5. Egg Dropping
6. Word Break
7. Palindrome Partitioning (MCM Variation)
8. Maximum profit in Job scheduling

Day27:

1. Revise OS notes that you would have made during your sem

2. If not made notes, spend 2 or 3 days and make notes from Knowledge Gate.

Day28:

1. Revise DBMS notes that you would have made during your semesters.
2. If not made notes, spend 2 or 3 days and make notes from Knowledge Gate.

Day29:

1. Revise CN notes, that you would have made during your sem.
2. If not made notes, spend 2 or 3 days and make notes from Knowledge Gate.

Day30:

1. Make a note of how will your represent your projects, and prepare all questions related to tech which you have used in your projects. Prepare a note which you can say for 3-10 minutes when he asks you that say something about the project.

Hurrah!! You are ready for your placement after a month of hard work without a cheat day.

LECTURE'S (GATE SMASHER IS GOOD, BUT CHECK IF YOU WANT TO)

Following are the lectures on Computer Networks.
*If at all you are preparing for interview :innocent: *
CN 1:
https://zoom.us/rec/share/yuxbJozg2VhIftbHzGaCc7EENbrdaaa81SRNqPMJmUaR3sQFeD6ntUBt2S7dNzjS
CN 2:
https://zoom.us/rec/share/6ZV7CoP70iRLXYXxxkj5WJJ-EqH4T6a81nAXqKcFnUxj2fH1pVc2avN0SwZL_eZU
CN 3:
https://zoom.us/rec/share/4utPHr3V8EhOWYmRtgLER4oEXYTMX6a8gChKq_cFyxoGUBiq4d4ETF8Qooi9xWe4
CN 4:
https://zoom.us/rec/share/3ZBXK6Hgq0pJRNb_63z0WK0GGLbsT6a8hydL8qUEn0ZRo-mhIjkiOqPt9JUdKYnp
CN 5:
https://zoom.us/rec/share/7PItNu3x-m5OfqPH8kDSY6gjF53FT6a8hyEY8_dcyBw1YcEseeOsDBHwgdI0KxtP

Here are the links of lectures on DBMS and OS.
DBMS 1 -
https://api.zoom.us/recording/share/RObkYnNpCxVyId-xasFEn4YIuhpJwgn_85NoewCiQciwIumekTziMw
DBMS 2 -
https://api.zoom.us/recording/share/rJRMvVY6ig9yaG92hEz7MsXzcp0o-kAmacH5r8YPVNiwIumekTziMw
OS 1 -
https://api.zoom.us/recording/share/kekqkxmmU_b24PyFo-bCyYiMV9Rd4rLuQgfMzBOfYqiwIumekTziMw
OS 2 -
https://api.zoom.us/recording/share/bdRLvFk358egTH-pWradAQ25mKpzn0FDABdmyk3xABCwIumekTziMw
OS 3 -
https://api.zoom.us/recording/share/Ppw_oHRGpG9JjzBqvy1rYc9xFniUXDcXSipC8HXHaL6wIumekTziMw

Notes pdf

DBMS Notes (Scaler) (Tarun Sir)

DBMS - 1
https://docs.google.com/document/d/12VkoaPxBbqf6UYM9Zx7yjWiCD8v2sUhmGuxbelvAwrE/edit#
DBMS - 2
https://docs.google.com/document/d/1mv5a47y4HwGHrIdauJaI8kyjzbDuouViPHbBkgVbTro/edit#