Self-Practice No. : **2**

Topics Covered : **Basic Math, Control Flow, Arrays, Functions, Bit Manipulation**

Date : **09-05-2024**

Solve the following problems

Q No.	Question Detail	Level
1	**Count ways to reach the n'th stair** **Problem Statement** : There are n stairs, a person standing at the bottom wants to reach the top. The person can climb either 1 stair or 2 stairs at a time. Count the number of ways, the person can reach the top (order does matter). **Example 1:** **Input:** n = 4 **Output:** 5 **Explanation:** You can reach 4th stair in 5 ways. Way 1: Climb 2 stairs at a time. Way 2: Climb 1 stair at a time. Way 3: Climb 2 stairs, then 1 stair and then 1 stair. Way 4: Climb 1 stair, then 2 stairs then 1 stair. Way 5: Climb 1 stair, then 1 stair and then 2 stairs. **Example 2:** **Input:** n = 10 **Output**: 89 **Explanation:**	Medium

	There are 89 ways to reach the 10th stair. **Constraints:** 1 ≤ n ≤ 10^4	
2	**Largest prime factor** **Problem statement** : Given a number N, the task is to find the largest prime factor of that number. **Example 1:** **Input:** N = 5 **Output:** 5 **Explanation:** 5 has 1 prime factor i.e 5 only. **Example 2:** **Input:** N = 24 **Output:** 3 **Explanation:** 24 has 2 prime factors 2 and 3 in which 3 is greater. **Constraints:** 2 <= N <= 10^9	Medium
3	**Mirror Upper triangle star pattern** **Problem Statement:** You are tasked with creating a Java program to print a mirror upper star triangle pattern. Given an integer N, the program should print a pattern with N rows where each row contains a mirrored upper triangle of stars. **Sample input 1:** N=7 **Sample output 1:**	Medium

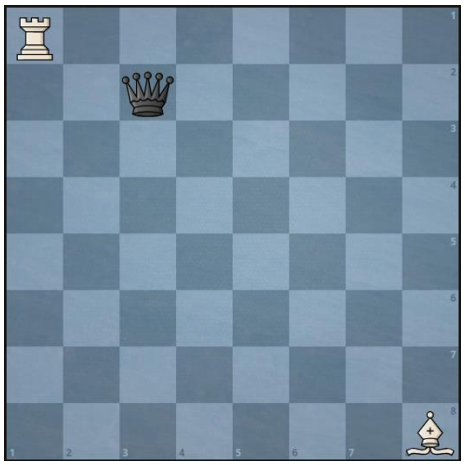
	<pre> * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *</pre> **Constraints**: 1 <= N <= 10	
4	**Occurrences of given digit** **Problem statement:** Rohan needs to find the number of occurrence of a digit in a given number .The input may lie within the range of integer. If the digit does not occur in the input it should print 0 else the count of digits. **Sample Input 1:** 1223457 D= 2 **Sample Output 1:** The digit 2 occurs 2 times in the number 1223457. **Sample Input 2:** 9876543 D=0 **Sample Output 2:** The digit 0 occurs 0 times in the number 9876543. **Constraints:**	Medium

	1 <= N <=10^6 0<=D<=9 Where D represents the digits	
5	**Single Number III** **Problem Statement**: Given an integer array nums, in which exactly two elements appear only once and all the other elements appear exactly twice. Find the two elements that appear only once. You can return the answer in any order.You must write an algorithm that runs in linear runtime complexity and uses only constant extra space. **Example 1:** **Input:** nums = [1,2,1,3,2,5] **Output:** [3,5] **Explanation:** [5, 3] is also a valid answer. **Example 2:** **Input:** nums = [-1,0] **Output:** [-1,0] **Constraints:** • 2 <= nums.length <= 3 * 10^4 • -2^31 <= nums[i] <= 2^31 - 1 • Each integer in nums will appear twice, only two integers will appear once.	Medium
6	**Find Xor-Beauty of Array** **Problem Statement:** You are given a 0-indexed integer array nums.The effective value of three indices i, j, and k is defined as ((nums[i] | nums[j]) & nums[k]).The xor-beauty of the array is the XORing of the effective values of all the possible triplets of indices (i, j, k) where 0 <= i, j, k < n.Return *the xor-beauty of* nums. Note that: • val1 | val2 is bitwise OR of val1 and val2. • val1 & val2 is bitwise AND of val1 and val2.	Medium

	Example 1: **Input:** nums = [1,4] **Output:** 5 **Explanation:** The triplets and their corresponding effective values are listed below: - (0,0,0) with effective value ((1 | 1) & 1) = 1 - (0,0,1) with effective value ((1 | 1) & 4) = 0 - (0,1,0) with effective value ((1 | 4) & 1) = 1 - (0,1,1) with effective value ((1 | 4) & 4) = 4 - (1,0,0) with effective value ((4 | 1) & 1) = 1 - (1,0,1) with effective value ((4 | 1) & 4) = 4 - (1,1,0) with effective value ((4 | 4) & 1) = 0 - (1,1,1) with effective value ((4 | 4) & 4) = 4 Xor-beauty of array will be bitwise XOR of all beauties = 1 ^ 0 ^ 1 ^ 4 ^ 1 ^ 4 ^ 0 ^ 4 = 5. **Example 2:** **Input: n**ums = [15,45,20,2,34,35,5,44,32,30] **Output:** 34 **Constraints:** • 1 <= nums.length <= 10^5 • 1 <= nums[i] <= 10^9	
7	**Number of Steps to Reduce a Number in Binary Representation to One** **Problem Statements:** Given the binary representation of an integer as a string s, return the number of steps to reduce it to 1 under the following rules: If the current number is even, you have to divide it by 2.If the current number is odd, you have to add 1 to it.It is guaranteed that you can always reach one for all test cases. **Example 1:** **Input**: s = "1101" **Output:** 6	Medium

	Explanation: "1101" corresponds to number 13 in their decimal representation. Step 1) 13 is odd, add 1 and obtain 14. Step 2) 14 is even, divide by 2 and obtain 7. Step 3) 7 is odd, add 1 and obtain 8. Step 4) 8 is even, divide by 2 and obtain 4. Step 5) 4 is even, divide by 2 and obtain 2. Step 6) 2 is even, divide by 2 and obtain 1. **Example 2:** **Input:** s = "10" **Output:** 1 **Constraints:** • 1 <= s.length <= 500 • s consists of characters '0' or '1' • s[0] == '1'	
8	**Minimum Moves to Capture the Queen** **Problem Statement:** There is a 1-indexed 8 x 8 chessboard containing 3 pieces. You are given 6 integers a, b, c, d, e, and f where: • (a, b) denotes the position of the white rook. • (c, d) denotes the position of the white bishop. • (e, f) denotes the position of the black queen. Given that you can only move the white pieces, return the minimum number of moves required to capture the black queen. Note that: • Rooks can move any number of squares either vertically or horizontally, but cannot jump over other pieces. • Bishops can move any number of squares diagonally, but cannot jump over other pieces. • A rook or a bishop can capture the queen if it is located in a square that they can move to. • The queen does not move.	Medium

Example 1:



Input: a = 1, b = 1, c = 8, d = 8, e = 2, f = 3

Output: 2

Explanation: We can capture the black queen in two moves by moving the white rook to (1, 3) then to (2, 3).
It is impossible to capture the black queen in less than two moves since it is not being attacked by any of the pieces at the beginning.

Constraints:

- 1 <= a, b, c, d, e, f <= 8
- No two pieces are on the same square.

9	**Rectangular numbers** **Problem statement :** Print the pattern in such a way that the outer rectangle is of the number 'N' and the number goes on decreasing as we move inside the rectangles. For **example,** if 'N' = 4, then pattern will be: 4 4 4 4 4 4 4 4 3 3 3 3 3 4 4 3 2 2 2 3 4 4 3 2 1 2 3 4 4 3 2 2 2 3 4 4 3 3 3 3 3 4 4 4 4 4 4 4 4	Medium

.

Sample Input 1:

2

2

1

Sample Output 1:

2 2 2

2 1 2

2 2 2

1

Explanation Of Sample Input 1:

Test case 1:

For the first test case of sample output 1, as the number is 2, so the outermost rectangle is of number 2. The moment we get inside the rectangle, we reduce the number by 1 and make another rectangle.

Test case 2:

For the second test case of sample output 1, as the number is 1, so the outermost rectangle is of number 1.

Sample Input 2:

1

4

Sample Output 2:

4 4 4 4 4 4 4

4 3 3 3 3 3 4

4 3 2 2 2 3 4

4 3 2 1 2 3 4

4 3 2 2 2 3 4

4 3 3 3 3 3 4

4 4 4 4 4 4 4

Explanation Of Sample Input 2:

Test case 1:

For the first test case of sample output 2, as the number is 4, so the outermost rectangle is of number 24. The moment we get inside the rectangle, we reduce the number by 1 and make another rectangle. This process goes on till we reach 1.

	Constraints: 1 <= T <= 5 1 <= N <= 100	
10	**Stickler Thief** **Problem Statement:** Stickler the thief wants to loot money from a society having n houses in a single line. He is a weird person and follows a certain rule when looting the houses. According to the rule, he will never loot two consecutive houses. At the same time, he wants to maximize the amount he loots. The thief knows which house has what amount of money but is unable to come up with an optimal looting strategy. He asks for your help to find the maximum money he can get if he strictly follows the rule. ith house has a[i] amount of money present in it. **Example 1:** **Input:** n = 5 a[] = {6,5,5,7,4} **Output:** 15 **Explanation:** Maximum amount he can get by looting 1st, 3rd and 5th house. Which is 6+5+4=15. **Example 2:** **Input:** n = 3 a[] = {1,5,3} **Output:** 5 **Explanation:** Loot only 2nd house and get maximum amount of 5.	Medium

	Constraints: 1 ≤ n ≤ 10^5 1 ≤ a[i] ≤ 10^4	