**Experiment 3.1**

**Competitive Coding Lab 8(Dynamic Programming)**

**Student Name: Sahul Kr. Parida UID: 20BCS4919**

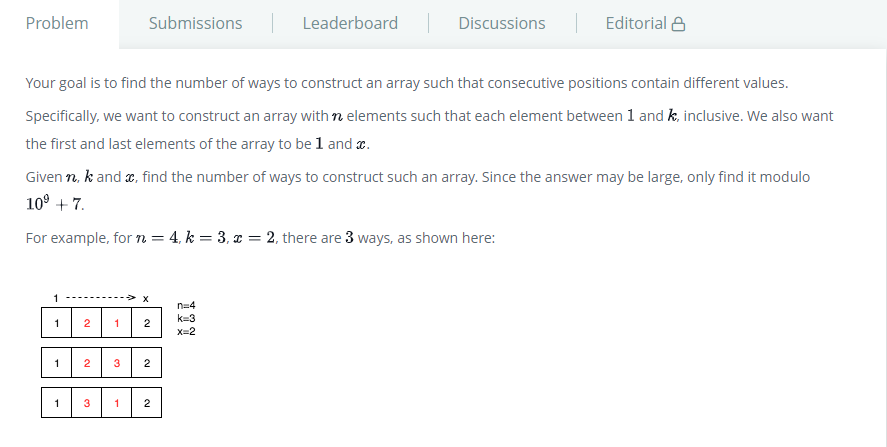
**Branch: CSE Section/Group: WM-904/B**

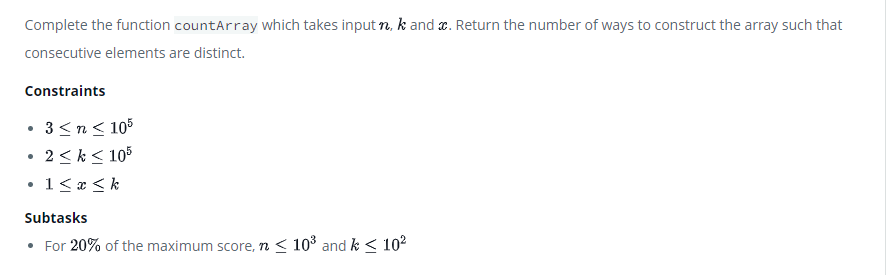
**Semester: 5th Date of Performance: 08/11/22**

**Subject Name: Competitive Coding(CC) Subject Code: 20CSP-314**

**PROBLEM STATEMENT 8.1: -**

<https://www.hackerrank.com/challenges/construct-the-array/problem?isFullScreen=false>





**SOLUTION:**

public static long countArray(int n, int k, int x) {

    // Return the number of ways to fill in the array.

    long dp[][] = new long[n][2];

        dp[0][0] = 1;

        dp[0][1] = 0;

        for (int i=1;i<n;i++)

        {

            dp[i][0] = (dp[i-1][1] \* (k-1)) % 1000000007;

            dp[i][1] = (dp[i-1][0] + dp[i-1][1] \* (k-2)) % 1000000007;

        }

        if (x == 1)

        {

            return dp[n-1][0];

        }

        else

        {

            return dp[n-1][1];

        }

    }

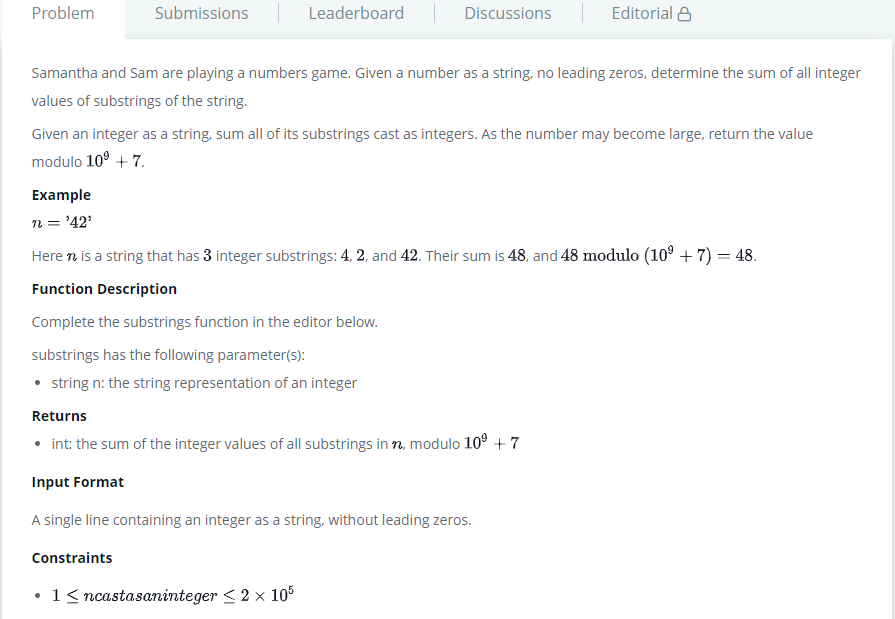
}

# TEST CASES:

# 

**PROBLEM STATEMENT 8.2: -**

<https://www.hackerrank.com/challenges/sam-and-substrings/problem?isFullScreen=false>



**SOLUTION:**

import java.io.\*;

public class Solution

{

    private final static int MOD = 1000000007;

    public static void main(String[] args) throws IOException

    {

        int[] balls = strNumToArr((new BufferedReader(new InputStreamReader(System.in))).readLine());

        int n = balls.length;

        for(int i = n - 2; i >= 0; --i)

        {

            balls[i] = (int)((balls[i+1] + (((long)balls[i])\*(n - i))%MOD)%MOD);

        }

        int pow = 1;

        int total = 0;

        for(int i = 0; i < n; ++i)

        {

            total = (int)((total + (((long)balls[i])\*pow)%MOD)%MOD);

            pow = (int)((pow\*10L)%MOD);

        }

        System.out.print(total);

    }

    private static int[] strNumToArr(String str)

    {

        int n = str.length();

        int[] ar = new int[n];

        for(char c : str.toCharArray())

        {

            ar[--n] = c - '0';

        }

        return ar;

    }

}

# TEST CASES:

# 