Day 1

Student Name: Jaysheel Bhatt **UID: 21BCS5088**
Branch: B.E. CSE **Section/Group: CC-613-A**
Subject Name: IT Skills **Date of Performance: 24-06-2024**

Question 1. Helping Machine

A machine recycles scrap bottles, the size of the bottle is categorised into types. For n number of bottles you have to calculate the number of bottles in each category type.

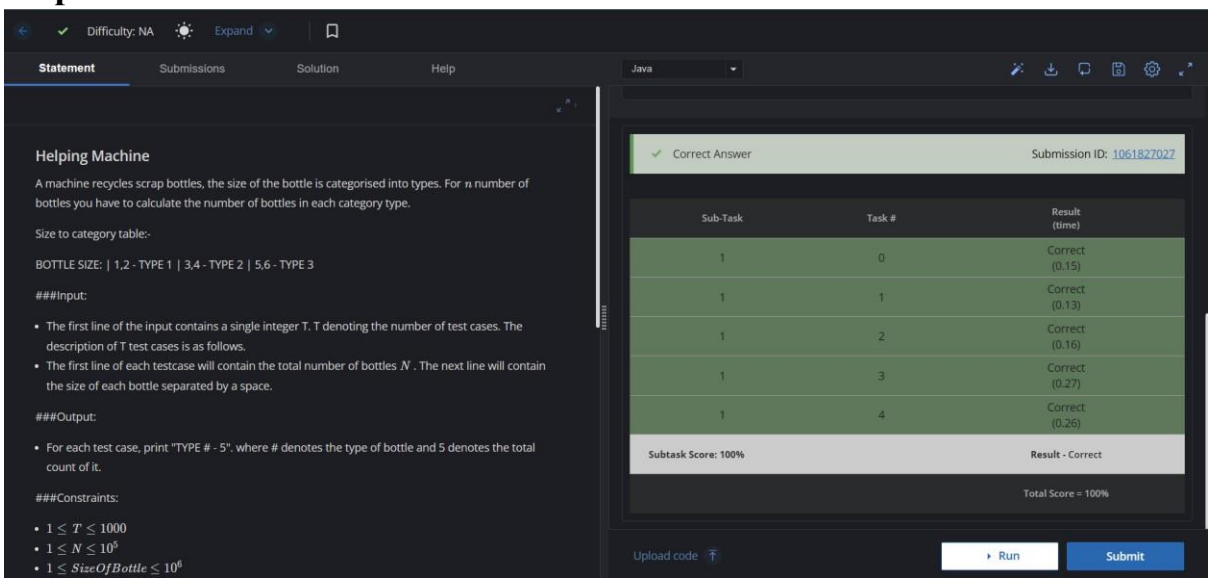
Link: https://www.codechef.com/problems/CMAN2?tab=statement

Code:

```
import java.util.*;
import java.lang.*;
import java.io.*; class
Codechef
{
        public static void main (String[] args) throws java.lang.Exception
        {
                Scanner sc=new Scanner(System.in);
                int n=sc.nextInt();
        for(int i=0;i<n;i++){
                   int noOfBottle=sc.nextInt();
                   int type1=0;
int type2=0;                int type3=0;
                   for(int j=0;j<noOfBottle;j++){

                      int bottle=sc.nextInt();
if(bottle==1 || bottle==2){
                         type1++;
                      }
                      else if(bottle==3 || bottle==4){
                         type2++;
                      }
                      else if (bottle==5 || bottle==6){
                         type3++;
                      }
```

```
                    }


System.out.println("TYPE 1 - "+type1+" TYPE 2 - "+type2+" TYPE 3 - "+type3);
                }
        }
}
```

Output:



Question 2. Sum Of Digits

You are being given a number N . (1<=N<=1000). You have to print the sum of digits of that particular number.

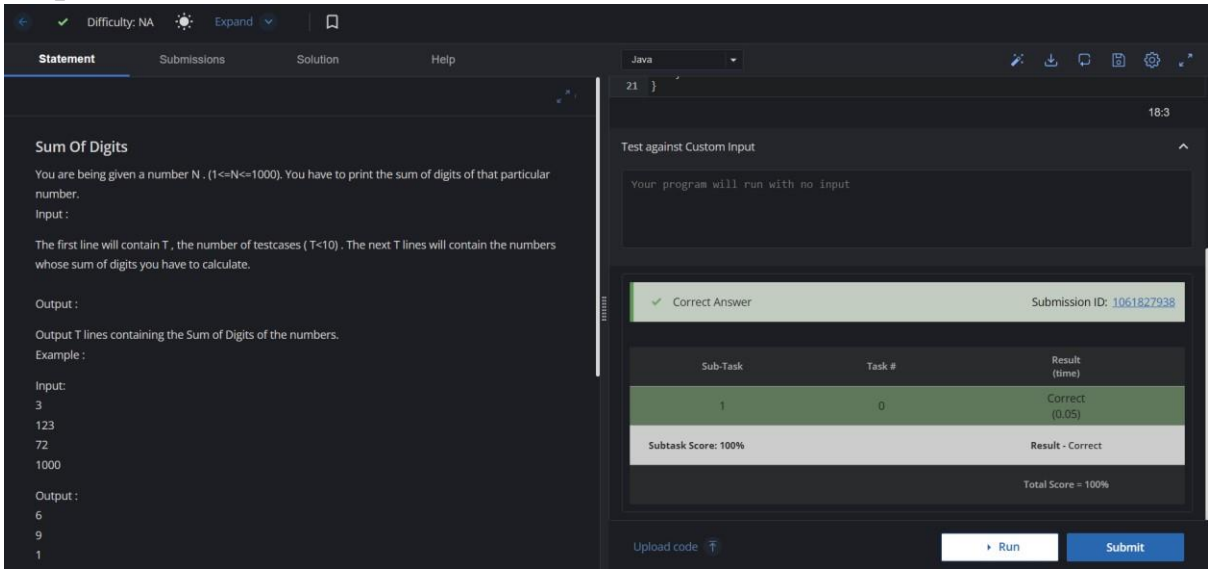
Link: https://www.codechef.com/submit/DEM3?tab=statement

Code:

```
import java.util.*;
import java.lang.*;
import java.io.*; class
Codechef
{
        public static void main (String[] args) throws java.lang.Exception
        {
                BufferedReader bf = new BufferedReader(new
InputStreamReader(System.in));
                int t = Integer.parseInt(bf.readLine());
```

```
                while(t>0){
                    String n = bf.readLine();
                    char[] v = n.toCharArray();
                    int ac = 0;
            for(char i:v){
                ac +=
Character.getNumericValue(i);
                    }
                    System.out.println(ac);
                    t--;
                }
            }
        }
```

Output:



Question3. Correct Sentence

You are given a sentence as a sequence of K words $S1$, $S2$, …, SK. Determine whether it could be a sentence formed by Chef, i.e. if it contains only the characters from the two given languages and each word contains only characters from a single language.

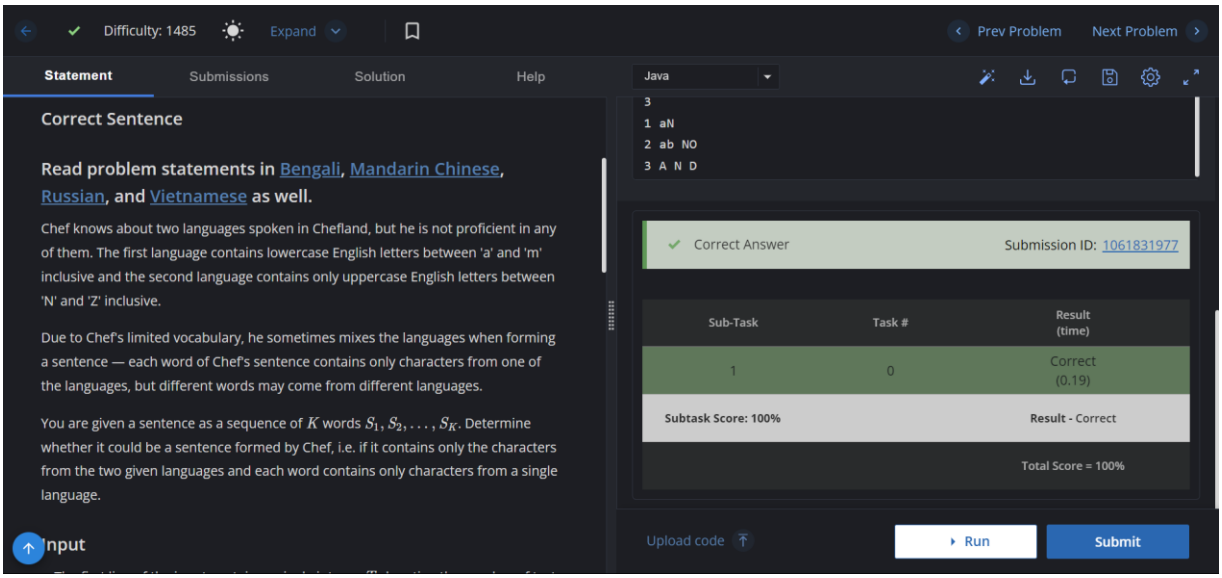
Link: https://www.codechef.com/problems/CORTSENT?tab=statement

Code:

```
#include <iostream>
```

```
#include <bits/stdc++.h>
#include <string> using
namespace std;
int main()
{
    int t;
    cin >> t;
    while (t--)
    {

        cin >> k;
        vector<string> v;
    int ans=0;
        for (int i = 0; i < k; i++)
        for (int j = 0; j < k; j++)
        {
            string n = v[j];
            int cnt1=0,cnt2=0;
            for (int i = 0; i < n.size(); i++)
            {
                if ((n[i] >= 'a' and n[i] <= 'm')  )
                {
                    cnt1++;
                }
                else if( (n[i] >= 'N' and n[i] <= 'Z')) cnt2++;
                if(cnt1==n.size() and cnt2==0){
                    ans++;
                }
                else if( cnt2==n.size() and cnt1==0){
                    ans++;
                }
            }
        }
        if(ans==k) cout<<"YES"<<endl;
        else cout<<"NO"<<endl;
    }
    return 0;
}
```

Output:



Question 4. Basic Calculator

Given a string s representing a valid expression, implement a basic calculator to evaluate it, and return the result of the evaluation.

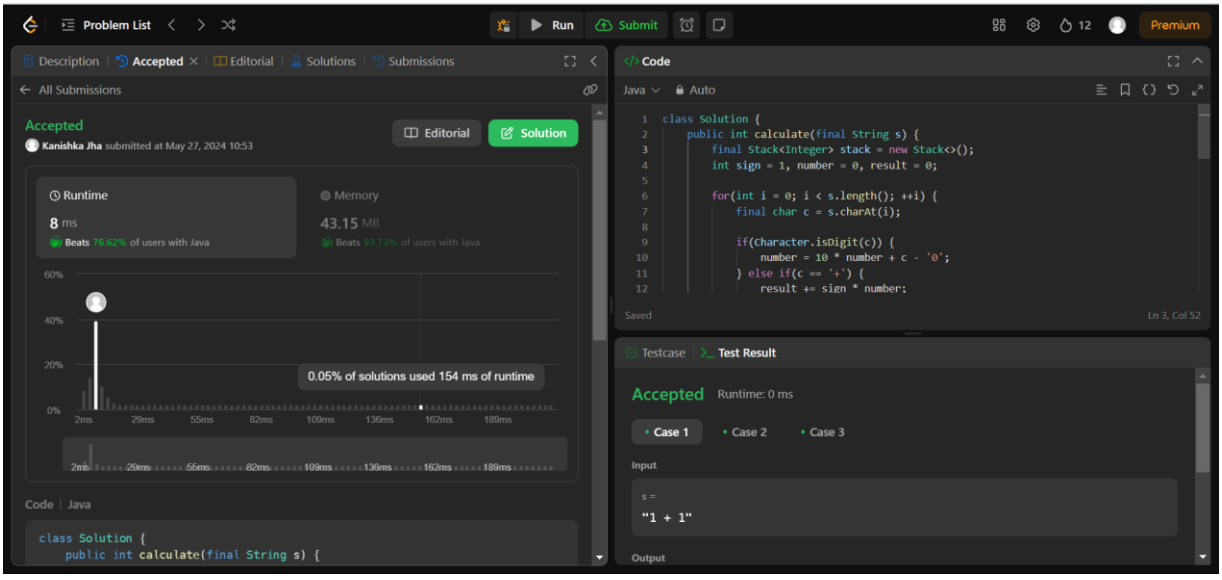
Link: https://leetcode.com/problems/basic-calculator/description

Code:

```
class Solution {
    public int calculate(final String s) {
        final Stack<Integer> stack = new Stack<>();
        int sign = 1, number = 0, result = 0;

        for(int i = 0; i < s.length(); ++i) {
            final char c = s.charAt(i);

            if(Character.isDigit(c)) {
                number = 10 * number + c - '0';
            } else if(c == '+') {
                result += sign * number;
                number = 0;
                sign = 1;
            } else if(c == '-') {
                result += sign * number;
                number = 0;
                sign = -1;
            } else if(c == '(') {
```

```
                stack.push(result);
                stack.push(sign);
                sign = 1;
                result = 0;
            } else if(c == ')') {
                result += sign * number;
                number = 0;
                result *= stack.pop();
                result += stack.pop();
            }
        }

        if(number != 0)
            result += sign * number;

        return result;
    }
}
```

Output:



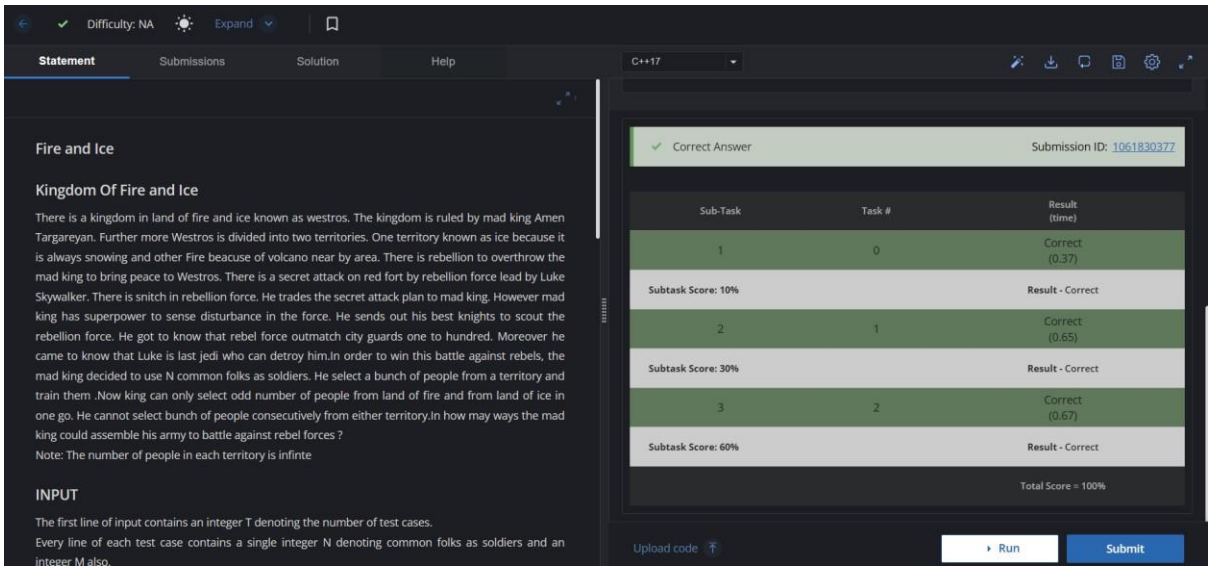
Question 5. Kingdom of Fire and Ice

Link: https://www.codechef.com/problems/FICE

Code:

```
#include <stdio.h> #include<inttypes.h>
void multiply(uint64_t F[2][2], uint64_t M[2][2],uint64_t k);
void power(uint64_t F[2][2], uint64_t n,uint64_t k); uint64_t
fib(uint64_t n,uint64_t k) { uint64_t F[2][2] =
{{1,1},{1,0}};   if (n == 0)     return 0;   power(F, n-1,k);
return F[0][0];
}
void power(uint64_t F[2][2], uint64_t n,uint64_t k)
{
 if( n == 0 || n == 1) return;
uint64_t M[2][2] = {{1,1},{1,0}};
power(F, n/2,k);   multiply(F, F,k);
if (n%2 != 0)      multiply(F, M, k);
}  void multiply(uint64_t F[2][2], uint64_t M[2][2],uint64_t
k) {


 uint64_t x =  (F[0][0]*M[0][0] + F[0][1]*M[1][0])%k;
uint64_t y =  (F[0][0]*M[0][1] + F[0][1]*M[1][1])%k;
uint64_t z =  (F[1][0]*M[0][0] + F[1][1]*M[1][0])%k;
uint64_t w =  (F[1][0]*M[0][1] + F[1][1]*M[1][1])%k;   F[0][0]
= x;
 F[0][1] = y;
 F[1][0] = z;
 F[1][1] = w;
} int
main() {
 uint64_t n,k,t;
scanf("%llu",&t);   while(t--)
 {
        scanf("%llu",&n);
        scanf("%llu",&k);
        printf("%llu\n", (2*fib(n,k))%k);
 }
return 0;
}
```

Output:



Question 6. Count Good Numbers

A digit string is good if the digits (0-indexed) at even indices are even and the digits at odd indices are prime (2, 3, 5, or 7).

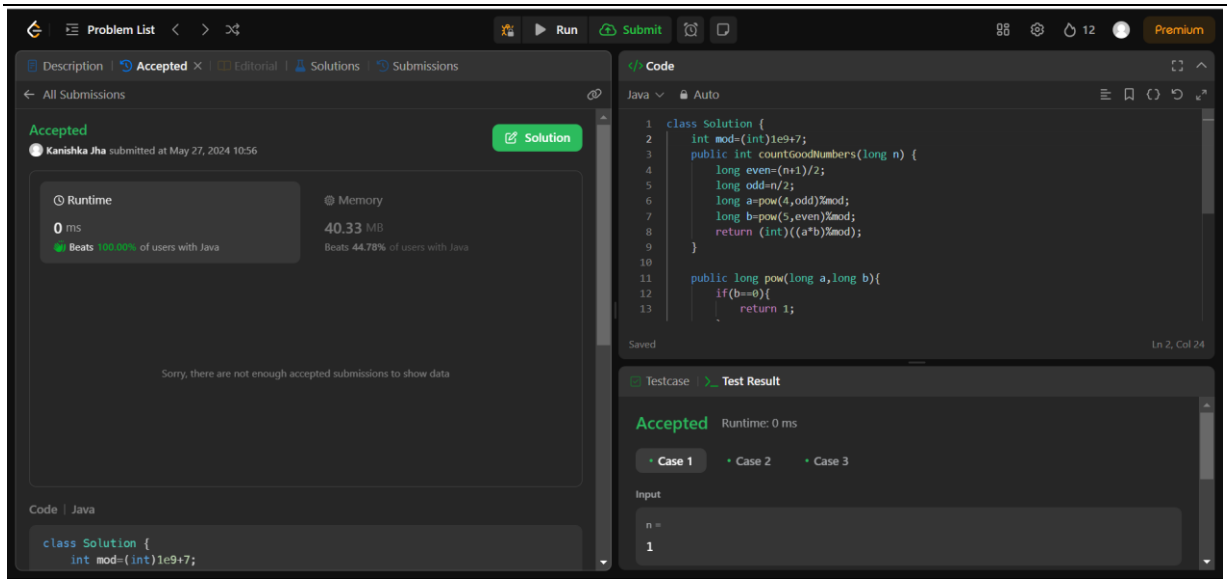
Link: https://leetcode.com/problems/count-good-numbers/description/

Code:

```
class Solution {
    int mod=(int)1e9+7;
    public int countGoodNumbers(long n) {
        long even=(n+1)/2;
        long odd=n/2;
        long a=pow(4,odd)%mod;
        long b=pow(5,even)%mod;
        return (int)((a*b)%mod);
    }

    public long pow(long a,long b){
        if(b==0){
            return 1;
        }
        long temp=pow(a,b/2);
        if(b%2==0){
            return (temp*temp)%mod;
        }
        else{
            return (a *temp*temp)%mod;
        }
```

```
        }
    }
```

Output:



Learning Outcome:

- Practical world problem-solving.
- Many new complex concepts.
- Data Structure and Algorithm Concepts.
- Working with Codchef and LeetCode Compiler.