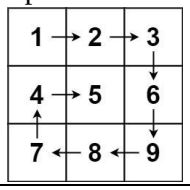
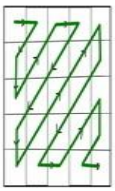
CSE3032 - Competitive Programming
WIN SEM (2022-2023) AMR
Class Number: AP2022236001007
Slot: L11+L12+L19+L20
ASSIGNMENT - 3

Last Date for Submission: Wednesday (11-02-2023)

Write the program using (C / C++ / Java / Python) to solve the following problems.

S.No	Problem Name	Link
1	Spiral Matrix 1 	https://www.hackerrank.com/contests/coding-test-1-bitshyderabad/challenges/spiral-matrix-1/problem
2	ZIG-ZAG Matrix	Input: 1 2 3 4 5 6 7 8 9 Output: 1 2 4 7 5 3 6 8 9 
3	Unique subsequences	https://www.hackerearth.com/practice/algorithms/stringalgorithm/basics-of-string-manipulation/practiceproblems/algorithm/unique-subsequence-264057c9/
4	subsequences	https://www.hackerearth.com/problem/algorithm/subsequences-2/
5	Good Subsequences	https://www.hackerearth.com/practice/algorithms/stringalgorithm/basics-of-string-manipulation/practiceproblems/algorithm/good-sequences-46c31a23/
6	SUBSEQ - Counting Subsequences	https://www.spoj.com/problems/SUBSEQ/
7	SUMMATION - SUMMATION	https://www.spoj.com/problems/SUMMATION/cstart=10

Note:

- If Code similarity is found, assignment will not be considered and Zero (0) Marks will be awarded.
- You have to upload a single document consisting of all the above programs and corresponding Output.
- You will be asked to explain the code, run and show the same program in the respective platforms (hacker rank / hacker earth / spoj)

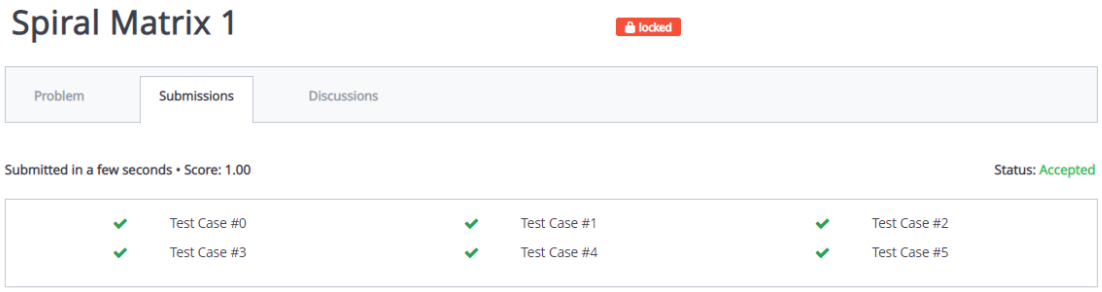
1) Spiral Matrix

Code:

```
import java.util.Scanner;
public class Main{
   static Scanner sc = new Scanner(System.in);
   static int[][] arr;
   static int n;
   static int rowStart;
   static int rowEnd;
   static int colStart;
   static int colEnd;
   public static void main(String [] args){
      n=sc.nextInt();
      arr=new int[n][n];
      for(int i=0;i<n;i++){
         for(int j=0;j<n;j++){
            arr[i][j]=sc.nextInt();
         }
      }
      rowStart=0;
      rowEnd=n-1;
      colStart=0;
      colEnd=n-1;
      while(rowStart<=rowEnd && colStart <= colEnd){
         for(int i=colStart;i<=colEnd;i++) {
            System.out.print(arr[rowStart][i] + " ");
         }
         rowStart++;
         for(int i=rowStart;i<=rowEnd;i++){
            System.out.print(arr[i][colEnd] + " ");
         }
         colEnd--;
         if(rowStart<=rowEnd){
```

```
            for(int i=colEnd;i>=colStart;i--){
                System.out.print(arr[rowEnd][i] + " ");
            }
            rowEnd--;
        }
        if(colStart<=colEnd){
            for(int i=rowEnd;i>=rowStart;i--){
                System.out.print(arr[i][colStart] + " ");
            }
            colStart++;
        }
    }
  }
}
```

Output:



2) Zig-Zag Matrix

Code:

```
import java.util.Scanner;
public class Assignments{
   static int[][] arr;
   static int n=3;
   static Scanner sc=new Scanner(System.in);
   public static void main(String [] args){
      arr=new int[n][n];
      for(int i=0;i<n;i++){
         for(int j=0;j<n;j++){
            arr[i][j]=sc.nextInt();
         }
      }
      int i=0,j=0,k=0;
      int[] a=new int[n*n];
      boolean up=true;
      while(k<n*n){
```

```
            a[k++]=arr[i][j];
            if(up){
               if(j==n-1){
                  i++;
                  up=false;
               }
                              else if(i==0){
                  j++;
                  up=false;
               }
                              else{
                  i--;
                  j++;
               }
            }
                  else{
               if(i==n-1){
                  j++;
                  up=true;
               }
                              else if(j==0){
                  i++;
                  up=true;
               }
                              else{
                  i++;
                  j--;
               }
            }
         }
         for(int num:a){
            System.out.print(num + " ");
         }
      }
   }
```

Output:

```
C:\Users\Windows\OneDrive\Documents\CP>javac Assignments.java

C:\Users\Windows\OneDrive\Documents\CP>java Assignments.java
1 2 3
4 5 6
7 8 9
1 2 4 7 5 3 6 8 9
C:\Users\Windows\OneDrive\Documents\CP>
```

3) Unique Subsequences

Code:

```
import java.util.*;
public class Main{
public static void main(String args[] ) throws Exception {
Scanner sc=new Scanner(System.in);
int T=sc.nextInt();
for(int i=0;i<T;i++)
{
int N=sc.nextInt();
String s=sc.next();
int count=1;
for(int j=0;j<N-1;j++)
{
if(s.charAt(j)==(s.charAt(j+1)))
{
continue;
}
count++;
}
System.out.println(count);
}
}
}
```

Output:

RESULT: Accepted

Refer judge environment

Score	Time (sec)	Memory (KiB)	Language
20	1.57828	88016	Java 8

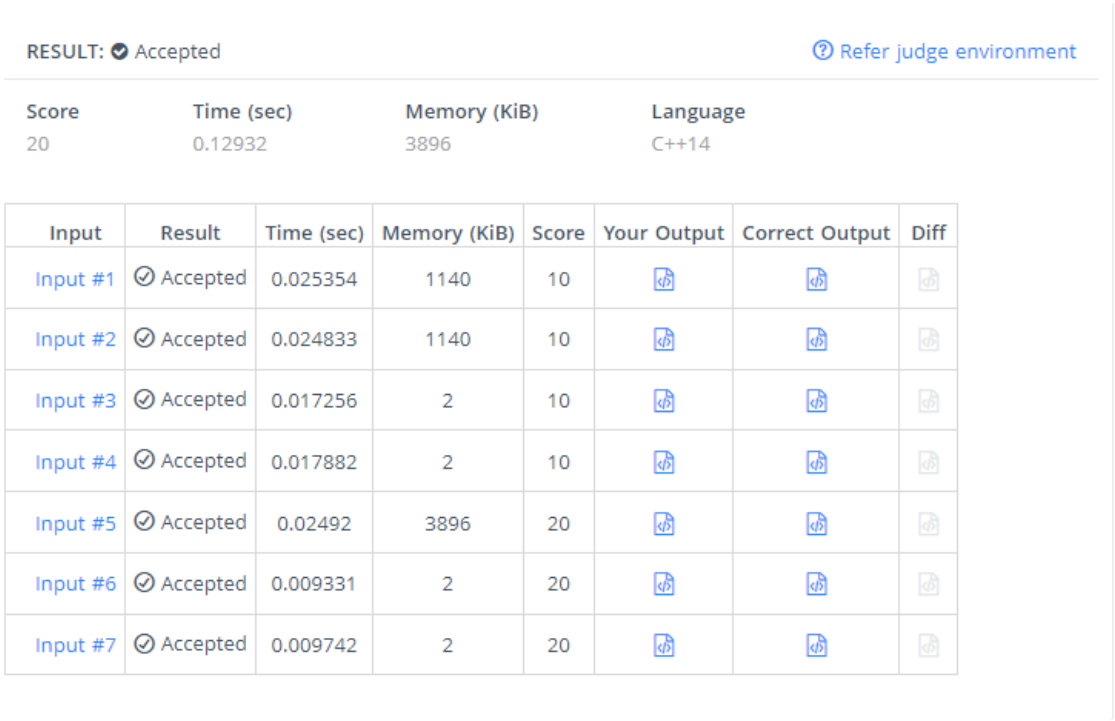
Input	Result	Time (sec)	Memory (KiB)	Score	Your Output	Correct Output	Diff
Input #1	Accepted	0.219867	88016	5			
Input #2	Accepted	0.228755	87876	5			
Input #3	Accepted	0.208829	86776	20			
Input #4	Accepted	0.189342	84472	20			
Input #5	Accepted	0.174619	83396	20			
Input #6	Accepted	0.181752	85860	5			
Input #7	Accepted	0.211551	88008	5			
Input #8	Accepted	0.163566	84748	20			

4) Subsequences

Code:

```
#include<bits/stdc++.h>
using namespace std;
int main(){
    int n,ele;
    cin>>n;
    vector<int> fre(100001),fre1(100001);
    for(int i=0;i<n;++i){
       cin >> ele;
       fre[ele]++;
       fre1[ele]++;
     }
     int i =100001;
     while(i>0 && !fre[i]) i--;
     i--;
    long long req = 0;
    while(i>0){
      if(fre[i] < fre[i+1]){
       req+=fre[i+1] - fre[i];
       fre[i] = fre[i+1];
      }
      i--;
     }
     if(req){
         cout<<req;
         return 0;
     }
    vector<vector<int>> subs(fre[1],vector<int>(0));
    for(int i=1;i<=100001;++i){
        for(int j=0;j<fre[i];++j){
            subs[j].push_back(i);
        }
    }
    cout<<fre[1]<<"\n";
    for(int i=fre[1]-1;i>=0;--i){
        for(int j = 0;j<subs[i].size();++j){
            cout<<subs[i][j]<<" ";
        }
        cout<<"\n";
    }
    return 0;
}
```

Output:



RESULT: Accepted — Refer judge environment

Score	Time (sec)	Memory (KiB)	Language
20	0.12932	3896	C++14

Input	Result	Time (sec)	Memory (KiB)	Score	Your Output	Correct Output	Diff
Input #1	Accepted	0.025354	1140	10			
Input #2	Accepted	0.024833	1140	10			
Input #3	Accepted	0.017256	2	10			
Input #4	Accepted	0.017882	2	10			
Input #5	Accepted	0.02492	3896	20			
Input #6	Accepted	0.009331	2	20			
Input #7	Accepted	0.009742	2	20			

5) Good Subsequences

Code:

```
import java.util.*;
public class Main{
public static void main(String args[] ) throws Exception {
         long modulo = 1000000007;
         Scanner sc = new Scanner(System.in);
         int T = sc.nextInt();
         while(T-- !=0)
         {
              String S = sc.next();
              int N = S.length(), i;
              long res[] = new long[27];
              for( i = 0; i<N; i++)
              {
                  int a = (int)S.charAt(i) - 97;
                  res[a]++;
              }

              long ans = 1;
              for(int k = 0; k < 27; k++)
              {
                   if(res[k] > 0)
                    {
                           ans = ans*(res[k])%modulo;
```

```
                    }
                }
                System.out.println(ans%modulo);

            }
}
}
```

Output:

RESULT: Accepted — Refer judge environment

Score	Time (sec)	Memory (KiB)	Language
20	2.05618	87348	Java 8

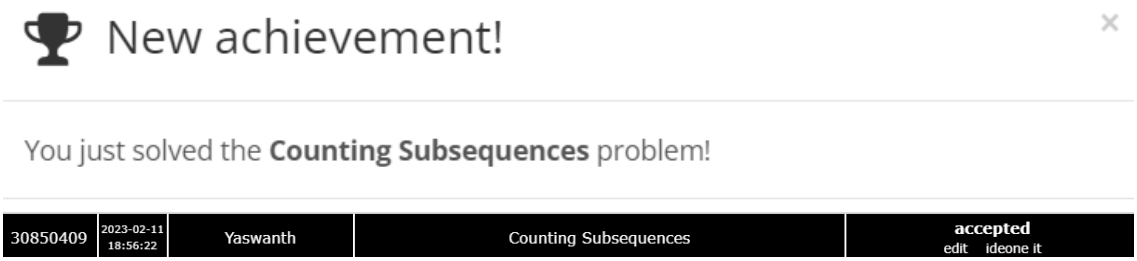
Input	Result	Time (sec)	Memory (KiB)	Score	Your Output	Correct Output	Diff
Input #1	Accepted	0.24193	84800	9			
Input #2	Accepted	0.163005	87348	8			
Input #3	Accepted	0.164115	85036	8			
Input #4	Accepted	0.163391	85164	9			
Input #5	Accepted	0.18032	85120	8			
Input #6	Accepted	0.179504	83352	8			
Input #7	Accepted	0.170613	85212	9			
Input #8	Accepted	0.187684	84860	8			
Input #9	Accepted	0.196515	83220	8			
Input #10	Accepted	0.179341	87056	9			
Input #11	Accepted	0.08261	83468	8			
Input #12	Accepted	0.147151	84892	8			

6) SUBSEQ - Counting Subsequences
Code:

```
#include <bits/stdc++.h>
using namespace std;

#define pb push_back
#define ff first
#define ss second
#define mp make_pair
#define memo(a,b) memset(a,b,sizeof(a))
#define INF 1e9
#define EPS 1e-8
#define PI 3.14159265358979323846

typedef long long ll ;
typedef unsigned long long ull ;

/* int dx[] = {1,-1,0,0} , dy[] = {0,0,1,-1}; */ // 4 Direction
/* int dx[] = {1,-1,0,0,1,1,-1,-1} , dy[] = {0,0,1,-1,1,-1,1,-1}; */ // 8 Direction
/* int dx[] = {1,-1,1,-1,2,2,-2,-2} , dy[] = {2,2,-2,-2,1,-1,1,-1}; */ // Knight Direction
/* int dx[] = {2,-2,1,1,-1,-1} , dy[] = {0,0,1,-1,1,-1}; */ // Hexagonal Direction

int main()
{
   //freopen("input.txt","r",stdin);
   //freopen("output.txt","w",stdout);
   int T ;
   scanf("%d",&T);
   while( T-- )
   {
      map <ll,ll> dp ;
      dp[0] = 1 ;

      ll N , sum = 0 , ans = 0 ;
      scanf("%lld",&N);

      for(int i=1;i<=N;i++)
      {
         int a ;
         scanf("%lld",&a);

         sum+=a;
```

```
            ans+=dp[sum-47];
            dp[sum]++;
        }
        printf("%lld\n",ans);
    }
    return 0;
}
```

Output:



7) SUMMATION – SUMMATION

Code:

```
def sm(arr, n):
ans = 0
# Finding sum of the array.
for i in range(0, n):
ans = ans + arr[i]
print(ans * pow(2, n - 1))
# Driver Code
n = int(input())
a = []
for i in range(0, n):
x = input()
tmp = list(map(int, input().split(" ")))
a.append(tmp)
for e in a:
sm(e, len(e))
exit()
```

Output: