**Data Structure and Algorithms**

**Codemonk Solved Quiz 2022**

**Rutwik Patel**

**AIMDek Trainee**

**Q.1) Monk and Inversions**

**Question Link -** https://www.hackerearth.com/problem/algorithm/monk-and-inversions-arrays-strings-e5aaa427/

**Source Code:-**

import java.util.\*;

class TestClass {

public static void main(String args[] ) throws Exception {

Scanner sc=new Scanner(System.in);

int count=0,i,j,p,q;

int cases=sc.nextInt();

while(cases!=0)

{

int n=sc.nextInt();

int arr[][]=new int[n][n];

count=0;

for(i=0; i<n; i++)

{

for(j=0; j<n; j++)

arr[i][j]=sc.nextInt();

}

for(i=0; i<n; i++)

{

for(j=0; j<n; j++)

{

for(p=i; p<n; p++)

{

for(q=j; q<n; q++)

{

if(arr[i][j]>arr[p][q])

count++;

}

}

}

}

System.out.println(count);

cases--;

}

}

}

**Q.2) Monk and Rotation**

**Question Link -** https://www.hackerearth.com/problem/algorithm/monk-and-rotation-3-bcf1aefe/

**Source Code:-**

import java.util.\*;

class TestClass {

public static void main(String args[] ) throws Exception {

Scanner sc=new Scanner(System.in);

int t=sc.nextInt();

while(t!=0)

{

int i;

int n=sc.nextInt();

int k=sc.nextInt();

int arr[]=new int[n];

for(i=0; i<n; i++)

arr[i]=sc.nextInt();

for(i=n-k; i<n; i++)

System.out.print(arr[i]+" ");

for (i=0; i<n-k; i++)

System.out.print(arr[i]+" ");

t--;

}

}

}

**Q.3) Minimum AND xor OR**

**Question Link -** https://www.hackerearth.com/practice/data-structures/arrays/1-d/practice-problems/algorithm/minimum-and-xor-or-6a05bbd4/

**Source Code:-**

import java.util.\*;

import java.util.Arrays;

class TestClass {

public static void main(String args[] ) throws Exception {

Scanner sc=new Scanner(System.in);

int cases=sc.nextInt();

for(int i=0; i<cases; i++)

{

int n=sc.nextInt();

int arr[]=new int[n];

int min=Integer.MAX\_VALUE;

for(int k=0; k<n; k++)

arr[k]=sc.nextInt();

Arrays.sort(arr);

for(int j=0; j<n-1; j++)

{

if((arr[j]^arr[(j+1)])<min)

min=(arr[j]^arr[(j+1)]);

}

System.out.println(min);

}

}

}

**Q.4) Monk and Suffix Sort**

**Question Link -** https://www.hackerearth.com/problem/algorithm/monk-and-suffix-sort-ebacdaf5/

**Source Code:-**

import java.util.\*;

class TestClass {

public static void main(String args[] ) throws Exception {

Scanner sc = new Scanner(System.in);

String s=sc.next();

int k=sc.nextInt();

List<String> list = new ArrayList<>();

for(int i=0; i<s.length(); i++)

list.add(s.substring(i));

Collections.sort(list);

System.out.println(list.get(k-1));

}

}

**Q.5) Monk and Nice Strings**

**Question Link -** https://www.hackerearth.com/problem/algorithm/monk-and-nice-strings-3-e5800d05/

**Source Code:-**

import java.util.\*;

class TestClass {

public static void main(String args[] ) throws Exception {

Scanner sc = new Scanner(System.in);

int len=sc.nextInt();

char clist[]=new char[len];

for(int i=0; i<len; i++)

clist[i]=sc.next().charAt(0);

for(int i=0; i<len; i++)

{

int count=0;

for(int j=i-1; j>=0; j--)

if(clist[j]<clist[i]) count++;

System.out.println(count);

}

}

}

**Q.6) Monk and Nice Strings**

**Question Link -** https://www.hackerearth.com/problem/algorithm/monk-being-monitor-709e0fd3/

**Source Code:-**

import java.util.\*;

class TestClass {

public static void main(String args[] ) throws Exception {

Scanner sc = new Scanner(System.in);

int cases=sc.nextInt();

int n=sc.nextInt();

HashMap<Integer,Integer> smap=new HashMap<>();

for(int i=0; i<n; i++)

{

int key=sc.nextInt();

try

{

smap.put(key,smap.get(key)+1);

}

catch(Exception e)

{

smap.put(key,1);

}

}

int diff = Collections.max(smap.entrySet(),Map.Entry.comparingByValue()).getKey()-Collections.min(smap.entrySet(),Map.Entry.comparingByValue()).getKey();

System.out.println(diff);

}

}