1)The Unlucky 13:

<https://www.hackerearth.com/submission/54645661/>

Code:

import java.util.Random;

import java.io.\*;

import java.util.Scanner;

public class Unlucky13

{

public static void main(String args[])

{

Scanner input = new Scanner(System.in);

int l = input.nextInt();

int n =input.nextInt();

int i;

for(i=1;i<=n;i++)

{

int res = generateCombo(l);

System.out.println(res);

}

}

public static int generateCombo(int l)

{

int m = (int) Math.pow(10, l - 1);

int res= m + new Random().nextInt(9 \* m);

if(res == 13)

{

generateCombo(l);

}

return res;

}

}

2) Monk and Nice Strings:

<https://www.hackerearth.com/submission/54642870/>

Code:

public class NiceStr

{

public static void main(String []args)

{

char[] arr = {'b', 'c', 'f', 'g', 'c', 'a'};

//char[] arr = {'a', 'c', 'd', 'b'};

int j = 0;

int noOfNiceStrings = 0;

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

{

if (i == 0)

{

System.out.println("String at index 0 has 0 nice strings");

}

else

{

j = i-1;

noOfNiceStrings = 0;

while (j >= 0){

if ((int)arr[i] > (int)arr[j]){

noOfNiceStrings++;

}

j--;

}

System.out.println(noOfNiceStrings);

}

}

}

}

3) Monk and Inversion:

<https://www.hackerearth.com/submission/54645559/>

Code:

import java.util.\*;

class InvertMonk {

public static void main(String[] args)

{

int p,n,i,j,k,m;

Scanner sc=new Scanner(System.in);

p=sc.nextInt();

for(int cnt=0;cnt<p;cnt++)

{

int count=0;

n=sc.nextInt();

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

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

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

if(arr[i][j]>arr[k][m])

count++;

System.out.println(count);

}

}

}

4) Monk and Rotation

<https://www.hackerearth.com/submission/54645788/>

Code:

import java.io.BufferedReader;

import java.io.InputStreamReader;

import java.util.\*;

class RotateMonk {

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

{

int n,k,i,t,po;

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

t=Integer.parseInt(br.readLine());

for(int j=0;j<t;j++)

{

String[] line=br.readLine().split(" ");

n=Integer.parseInt(line[0]);

k=Integer.parseInt(line[1]);

k=(k%n);

int a[]=new int[n];

int b[]=new int[n];

String A=br.readLine();

if(k==0)

System.out.println(A);

else

{

po=findIndx(A," ",n-k);

String str1=A.substring(po+1);

String str2=A.substring(0,po);

System.out.println(str1+" "+str2);

}

}

}

public static int findIndx(String str, String substr, int n)

{

int pos = str.indexOf(substr);

while (--n > 0 && pos != -1)

pos = str.indexOf(substr, pos + 1);

return pos;

}

}