Data Structure and Algorithms

(HackerEarth solved Quiz) 2022

Dipamkumar Donga

AIMDek Trainee

**Arrays & Strings**

**Q 1) Monk and Rotation**

<https://www.hackerearth.com/practice/codemonk/>

Java 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 n = sc.nextInt();

int k = sc.nextInt();

sc.nextLine();

String[] str = sc.nextLine().split(" ");

StringBuffer output = new StringBuffer();

k = k % n;

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

if(i<k)

output.append(str[n + i - k] + " ");

else

output.append(str[i - k] + " ");

}

System.out.println(output);

t--;

}

}

}

**Q 2) Monk and Inversions**

<https://www.hackerearth.com/practice/codemonk/>

Java source code:

import java.util.\*;

class TestClass {

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

// Write your code here

Scanner sc = new Scanner(System.in);

int t = sc.nextInt();

while(t>0){

int n = sc.nextInt();

int[][] 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 ans=0;

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

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

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

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

if(arr[i][j] > arr[p][q] && i<=p && j<=q)

ans+=1;

System.out.println(ans);

t--;

}

}

}

**Q 3) Minimum AND xor OR**

<https://www.hackerearth.com/practice/codemonk/>

Java source code:

import java.util.\*;

class TestClass {

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

// Write your code here

Scanner sc = new Scanner(System.in);

int t = sc.nextInt();

while(t > 0){

int n = sc.nextInt();

int[] arr = new int[n];

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

arr[i] = sc.nextInt();

Arrays.sort(arr);

int min = arr[n-1];

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

if(min > (arr[i] ^ arr[i-1]))

min = arr[i] ^ arr[i-1];

System.out.println(min);

t--;

}

}

}

**Q 4) Cyclic shift**

<https://www.hackerearth.com/practice/codemonk/>

Python source code:

# Write your code here

t = int(input())

while t > 0:

n,k = map(int, input().split())

s = input()

max = ""

cycle = -1

for i in range(n):

if max < s:

max = s

position = i

elif max == s:

cycle = i - position

break

s = s[1:] + s[:1]

if cycle == -1:

print(position + (k-1)\*n)

else:

print(position + (k-1)\*cycle)

t -= 1

**Sorting**

**Q 5) Monk and Nice Strings**

<https://www.hackerearth.com/practice/codemonk/>

Java source code:

import java.util.\*;

class TestClass {

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

// Write your code here

Scanner s = new Scanner(System.in);

int n = s.nextInt();

String[] str = new String[n];

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

str[i] = s.next();

int count = 0;

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

// {System.out.println(str[i] + " " + str[j] + " " + str[i].compareTo(str[j]));

if (str[i].compareTo(str[j]) > 0)

count++;

System.out.println(count);

}

}

}

**Q 6) Monk and Nice Strings**

<https://www.hackerearth.com/practice/codemonk/>

Python source code:

# Write your code here

line = input().split()

s = line[0]

k = int(line[1])

n = len(s)

arr = []

for i in range(n):

arr.append(s[i:])

arr.sort()

print(arr[k-1])

**Q 7) Monk and Sorting Algorithm**

<https://www.hackerearth.com/practice/codemonk/>

Python source code:

n = int(input())

arr = list(map(int, input().strip().split(" ")))

max\_arr = max(arr)

mul = 1

r = 10\*\*5

while max\_arr:

arr.sort(key = lambda x: (x/mul)%r)

print(' '.join(map(str, arr)))

mul \*= r

max\_arr //= r