NAME : JAYAPRIYA B

ROLL NO : 241701021

COURSE : B.E COMPUTER SCIENCE AND DESIGN

WEEK 8

Question 1
Correct
Marked out of 1.00
Flag question

Coders here is a simple task for you, you have given an array of size ***N*** and an integer ***M***.

Your task is to calculate the ***difference between maximum sum and minimum sum of N-M*** elements of the given array.

**Constraints:**

***1<=t<=10***
***1<=n<=1000***
***1<=a[i]<=1000***

**Input:**

First line contains an integer ***T*** denoting the number of testcases.
First line of every testcase contains two integer ***N*** and ***M***.
Next line contains ***N*** space separated integers denoting the elements of array

**Output:**

For every test case print your answer in new line

SAMPLE INPUT

```
1
5 1
1 2 3 4 5
```

SAMPLE OUTPUT

```
4
```

Explanation

M is 1 and N is 5 so you have to calculate maximum and minimum sum using (5-1 =) 4 elements.
Maximum sum using the 4 elements would be (2+3+4+5=)14.
Minimum sum using the 4 elements would be (1+2+3+4=)10.
Difference will be 14-10=4.

```c
#include<stdio.h>
int main()
{
    int t;
    scanf("%d",&t);
    while(t--)
    {
        int n,m,d,min,temp;
        scanf("%d %d",&n,&m);
        d=n-m;
        int arr[n];
        for(int i=0;i<n;i++)
        scanf("%d",&arr[i]);
        for(int j=0;j<n;j++)
        {
            min=j;
            for(int k=j;k<n;k++)
            {
                if(arr[k]<arr[min])
                min=k;
            }
            temp=arr[min];
            arr[min]=arr[j];
            arr[j]=temp;
        }
        int maxsum=0,minsum=0;
        for(int a=0;a<d;a++)
        minsum+=arr[a];
        for(int b=n-1;b>m-1;b--)
        maxsum+=arr[b];
        printf("%d\n",maxsum-minsum);
    }
    return 0;
}
```

| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 1<br>5 1<br>1 2 3 4 5 | 4 | 4 | ✓ |

Question **2**

Correct

Marked out of 1.00

Flag question

A new deadly virus has infected large population of a planet. A brilliant scientist has discovered a new strain of virus which can cure this disease. Vaccine produced from this virus has various strength depending on midichlorians count. A person is cured only if midichlorians count in vaccine batch is more than midichlorians count of person. A doctor receives a new set of report which contains midichlorians count of each infected patient, Practo stores all vaccine doctor has and their midichlorians count. You need to determine if doctor can save all patients with the vaccines he has. The number of vaccines and patients are equal.

**Input Format**

First line contains the number of vaccines - N. Second line contains N integers, which are strength of vaccines. Third line contains N integers, which are midichlorians count of patients.

**Output Format**

Print a single line containing '***Yes***' or '***No***'.

**Input Constraint**

***1 < N < 10***

Strength of vaccines and midichlorians count of patients fit in integer.

**SAMPLE INPUT**

5

123 146 454 542 456

100 328 248 689 200

**SAMPLE OUTPUT**

No

```c
#include<stdio.h>
int main()
{
    int n,min1,min2,temp,flag=1;
    scanf("%d",&n);
    int vac[n],pat[n];
    for(int i=0;i<n;i++)
    scanf("%d",&vac[i]);
    for(int i=0;i<n;i++)
    scanf("%d",&pat[i]);
    for(int j=0;j<n-1;j++)
    {
        min1=j,min2=j;
        for(int k=j;k<n;k++)
        {
            if(vac[k]<vac[min1])
            min1=k;
            if(vac[k]<vac[min2])
            min2=k;
        }
        temp=vac[min1];
        vac[min1]=vac[j];
        vac[j]=temp;
        temp=pat[min2];
        pat[min2]=pat[j];
        pat[j]=temp;
    }
    for(int i=0;i<n;i++)
    {
        if(vac[i]<=pat[i])
        {
            flag=0;
            break;
        }
    }
    if(flag==1)
    printf("Yes");
    else
    printf("No");
}
```

| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 5<br>123 146 454 542 456<br>100 328 248 689 200 | No | No | ✓ |

Passed all tests! ✓

Question **3**

Correct

Marked out of 1.00

Flag question

You are given an array of n integer numbers $a_1, a_2, \ldots, a_n$. Calculate the number of pair of indices $(i, j)$ such that $1 \le i < j \le n$ and $a_i$ xor $a_j = 0$.

**Input format**

- First line: $n$ denoting the number of array elements
- Second line: n space separated integers $a_1, a_2, \ldots, a_n$.

**Output format**

Output the required number of pairs.

**Constraints**

$1 \le n \le 10^6$

$1 \le a_i \le 10^9$

**SAMPLE INPUT**

5

1 3 1 4 3

**SAMPLE OUTPUT**

2

Explanation

The 2 pair of indices are $(1, 3)$ and $(2,5)$.

```c
#include<stdio.h>
int main(){
int n,count=0;
scanf("%d",&n);
int arr[n];
for(int i=0;i<n;i++)
scanf("%d",&arr[i]);
for(int i=0;i<n-1;i++)
{
    for(int j=i+1;j<n;j++)
    {
        if((arr[i]^arr[j])==0)
        count++;
    }
}
printf("%d",count);
}
```

| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 5<br>1 3 1 4 3 | 2 | 2 | ✓ |

Passed all tests! ✓

Question **4**

Correct

Marked out of 1.00

Flag question

You are given an array ***A*** of non-negative integers of size ***m***. Your task is to sort the array in non-decreasing order and print out the original indices of the new sorted array.

**Example:**

A={4,5,3,7,1}

After sorting the new array becomes A={1,3,4,5,7}.

The required output should be "4 2 0 1 3"

**INPUT :**

The first line of input consists of the size of the array
The next line consists of the array of size m

**OUTPUT :**

Output consists of a single line of integers

**CONSTRAINTS:**

***1<=m<=106***
***0<=A[i]<=106***

NOTE: The indexing of the array starts with 0.

**SAMPLE INPUT**

5
4 5 3 7 1

**SAMPLE OUTPUT**

4 2 0 1 3

```c
#include<stdio.h>
int main()
{
    int n;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++)
    scanf("%d",&arr[i]);
    int max=arr[0];
    for(int i=1;i<n;i++)
    {
        if(arr[i]>max)
        max=arr[i];
    }
    max++;
    int min=0;
    for(int a=0;a<n;a++)
    {
        for(int b=0;b<n;b++)
        {
            if(arr[b]<arr[min])
            min=b;
        }
        printf("%d ",min);
        arr[min]=max;
    }
}
```

| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 5<br>4 5 3 7 1 | 4 2 0 1 3 | 4 2 0 1 3 | ✓ |

Passed all tests! ✓