

Programming using C

week08 practice session and coding

Name:**S.Jananejaipratha**

Department:**AIML-'B'**

Roll No.:**241501079**

Source: 8
Editor: 1
Editor: 2
Editor: 3
Editor: 4
Editor: 5
Editor: 6
Editor: 7
Editor: 8
Editor: 9
Editor: 10
Editor: 11
Editor: 12
Editor: 13
Editor: 14
Editor: 15
Editor: 16
Editor: 17
Editor: 18
Editor: 19
Editor: 20
Editor: 21
Editor: 22
Editor: 23
Editor: 24
Editor: 25
Editor: 26
Editor: 27
Editor: 28
Editor: 29
Editor: 30
Editor: 31
Editor: 32
Editor: 33
Editor: 34
Editor: 35
Editor: 36
Editor: 37
Editor: 38
Editor: 39
Editor: 40
Editor: 41
Editor: 42
Editor: 43
Editor: 44
Editor: 45
Editor: 46
Editor: 47
Editor: 48
Editor: 49
Editor: 50
Editor: 51
Editor: 52
Editor: 53
Editor: 54
Editor: 55
Editor: 56
Editor: 57
Editor: 58
Editor: 59
Editor: 60
Editor: 61
Editor: 62
Editor: 63
Editor: 64
Editor: 65
Editor: 66
Editor: 67
Editor: 68
Editor: 69
Editor: 70
Editor: 71
Editor: 72
Editor: 73
Editor: 74
Editor: 75
Editor: 76
Editor: 77
Editor: 78
Editor: 79
Editor: 80
Editor: 81
Editor: 82
Editor: 83
Editor: 84
Editor: 85
Editor: 86
Editor: 87
Editor: 88
Editor: 89
Editor: 90
Editor: 91
Editor: 92
Editor: 93
Editor: 94
Editor: 95
Editor: 96
Editor: 97
Editor: 98
Editor: 99
Editor: 100

Given two integers M and N , you have given a set of M and N elements.

Your task is to calculate the difference between maximum sum and minimum sum of M elements of the given set.

Constraints:

$1 \leq M \leq 100$
 $1 \leq N \leq 1000$
 $1 \leq A[i] \leq 1000$

Input:

First line contains an integer T denoting the number of test cases.
Next line of every test case contains two integers M and N .
Then line contains M space separated integers denoting the elements of array.

Output:

For every test case print your answer in a new line.

Sample Input:

1
5 7
1 2 3 4 5

Sample Output:

4

Explanation:

With $T=1$ and $N=7$, you have to calculate maximum and minimum sum using 5 to 7 elements.

Maximum sum using the 5 elements would be $2+3+4+5+6=20$.

Minimum sum using the 5 elements would be $1+2+3+4+5=15$.

Difference will be $20-15=5$.

Answer: (Initially, output 2 %)

```
1 // C++ program to find the difference between maximum and minimum sum of M elements of the given set.
2 #include <iostream>
3 using namespace std;
4 int main()
5 {
6     int T;
7     cin >> T;
8     while (T--)
9     {
10         int M, N;
11         cin >> M >> N;
12         int arr[M];
13         for (int i = 0; i < M; i++)
14             cin >> arr[i];
15         sort(arr, arr + M);
16         int sum1 = 0, sum2 = 0;
17         for (int i = 0; i < M; i++)
18             sum1 += arr[i];
19         for (int i = M - 1; i >= 0; i--)
20             sum2 += arr[i];
21         cout << sum2 - sum1 << endl;
22     }
23 }
```

Input	Expected	Got
1 5 7 1 2 3 4 5	4	5

Passed all tests. ✓

Activate Windows

2

Points

Maximum of 100

0 flag awarded

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 resistance count. A person is cured only if resistance count in vaccine batch is more than resistance count of person. A doctor receives a new set of report which contains resistance count of each infected patient. Doctor stores all vaccine doctor has and their resistance count. You need to determine if doctor can cure all patients with the vaccine 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 vaccine. Third line contains N integers, which are resistance count of patients.

Output Format

Print a single line containing "Yes" or "No".

Input Constraints

$1 \leq N \leq 10$

Strength of vaccine and resistance count of patients fit in integer.

SAMPLE INPUT

```
5
123 145 454 542 456
100 124 245 689 200
```

SAMPLE OUTPUT

No

Answer: (generally requires 0 %)

```
1 #include<iostream.h>
2 int main()
3 {
4     int n,start,end,temp,flag=1;
5     scanf("%d",&n);
6     int arr[n],pat[n];
7     for(int i=0;i<n;i++)
8         scanf("%d",&arr[i]);
9     for(int j=0;j<n;j++)
10    {
11        start=j,end=j;
12        for(int k=j+1;k<n;k++)
13        {
14            if(arr[k]<arr[start])
15                start=k;
16            if(pat[k]>pat[end])
17                end=k;
18        }
19        temp=arr[start];
20        arr[start]=arr[end];
21        arr[end]=temp;
22        temp=pat[start];
23        pat[start]=pat[end];
24        pat[end]=temp;
25    }
26    for(int i=0;i<n;i++)
27    {
28        if(arr[i]>pat[i])
29        {
30            flag=0;
31            break;
32        }
33    }
34    if(flag==1)
35        printf("Yes");
36    else
37        printf("No");
38    return 0;
39 }
```

	Input	Expected	Got
✓	5 123 145 454 542 456 100 124 245 689 200	No	No ✓

Activate Windows
Go to Settings to activate Windows.

Question 3

Answer

Model used of 17%

Flag question

You are given an array of n integer numbers a_1, a_2, \dots, a_n . Calculate the number of pair of indices i, j such that $1 \leq i < j \leq n$ and $a_i \text{ xor } a_j = 0$.

Input format

- First line is denoting the number of array elements
- Second line is space separated integers a_1, a_2, \dots, a_n

Output format

Output the required number of pair.

Constraints

- $1 \leq n \leq 10^6$
- $1 \leq a_i \leq 10^9$

SAMPLE INPUT

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

SAMPLE OUTPUT

```
2
```

Explanation

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

Answer: (possibly integer 0 %)

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5     int n, count=0;
6     int arr[1000000];
7     for(int i=0; i<n; i++)
8     {
9         cin>>arr[i];
10    }
11    for(int i=0; i<n; i++)
12    {
13        for(int j=i+1; j<n; j++)
14        {
15            if((arr[i]^arr[j])==0)
16                count++;
17        }
18    }
19    cout<<count<<endl;
20    return 0;
21 }
```

	Input	Expected	Got	
✓	5 1 3 1 4 3	2	2	✓

Passed all test ✓

4

Points

Made out of 100

0 / 100

0 / 100

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

Example:

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

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

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 **n**.

OUTPUT :

Output consists of a single line of integers.

CONSTRAINTS:

1 ≤ n ≤ 1000

0 ≤ A[i] ≤ 1000

NOTE: The indexing of the array starts with 0.

SAMPLE INPUT

5
4 5 2 7 1

SAMPLE OUTPUT

4 2 0 1 3

Answer: (partialy right: 0 %)

```
1 //Bubble Sort
2 int main()
3 {
4     int A;
5     scanf("%d",&A);
6     int arr[A];
7     for(int i=0;i<A;i++)
8     {
9         scanf("%d",&arr[i]);
10    }
11    for(int i=0;i<A;i++)
12    {
13        for(int j=i+1;j<A;j++)
14        {
15            if(arr[i]>arr[j])
16            {
17                int temp=arr[i];
18                arr[i]=arr[j];
19                arr[j]=temp;
20            }
21        }
22    }
23    for(int i=0;i<A;i++)
24    {
25        printf("%d ",arr[i]);
26    }
27    return 0;
28 }
```

	Input	Expected	Got	
✓	5	4 2 0 1 3	4 2 0 1 3	✓
	4 5 2 7 1			

Passed all tests. ✓

Activate Windows
Go to Settings to activate Windows.