**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

**Answer:** (penalty regime: 0 %)

```c
#include<stdio.h>
void bubblesort(int arr[],int n){
    int temp,i,j;
    for(i=0;i<n;i++){
        for(j=0;j<n-1-i;j++){
            if(arr[j]>arr[j+1]){
                temp=arr[j];
                arr[j]=arr[j+1];
                arr[j+1]=temp;
            }
        }
    }
}
int main(){
    int N;
    scanf("%d",&N);
    int vaccines[N];
    int patients[N];
    for(int i=0;i<N;i++){
        scanf("%d",&vaccines[i]);
    }
    for(int i=0;i<N;i++){
        scanf("%d",&patients[i]);
    }
    bubblesort(vaccines,N);
    bubblesort(patients,N);
    for(int i=0;i<N;i++){
        if(vaccines[i]<=patients[i]){
            printf("No\n");
            return 0;
```

*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

**Answer:** (penalty regime: 0 %)

```c
#include<stdio.h>
void bubblesort(int arr[],int n){
    int temp,i,j;
    for(i=0;i<n;i++){
        for(j=0;j<n-1-i;j++){
            if(arr[j]>arr[j+1]){
                temp=arr[j];
                arr[j]=arr[j+1];
                arr[j+1]=temp;
            }
        }
    }
}
int main(){
    int N;
    scanf("%d",&N);
    int vaccines[N];
    int patients[N];
    for(int i=0;i<N;i++){
        scanf("%d",&vaccines[i]);
    }
    for(int i=0;i<N;i++){
        scanf("%d",&patients[i]);
    }
    bubblesort(vaccines,N);
    bubblesort(patients,N);
    for(int i=0;i<N;i++){
        if(vaccines[i]<=patients[i]){
            printf("No\n");
            return 0;
        }
    }
    printf("Yes\n");
    return 0;
}
```

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

Passed all tests! ✓

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 $(2,5)$.

**Answer:** (penalty regime: 0 %)

```
#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 count =0;
    for(int i=0;i<n;i++){
        for(int j=i+1;j<n;j++){
            if((arr[i]^arr[j])==0) count
        }
    }
    printf("%d",count);
    return 0;
}
```

$1 \le n \le 10^6$

$1 \le a_i \le 10^9$

**SAMPLE INPUT**

5

1 3 1 4 3

**SAMPLE OUTPUT**

2

Explanation

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

**Answer:** (penalty regime: 0 %)

```
#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 count =0;
    for(int i=0;i<n;i++){
        for(int j=i+1;j<n;j++){
            if((arr[i]^arr[j])==0) count
        }
    }
    printf("%d",count);
    return 0;
}
```

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

Passed all tests! ✓

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.

Passed all tests! ✔

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

**Answer:** (penalty regime: 0 %)

```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];
```

**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

**Answer:** (penalty regime: 0 %)

```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 i=0;i<n;i++){
        for(int j=0;j<n;j++){
            if(arr[j]<arr[min]) min=j;
        }
        printf("%d ",min);
        arr[min]=max;
    }
    return 0;
}
```

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

Passed all tests! ✓

Finish review