# CSC 225 FALL 2022
# ALGORITHMS AND DATA STRUCTURES I
# ASSIGNMENT 3 - PROGRAMMING
# UNIVERSITY OF VICTORIA

Alice and Bob each have an array of integers, namely $\boldsymbol{A}$ and $\boldsymbol{B}$. $\boldsymbol{A}$ and $\boldsymbol{B}$ have the same size, say $\boldsymbol{n}$. Alice and Bob have come up with a new definition called **"weird equality"**: two arrays $\boldsymbol{A}$ and $\boldsymbol{B}$ are **weirdly equal** to each other if **at least** one of the following four conditions is met:

1) $\boldsymbol{A} = \boldsymbol{B}$, meaning that the elements in the corresponding indices are the same.

If $\boldsymbol{n}$ is divisible by 2, we divide each of the arrays into 2 parts of equal size. So, $\boldsymbol{A}$ will be divided into $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ (from left to right). Similarly, $\boldsymbol{B}$ is divided into $\boldsymbol{B_1}$ and $\boldsymbol{B_2}$. Now, the following conditions are checked and if any of them are satisfied we can still say that the arrays are **weirdly equal** to each other.

2) ($\boldsymbol{A_1}$ is weird equal to $\boldsymbol{B_1}$) AND ($\boldsymbol{A_2}$ is weird equal to$\boldsymbol{B_2}$)

3) ($\boldsymbol{A_1}$ is weird equal to $\boldsymbol{B_1}$) AND ($\boldsymbol{A_1}$ is weird equal to $\boldsymbol{B_2}$)

4) ($\boldsymbol{A_2}$ is weird equal to $\boldsymbol{B_2}$) AND ($\boldsymbol{A_2}$ is weird equal to $\boldsymbol{B_1}$)

**Note:** if $\boldsymbol{n}$ is not divisible by 2, then conditions 2, 3, and 4 are **NOT** satisfied.

At the moment, Alice and Bob are thinking day and night to determine whether their arrays satisfy the weird equality or not. Write a recursive program for them to determine this and end their pain!

**Input Format**

The first line has an integer $\boldsymbol{1 \leq n \leq 10^4}$. The second line has the $\boldsymbol{n}$ integers, $\boldsymbol{a_1, a_2, \ldots, a_{n-1}}$, of array $\boldsymbol{A}$, and the third line has the $\boldsymbol{n}$ integers, $\boldsymbol{b_1, b_2, \ldots, b_{n-1}}$, of array $\boldsymbol{B}$, where $\boldsymbol{0 \leq a_i, b_i \leq 10^8}$, for each $\boldsymbol{0 \leq i \leq n-1}$.

**Output Format**

On one line print "YES" if the arrays are weird equal to each other, and "NO" if they are not. This output is case-sensitive and the quotes are just for clarity.

**Sample Input 0**

```
5
10 11 8 19 11
10 11 8 19 11
```

**Sample Output 0**

```
YES
```

**Explanation 0**

In this sample, condition 1 is satisfied since the two arrays are the same.

**Sample Input 1**

```
8
10 2 8 9 3 7 4 1
10 2 8 9 4 1 4 1
```

**Sample Output 1**

```
YES
```

**Explanation 1**

In this sample, condition 2 is satisfied in a recursive manner. In fact, A1 and B1 are the same and so they are weird equal, as well. Moreover, A2 and B2 are weird equal to each other recursively, by condition 4.

**Sample Input 2**

```
8
10 2 8 9 3 7 4 1
10 10 10 2 3 7 4 1
```

**Sample Output 2**

```
YES
```

**Explanation 2**

Similar to the previous sample, the two arrays are weirdly equal, recursively.

**Sample Input 3**

```
6
10 2 8 9 3 7
10 10 10 9 3 7
```

**Sample Output 3**

```
NO
```

**Explanation 3**

The arrays are not exactly the same so condition 1 is not satisfied. To check other conditions, we divide the array in half but none of the conditions are satisfied even recursively.

### Submission

You must solve this problem using Java. You will read from standard input and print to standard output. You must use one of the template provided, `Solution.java`.

### Evaluation Criteria

The programming assignment will be marked out of 20, based on a combination of automated testing and human inspection. The following score ranges will apply to this assignment:

| Score | Description |
|---|---|
| 0 – 5 | Submission does not compile. |
| 5 – 10 | Compiles but incorrectly reports YES or NO. |
| 10 – 15 | Correctly reports the solution but does inefficiently. That is, the runtime is $T(n) \notin O(n^2)$ time. |
| 15 – 20 | Correctly and efficiently reports the solution. That is, the runtime is $T(n) \in O(n^2)$ time. |