We define the distance between two array values as the number of indices between the two values. Given a, find the minimum distance between any pair of equal elements in the array. If no such value exists, print -1.

For example, if $a = [3, 2, 1, 2, 3]$, there are two matching pairs of values: 3 and 2. The indices of the 3's are $i = 0$ and $j = 4$, so their distance is $d[i, j] = |j - i| = 4$. The indices of the 2's are $i = 1$ and $j = 3$, so their distance is $d[i, j] = |j - i| = 2$.

Function Description

Complete the *minimumDistances* function in the editor below. It should return the minimum distance between any two matching elements.

minimumDistances has the following parameter(s):

- *a*: an array of integers

Input Format

The first line contains an integer n, the size of array a.
The second line contains n space-separated integers $a[i]$.

Constraints

- $1 \leq n \leq 10^3$
- $1 \leq a[i] \leq 10^5$

Output Format

Print a single integer denoting the minimum $d[i, j]$ in a. If no such value exists, print -1.

Sample Input

```
6
7 1 3 4 1 7
```

Sample Output

```
3
```

Explanation
Here, we have two options:

- $a[1]$ and $a[4]$ are both 1, so $d[1, 4] = |1 - 4| = 3$.
- $a[0]$ and $a[5]$ are both 7, so $d[0, 5] = |0 - 5| = 5$.

The answer is $min(3, 5) = 3$.