HackerRank Sample Test 09m : 22s to test end

C

☆ Find the number!

Given an unsorted array of n elements, find if the element k is present in the array or not.
Complete the *findNumber* function in the editor below. It has 2 parameters:

1. An array of integers, arr, denoting the elements in the array.
2. An integer, k, denoting the element to be searched in the array.

The function must return a string "YES" or "NO" denoting if the element is present in the array or not.

Input Format

The first line contains an integer n, denoting the number of elements in the array arr.
Each line *i* of the *n* subsequent lines (where $0 \leq i < n$) contains an integer describing arr_i.
The next line contains an integer, k, the element that needs to be searched.

Constraints

- $1 \leq n \leq 10^5$
- $1 \leq arr[i] \leq 10^9$

Output Format

The function must return a string "YES" or "NO" denoting if the element is present in the array or not. This is printed to stdout by locked stub code in the editor.

Sample Input 0

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

Sample Output 0

```
YES
```

Explanation 0

HackerRank Sample Test 09m : 22s to test end

Sample Input 1

```
3
2
3
1
5
```

Sample Output 1

```
NO
```

Explanation 1

Given the array [2, 3, 1] and k = 5. There is no element 5 in the array and therefore we print "NO".

YOUR ANSWER

We recommend you take a quick tour of our editor before you proceed. The timer will pause up to 90 seconds for the tour. Start tour

Draft saved 04:21 pm Original code Python 2

```
#!/bin/python

import sys
import os


# Complete the function below.

def findNumber(arr, k):
    if k in arr:
        return "YES"
    return "NO"
```

HackerRank Sample Test 09m : 22s to test end

```
14  f = open(os.environ['OUTPUT_PATH'], 'w')
15
16
17  _arr_cnt = 0
18  _arr_cnt = int(raw_input())
19  _arr_i=0
20  _arr = []
21  while _arr_i < _arr_cnt:
22      _arr_item = int(raw_input());
23      _arr.append(_arr_item)
24      _arr_i+=1
25
26
27
28  _k = int(raw_input());
29
30  res = findNumber(_arr, _k)
31  f.write(res + "\n")
32
33  f.close()
34
```

Line: 16 Col: 1

1

2

3

Test against custom input

Run Code | Submit code & Continue

(You can submit any number of times)

Download sample test cases *The input/output files have Unix line endings. Do not use Notepad to edit them on windows.*

Compiled successfully. All available test cases passed!

Tip: Debug your code against custom input

Test Case #1: ✔
Test Case #2: ✔
Test Case #3: ✔
Test Case #4: ✔
Test Case #5: ✔
Test Case #6: ✔

HackerRank Sample Test 09m : 22s to test end

Test Case #10: ✔
Test Case #11: ✔
Test Case #12: ✔
Test Case #13: ✔

Testcase 1: *Success*

Input [Download]

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

Your Output

```
YES
```

Expected Output [Download]

```
YES
```

Testcase 2: *Success*

Input [Download]

```
3
2
3
1
5
```

Your Output

```
NO
```

Expected Output [Download]

```
NO
```

Testcase 3: *Success*

Your Output

```
Output hidden
```

Your Output

```
Output hidden
```

Testcase 5: *Success*

Your Output

```
Output hidden
```

Testcase 6: *Success*

Your Output

```
Output hidden
```

Testcase 7: *Success*

Your Output

```
Output hidden
```

Testcase 8: *Success*

Your Output

```
Output hidden
```

Testcase 9: *Success*

Your Output

```
Output hidden
```

Testcase 10: *Success*

Your Output

```
Output hidden
```

Testcase 11: *Success*

Your Output

```
Output hidden
```

Your Output

```
Output hidden
```

Testcase 13: *Success*

Your Output

```
Output hidden
```