Picking Numbers

84.8 more points to get your gold badge!

Rank: 288788 | Points: 765.2/850

Your Picking Numbers submission got 20.00 points. Share Post

You are now 84.8 points away from the gold level for your problem solving badge.

Try the next challenge | Try a Random Challenge

Problem Submissions Leaderboard Editorial

Given an array of integers, find the longest subarray where the absolute difference between any two elements is less than or equal to $\mathbf{1}$.

Example

$a = [1, 1, 2, 2, 4, 4, 5, 5, 5]$

There are two subarrays meeting the criterion: $[1, 1, 2, 2]$ and $[4, 4, 5, 5, 5]$. The maximum length subarray has $\mathbf{5}$ elements.

Function Description

Complete the pickingNumbers function in the editor below.

pickingNumbers has the following parameter(s):

- int a[n]: an array of integers

Returns

- int: the length of the longest subarray that meets the criterion

Input Format

The first line contains a single integer n, the size of the array a.

The second line contains n space-separated integers, each an $a[i]$.

Constraints

- $2 \leq n \leq 100$
- $0 < a[i] < 100$
- The answer will be ≥ 2.

Sample Input 0

```
6
4 6 5 3 3 1
```

Sample Output 0

```
3
```

Explanation 0

We choose the following multiset of integers from the array: $\{4, 3, 3\}$. Each pair in the multiset has an absolute difference ≤ 1 (i.e., $|4 - 3| = 1$ and $|3 - 3| = 0$), so we print the number of chosen integers, 3, as our answer.

Sample Input 1

```
6
1 2 2 3 1 2
```

Sample Output 1

```
5
```

Explanation 1

We choose the following multiset of integers from the array: $\{1, 2, 2, 1, 2\}$. Each pair in the multiset has an absolute difference ≤ 1 (i.e., $|1-2| = 1$, $|1-1| = 0$, and $|2-2| = 0$), so we print the number of chosen integers, 5, as our answer.

Change Theme Language Python 3

```
#
# Complete the 'pickingNumbers' function below.
#
# The function is expected to return an INTEGER.
# The function accepts INTEGER_ARRAY a as parameter.
#

def pickingNumbers(a):
    # Write your code here
    a.sort()
    answer = 0
    for start in range(len(a) - 1):
        for end in range(start + 1, len(a)):
            if abs(a[end] - a[start]) <= 1:
                if end - start + 1 > answer:
                    answer = end - start + 1
            else:
                break
    return answer

if __name__ == '__main__':
    fptr = open(os.environ['OUTPUT_PATH'], 'w')

    n = int(input().strip())

    a = list(map(int, input().rstrip().split()))

    result = pickingNumbers(a)

    fptr.write(str(result) + '\n')

    fptr.close()

```

EMACS Line: 27 Col: 18

Upload Code as File | Test against custom input | Run Code | Submit Code

You have earned 20.00 points!

You are now 84.8 points away from the gold level for your problem solving badge.

77% 765.2/850

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

- Test case 0
- Test case 1
- Test case 2
- Test case 3
- Test case 4
- Test case 5
- Test case 6

Compiler Message

```
Success
```

Input (stdin) Download

```
6
4 6 5 3 3 1
```

Expected Output Download

```
3
```