Sherlock and MiniMax

HackerRank

Watson gives Sherlock an array of integers. Given the endpoints of an integer range, for all M in that inclusive range, determine the minimum(abs(arr[i]-M) for all $1 \leq i \leq |arr|$)). Once that has been determined for all integers in the range, return the M which generated the maximum of those values. If there are multiple M's that result in that value, return the lowest one.

For example, your array $arr = [3, 5, 7, 9]$ and your range is from $p = 6$ to $q = 8$ inclusive.

```
M       |arr[1]-M|      |arr[2]-M|      |arr[3]-M|      |arr[4]-M|      Min
6           3               1               1               3            1
7           4               2               0               2            0
8           5               3               1               1            1
```

We look at the `Min` column and see the maximum of those three values is 1. Two M's result in that answer so we choose the lower value, 6.

Function Description

Complete the *sherlockAndMinimax* function in the editor below. It should return an integer as described.

sherlockAndMinimax has the following parameters:
- *arr*: an array of integers
- *p*: an integer that represents the lowest value of the range for M
- *q*: an integer that represents the highest value of the range for M

Input Format

The first line contains an integer n, the number of elements in arr.
The next line contains n space-separated integers $arr[i]$.
The third line contains two space-separated integers p and q, the inclusive endpoints for the range of M.

Constraints

$1 \leq n \leq 10^2$
$1 \leq arr[i] \leq 10^9$
$1 \leq p \leq q \leq 10^9$

Output Format

Print the value of M on a line.

Sample Input

```
3
5 8 14
4 9
```

Sample Output

```
4
```

Explanation

$arr = [5, 8, 14], \text{range} = [4 - 9]$

```
M       |arr[1]-M|      |arr[2]-M|      |arr[3]-M|      Min
4           1               4              10            1
5           0               3               9            0
6           1               2               8            1
7           2               1               7            1
8           3               0               6            0
9           4               1               5            1
```

For $M = 4, 6, 7$, or 9, the result is 1. Since we have to output the smallest of the multiple solutions, we print 4.