Max Min

You will be given a list of integers, arr, and a single integer k. You must create an array of length k from elements of arr such that its unfairness is minimized. Call that array arr'. Unfairness of an array is calculated as

$$max(arr') - min(arr')$$

Where:

- max denotes the largest integer in arr'.
- min denotes the smallest integer in arr'.

Example

$arr = [1, 4, 7, 2]$

$k = 2$

Pick any two elements, say $arr' = [4, 7]$.

$unfairness = max(4, 7) - min(4, 7) = 7 - 4 = 3$

Testing for all pairs, the solution $[1, 2]$ provides the minimum unfairness.

Note: Integers in arr may not be unique.

Function Description

Complete the maxMin function in the editor below.

maxMin has the following parameter(s):

- int k: the number of elements to select
- int arr[n]:: an array of integers

Returns

- int: the minimum possible unfairness

Input Format

The first line contains an integer n, the number of elements in array arr.

The second line contains an integer k.

Each of the next n lines contains an integer $arr[i]$ where $0 \leq i < n$.

Constraints

$2 \leq n \leq 10^5$

$2 \leq k \leq n$

$0 \leq arr[i] \leq 10^9$

Sample Input

Sample Input #01

```
10
4
1
2
3
4
10
20
30
40
100
200
```

Sample Output

Author amititkgp
Difficulty Medium
Max Score 100
Submitted By 5810

NEED HELP?

View discussions
View editorial
View top submissions

RATE THIS CHALLENGE

MORE DETAILS

Download problem statement
Download sample test cases
Suggest Edits

Sample Output #01

```
3
```

Explanation

Explanation #01

Here $K = 4$; selecting the 4 integers $1, 2, 3, 4$, unfairness equals

```
max(1,2,3,4) - min(1,2,3,4) = 4 - 1 = 3
```

Change Theme Language Rust

```rust
use std::env;
use std::fs::File;
use std::io::{self, BufRead, Write};

/*
 * Complete the 'maxMin' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts following parameters:
 *  1. INTEGER k
 *  2. INTEGER_ARRAY arr
 */

fn maxMin(k: i32, arr: &[i32]) -> i32 {

}

fn main() {
    let stdin = io::stdin();
    let mut stdin_iterator = stdin.lock().lines();

    let mut fptr = File::create(env::var("OUTPUT_PATH").unwrap()).unwrap();

    let n = stdin_iterator.next().unwrap().unwrap().trim().parse::<i32>().unwrap();

    let k = stdin_iterator.next().unwrap().unwrap().trim().parse::<i32>().unwrap();

    let mut arr: Vec<i32> = Vec::with_capacity(n as usize);

    for _ in 0..n {
        let arr_item = stdin_iterator.next().unwrap().unwrap().trim().parse::<i32>().unwrap();
        arr.push(arr_item);
    }

    let result = maxMin(k, &arr);

    writeln!(&mut fptr, "{}", result).ok();
}
```

Line: 39 Col: 1

Upload Code as File

Test against custom input

Run Code

Submit Code