River Records

Given an array of integers, without reordering, determine the maximum difference between any element and any prior smaller difference.  If there is never a lower prior element, return -1.

****Example****

*arr = [5, 3, 6, 7, 4]*

There are no earlier elements than *arr[0]*.

There is no earlier reading with a value lower than *arr[1].*

There are two lower earlier readings with a value lower than *arr[2] = 6:*

* *arr[2] - arr[1] = 6 - 3 = 3*
* *arr[2] - arr[0] = 6 - 5 = 1*

There are three lower earlier readings with a lower value than *arr[3] = 7:*

* *arr[3] - arr[2] = 7 - 6 = 1*
* *arr[3] - arr[1] = 7 - 3 = 4*
* *arr[3] - arr[0] = 7 - 5 = 2*

There is one lower earlier reading with a lower value than *arr[4] = 4:*

* *arr[4] - arr[1] = 4 - 3 = 1*

The maximum trailing record is *arr[3] - arr[1] = 4.*

****Example****

*arr = [4, 3, 2, 1]*

No item in *arr*has a lower earlier reading, therefore return*-1*

****Function Description****

Complete the function *maximumTrailing* in the editor below.

*maximumTrailing* has the following parameter(s):

*int* *arr[n]:*  an array of integers

****Returns:****

*int:* the maximum trailing difference*,* or *-1* if no element in *arr* has a lower earlier value

**Constraints**

* *1 ≤ n ≤ 2 × 105*
* *−106 ≤ arr[i] ≤ 106*and  0*≤ i < n*

Input Format For Custom Testing

Input from stdin will be processed as follows and passed to the function:

The first line contains a single integer, *n*, the number of elements in the array *arr*.

Each of the *n* subsequent lines contains a single integer, each an element *arr[i]* where *0 ≤ i < n*.

Sample Case 0

**Sample Input 0**

STDIN         Function****-----         --------****7****→**** arr[] size n = 7

2****→****arr = [2, 3, 10, 2, 4, 8, 1]

3

10

2

4

8

1

**Sample Output**

8

**Explanation**

Differences are calculated as:

* *3 - [2] = [1]*
* *10 - [3, 2] = [7, 8]*
* *4 - [2, 3, 2] = [2, 1, 2]*
* *8 - [4, 2, 3, 2] = [4, 6, 5, 6]*

The maximum trailing difference is *10 - 2 = 8.*

Sample Case 1

**Sample Input 1**

STDIN         Function ****-----         --------**** 6****→**** arr[] size n = 6

7****→****arr = [7, 9, 5, 6, 3, 2]

9

5

6

3

2

**Sample Output**

2

**Explanation**

Differences are calculated as:

* *9 - [7] = 2*
* *6 - [5] = 1*

The maximum trailing difference is *2.*

import java.io.\*;

import java.math.\*;

import java.security.\*;

import java.text.\*;

import java.util.\*;

import java.util.concurrent.\*;

import java.util.function.\*;

import java.util.regex.\*;

import java.util.stream.\*;

import static java.util.stream.Collectors.joining;

import static java.util.stream.Collectors.toList;

class Result {

/\*

\* Complete the 'maxTrailing' function below.

\*

\* The function is expected to return an INTEGER.

\* The function accepts INTEGER\_ARRAY arr as parameter.

\*/

public static int maxTrailing(List<Integer> arr) {

// Write your code here

}

}

public class Solution {

public static void main(String[] args) throws IOException {

BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(System.in));

BufferedWriter bufferedWriter = new BufferedWriter(new FileWriter(System.getenv("OUTPUT\_PATH")));

int arrCount = Integer.parseInt(bufferedReader.readLine().trim());

List<Integer> arr = IntStream.range(0, arrCount).mapToObj(i -> {

try {

return bufferedReader.readLine().replaceAll("\\s+$", "");

} catch (IOException ex) {

throw new RuntimeException(ex);

}

})

.map(String::trim)

.map(Integer::parseInt)

.collect(toList());

int result = Result.maxTrailing(arr);

bufferedWriter.write(String.valueOf(result));

bufferedWriter.newLine();

bufferedReader.close();

bufferedWriter.close();

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*