Jutge.org

The Virtual Learning Environment for Computer Programming

The height of a mountain's summit X33007_en

You must implement a function which receives a vector v of natural numbers with two parts. In the first, elements have strictly increasing values, and in the second, they have strictly decreasing values. More formally, there is a valid index i in v such that, for all j before i the condition $v[j] < v[j+1]$ is true, and for all j after i the condition $v[j-1] > v[j]$ is true. It is guaranteed that the vector v has at least three elements, and that the maximum (i.e., $v[i]$ for the i mentioned previously) does not occur exactly at the beginning or exactly at the end of the vector.

The function must return the maximum value of v. This is the header:

```
// Pre:  Let n be v.size(). Then n >= 3 and for all i in {0..n-1}, v[i] >= 0.
//       Also, there exists i in {1..n-2} such that v[0..i] is strictly increas
//                                                  v[i..n-1] is strictly decre
// Post: Returns the maximum value of v.
int summitOfMountain(const vector<int> &v);
```

Observation

You only need to submit the required procedure; your main program will be ignored.

Observation

Evaluation over 10 points:

- Slow solution: 5 points.
- Fast solution: 10 points.

We define the fast solution to be one that is correct, of logarithmic cost and passes both the public and private test cases. We define a slow solution to be one that is not fast, but it is correct and passes the public test cases.

Problem information

Author: PRO1

Generation: 2026-01-25T15:13:49.843Z