

## Problem Statement

Fox Ciel is the owner of a ski resort. The ski resort has  $N$  places numbered 0 through  $N-1$ . You are given a `int[] altitude`. For each  $i$ , the  $i$ -th element of `altitude` is the altitude of the place  $i$ .

The skiers would like to follow the path (place 0)  $\rightarrow$  (place 1)  $\rightarrow$  ...  $\rightarrow$  (place  $N-1$ ). The trip will only be possible if the altitudes of the places are non-increasing. In order to make the trip possible, Ciel now needs to decrease the altitudes of some places. In other words, Ciel wants to decrease some of the altitudes so that `altitude[0]  $\geq$  altitude[1]  $\geq$  ...  $\geq$  altitude[N-1]` holds. It costs 1 unit of money to decrease the altitude of one place by 1 unit of height.

Return the minimal cost required for the change.

## Definition

Class: SkiResortsEasy  
Method: minCost  
Parameters: `int[]`  
Returns: `int`  
Method signature: `int minCost(int[] altitude)`  
(be sure your method is public)

## Constraints

- `altitude` will contain between 2 and 50 elements, inclusive.
- Each element of `altitude` will be between 0 and 1,000, inclusive.

## Examples

0)

`{30, 20, 20, 10}`

Returns: 0

The altitudes are already non-increasing, so Ciel doesn't need to change anything.

1)

`{5, 7, 3}`

Returns: 2

Ciel should change the altitude of place 1 (0-based index) from 7 to 5. This changes the sequence of altitudes to `{5, 5, 3}`. The cost of the change is  $7 - 5 = 2$  units of money.

2)

`{6, 8, 5, 4, 7, 4, 2, 3, 1}`

Returns: 6

Ciel should change the altitudes to `{6, 6, 5, 4, 4, 4, 2, 2, 1}`.

3)

`{749, 560, 921, 166, 757, 818, 228, 584, 366, 88}`

Returns: 2284

4)

`{712, 745, 230, 200, 648, 440, 115, 913, 627, 621, 186, 222, 741, 954, 581, 193, 266, 320, 798, 745}`

Returns: 6393