

Pazza gioia (caduta)

As OII tradition wants, the contest day precedes the long celebrations night. The organizers will throw a wild party in full Trentino style, during which boys and girls will party hard. Obviously, for “partying hard” we mean to indulge in intellectually stimulating activities which, at the same time, aren’t too tiring!

This year, the selected activity was: building an incredibly long chain of dominoes.

The party lasts all night and is abruptly interrupted when Monica comes to take the athletes to the famous Galilei gymnasium, where the awards ceremony will take place.

So far, athletes have been able to place in N tiles, each one exactly 1 centimeter away from the previous one.

Not having enough traditional tiles, they used whatever they could get a hold of: each “tile” could therefore be a few centimeters high but also several meters!

Athletes now want to conclude the party in great style, by pushing the first card and crashing the whole domino chain.

The fall of a tile of height h causes the fall of the following $h - 1$ tiles: so a tile of height 1 doesn’t cause the fall of any other tile.

Because of the exhaustion of the long night, the athletes fear that they might have misplaced some tile.

There is no time to rebuild everything: help them understand if it’s possible to swap two tiles (but only two!) so that the whole domino chain will fall!



Figure 1: Athletes at work

Clarifications

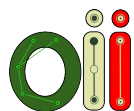
The tiles are indexed from 0 to $N - 1$. Position 0 is the leftmost and the tiles fall from left to right.

After making the exchange, the card in position 0 is dropped first.

If the card at index i falls, and its height is h , then all the cards in the positions up to $i + h - 1$ (included) will fall.

The tile in position 0 always drops, and a tile in position $i \geq 1$ falls if and only if there is a card in position $i' < i$ that falls, with height of at least $i - i' + 1$.

Note that all tiles fall if and only if the tile in position $N - 1$ falls.



Implementation

You will need to submit a single file with extension `.cpp` or `.c`.

📎 Attached to this task you will find a template `catena.cpp` and `catena.c` with an implementation example.

You will need to implement the following function.

```
C/C++ stato_t correggi(int N, int altezze[], coppia_t* scambio);
```

- The integer number N represents the number of tiles.
- The `altezze` array, indexed from 0 to $N - 1$, contains the height of each tile. The tile is initially in position i , for $0 \leq i \leq N - 1$, has height `altezze[i]`.
- The data type `stato_t` is an `enum` that can assume values `OK`, `SOLVED` or `IMPOSSIBILE`. The function must return:
 - `OK`, if no swap is required for all the tiles to fall; otherwise,
 - `SOLVED`, if it is possible to swap two tiles so that, after that, all the tiles fall; otherwise,
 - `IMPOSSIBILE`, when a single exchange is not enough.
- The data type `coppia_t` is a `struct` which contains the fields `domino1` and `domino2`. If the function returns `SOLVED`, the fields `domino1` and `domino2` of the output parameter `scambio` must be populated with the positions of the two tiles to be swapped.

The function `correggi` will be called only once. The return value of the function will be recorded and, if this value is `SOLVED`, the positions in the `scambio` struct will be stored as well.

Sample Grader

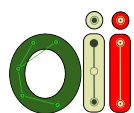
You can find a simplified version of the grader used during the competition in this task's directory. You can use it to test your solutions. The sample grader reads data from `stdin`, calls the functions that you have to implement and writes on `stdout`, according to the following format.

The input file consists of 2 rows, containing:

- row 1, integer number N ;
- row 2, integer numbers `heights[i]`, for $i = 0 \dots N - 1$, separated by spaces.

The output file consists of a single line that contains one of the following options:

- the string `OK`, if the domino chain is completely knocked down without any swap;
- the string `IMPOSSIBLE`, if it is not possible to adjust the chain with only one swap;
- two integer numbers i and j , separated by space, if you can adjust the chain by swapping the tiles in the positions i and j .



Constraints

- $1 \leq N \leq 5\,000\,000$.
- $1 \leq \text{altezze}[i] \leq 1000$ for each $i = 0, \dots, N - 1$.

Scoring

Your program will be tested on several test cases grouped into subtasks. To get the score for a subtasks, you must correctly solve all the tests in it.

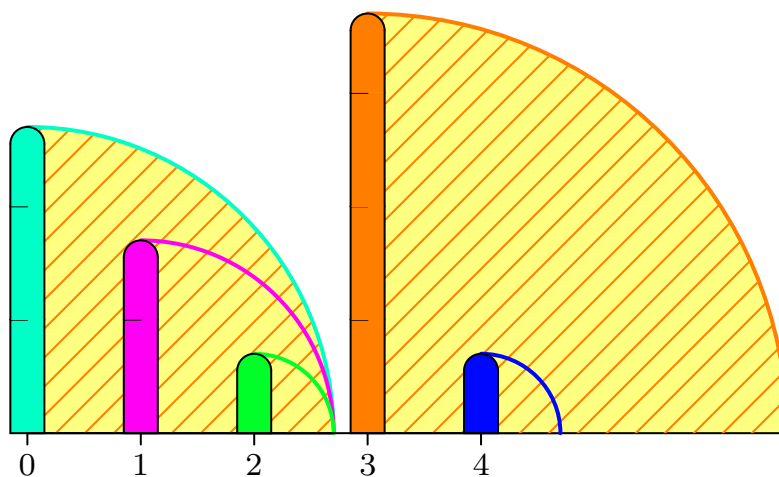
- **Subtask 1** [0 punti]: Sample cases.
- **Subtask 2** [4 punti]: The heights are all the same.
- **Subtask 3** [7 punti]: $N \leq 5000$, the answer is OK or IMPOSSIBILE.
- **Subtask 4** [9 punti]: The answer is OK or IMPOSSIBILE.
- **Subtask 5** [13 punti]: $N \leq 50$.
- **Subtask 6** [19 punti]: $N \leq 1000$.
- **Subtask 7** [28 punti]: $N \leq 3000$.
- **Subtask 8** [11 punti]: $N \leq 100\,000$.
- **Subtask 9** [9 punti]: No specific limitations.

Examples

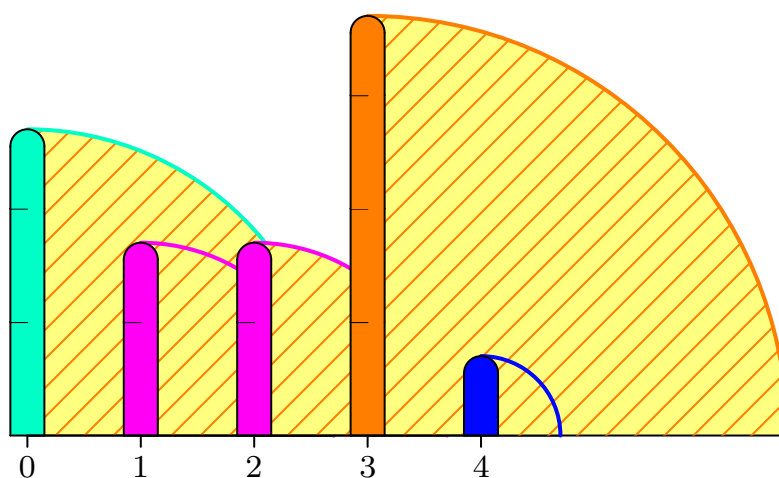
stdin	stdout
5 3 2 1 4 1	2 3
5 3 2 2 4 1	OK
5 1 1 1 1 1	IMPOSSIBILE

Explanation

In **first example** it's sufficient to exchange the tiles in positions 2 and 3:



In **second example** the tile at position 2 is high enough: the chain is fine as it is.



In **third example** the tiles are too low: it's impossible for the chain to fall.

