

## ☆ Snake Sequence

Given a 2D-grid of integers input, find and return the size of the maximum Snake sequence.

A snake sequence is made up of adjacent numbers in the grid such that for each number, the number on the right or the number below it is  $+1$  or  $-1$  its value. For example, if you are at location  $(x, y)$  in the grid, you can either move right i.e.  $(x, y+1)$  if that number is  $\pm 1$  or move down i.e.  $(x+1, y)$  if that number is  $\pm 1$ . Your starting position is always top left corner of the grid.  
*Note: A snake sequence is always of at least length 1.*

For example:

9, 6, 5, 2

8, 7, 6, 5

7, 3, 1, 6

1, 1, 1, 7

In above grid, the longest snake sequence is: (9, 8, 7, 6, 5, 6, 7)  $\rightarrow$  7

Input: The 2-Dimensional integer array.

4

4

9 6 5 2

8 7 6 5

7 3 1 6

1 1 1 7

Output: An integer denoting the max snake sequence.

7

YOUR ANSWER

Original Code

C++



## ☆ Reduce to Zero

Given a number  $n$ , reduce  $n$  to 0 by performing either of the two operations:

1. If  $n = a * b$ , change  $n$  to  $\max(a, b)$  (or)
2. decrement  $n$  by 1

Return the minimum number of moves required to reduce  $n$  to 0

If reducing the number to 0 is impossible, return -1.

Sample Input

6

Sample Output

4

Explanation

1. 3 ( $6 = 2 * 3 \rightarrow 3$ )
  2. 2 (decrement 1)
  3. 1 (decrement 1)
  4. 0 (decrement 1)
- 4 total operations

**YOUR ANSWER**

Original Code

C++

