There are 'n' ants on a 'n+1' length rod. The ants are numbered from 1 to n and are initially placed at positions starting from position 1 till position n. They are moving either in left direction (denoted by '-1') or in the right direction (denoted by '1'). Whenever an ant crosses the boundary of the rod it falls off the rod. You are given the initial direction of the ants. Now, whenever two ants collide their direction switches, i.e. the ant going in left direction ('-1) changes it's direction towards right ('1') and the ant going in the right direction ('1') changes it's direction towards left ('-1'). Find last ant to fall off the rod.   
**Note**: In case two ants are falling simultaneously in the end print the index of the lower indexed ant.   
  
**Input Format:**  
First line contains the integer 'n' denoting the total number of ants s.t. 1 <= n <= 1,000  
Second line contains 'n' space separated numbers (either '1' or '-1') denoting the initial directions of the ants.   
  
**Output Format:**  
Output a single integer which is the index (lower index in case two ants are falling simultaneously in the end) of the last ant to fall off the table.

Consider a matrix M of integers. Divide M into 4 sub-matrices. These sub-matrices are called as Quadrants. Report the Quadrant number which has the smallest minimum-element. If two or more quadrants have same smallest minimum, report the smallest quadrant index.   
The matrix M is divided into four quadrants by halving the rows and columns. If row/column is an odd number, divide them in such a way that the first half of the row/column should be one smaller than the second half.   
  
The four quadrants are numbered from 1 to 4 in the structure shown below: Q1 | Q2  
---+---  
 Q3 | Q4  
**INPUT FORMAT:** M is a matrix of integers. You would be given two numbers m and n specifying the number of rows and columns. This would be followed by m lines of n integers each specifying the data of the m\*n matrix M.   
n and m will be greater than 1 and less than 12. **OUTPUT FORMAT:** Print in a separate line, the quadrant number with the smallest minimum-element.

You are the encoded form of a data string as follows: consecutive  
occurrences of a letter (up to 9) are represented by the letter  
followed by the number of occurrences.  
  
For example, the string  
  
a9a3b2c4de  
  
stands for the string  
  
aaaaaaaaaaaabbccccdc  
  
- that is, 12 consecutive occurrences of a, followed by 2 bs, and then  
  4 cs, followed by a d and finally c  
  
Given an encoded form, you have to output the data string.  
  
Input  
-----  
The encoded form of the string, made as per the following rules.  
1. If a character occurs only once, then in the encoded string, it  
   appears as such (for example, 'd' in the above string.)   
2. If the number of consecutive occurrences of the character is  
   between 2 and 9, then it is represented as the character followed  
   by the number of occurrences (e.g. aaaab is represented as a4b).    
3. If the number of consecutive occurrences of a character is greater  
   than 9, then group 9 occurrences as per rule 2. Iterate the set of  
   rules on the remaining string.  
  
Output  
------  
The original string, consisting only of characters whose  
encoding was given as input.

In a string, a "run" is a substring with consisting of consecutive  
occurrences of the same character. For example, the string  
"mississippi" contains the following runs - "ss", "ss" and "pp".  
  
In this question, given a string, you have to output the length of the  
longest run in the string.  
  
Input  
-----  
A string, having length at most 100. The string is guaranteed to have  
at least one run.  
  
Output  
------  
The length of the longest run in the string.  
  
Sample Input  
------------  
abbaaacccc  
  
Sample Output  
-------------  
4