

Adding up Triangles

Take a look at the triangle on the left of the figure below. It is made of 9 (unit) triangles arranged in three rows ($N = 3$). Needless to say, a unit triangle is a triangle with $N = 1$.



If you study the figure for few seconds, you'll realize that you can find 13 different triangles (which we'll call sub-triangles.) Of these 13 sub-triangles we have: Nine unit triangle; three with $N = 2$, and one with $N = 3$. The following table lists the number of sub-triangles in arrangements with $N < 5$.

# of Rows:	N = 1	N = 2	N = 3	N = 4
# of Sub-triangles:	1	5	13	27

Let's define the value of a unit triangle to be the integer value written in that triangle. In general, the value of a triangle is the sum of values in all its unit triangles. The triangle on the right is the same as the other one but with the sub-triangle having the largest value being highlighted. Write a program to determine the sub-triangle with the largest value.

Input

Your program will be tested on one or more test cases. Each test case is specified in a single line made of integers (separated by spaces.) The first integer is the number of rows in the test case, and the remaining integers are the values of the unit triangles specified in a top-down, left-to-right order. (the first test case in the example below is the same as the one in the figure.) The last line of the input file contains the number 0 (which is not part of the test cases.)

The maximum number of rows is 400. The absolute value of a unit triangle is less than 1000.

Output

For each test case, print the result using the following format:

k. V

where k is the test case number (starting at 1,) is a single space, and V is the maximum value of a sub-triangle in that test case.

Example

Input:
3 6 -24 0 12 -10 12 40 -4 6

4 1 1 -1 1 1 -1 1 1 -1 1 1 -1 1 -1 1
0

Output:

1. 54
2. 4