Problem: Maximum Sum (adaptation of acmuva problem #108)

Time Limit: 2 seconds

The Problem

Given a 2-dimensional array of positive and negative integers, find the sub-rectangle with the largest sum. The sum of a rectangle is the sum of all the elements in that rectangle. In this problem the sub-rectangle with the largest sum is referred to as the maximal sub-rectangle. A sub-rectangle is any contiguous sub-array of size 1x1 or greater located within the whole array. As an example, the maximal sub-rectangle of the array:

0	-2	-7	0
9	2	-6	2
-4	1	-4	1
-1	8	0	-2

is in the lower-left-hand corner:

9	2
-4	1
-1	8

and has a sum of 15.

The Input

The first line of the input is an single positive integer *k (k < 15)*, signifying the number of test cases. Each test case will follow. The first line of each test case will contain a single positive integer n ($n \leq 500$) indicating the size of the square two-dimensional array. This is followed by n lines that contain n integers each separated by white space. There may also be white space preceding the first integer on a line. The k^{th} ($1 \leq k \leq n$) line of input contains the values of the k^{th} row of the array, in order. All of these values will be integers in the range [-127, 127].

The Output

For each test case output the maximal sum of any contiguous sub-rectangle on a line by itself.

Sample Input

```
3
2
3 -1
-5 2
3
6 -2 3
5 1 -7
8 9 -2
4
0   -2    -7   0
9    2    -6   2
-4   1    -4   1
-1   8    0   -2
```

Sample Output

```
3
27
15
```

G: Profits

Your friends have just opened a new business, and you want to see how well they are doing. The business has been running for a number of days, and your friends have recorded their net profit on each day. You want to find the largest total profit that your friends have made during any consecutive time span of at least one day. For example, if your friends' profits looked like this:

```
Day 1: -3
Day 2: 4
Day 3: 9
Day 4: -2
Day 5: -5
Day 6: 8
```

Their maximum profit over any span would be 14, from days 2 to 6.

Input

There will be several test cases in the input. Each test case will begin with an integer **N** (1 ≤ **N** ≤ 250,000) on its own line, indicating the number of days. On each of the next **N** lines will be a single integer **P** (-100 ≤ **P** ≤ 100), indicating the profit for that day. The days are specified in order. The input will end with a line with a single 0.

Output

For each test case, output a single integer, representing the maximum profit over any non-empty span of time. Print each integer on its own line with no spaces. Do not print any blank lines between answers.

Sample Input

```
6
-3
4
9
-2
-5
8
2
-1000
-19
0
```

Sample Output

```
14
-19
```

Square Free Numbers

Filename: sqfree

Danny is afraid of perfect squares, because he once had bad luck on the 1st, 4th, 9th, 16th and 25th of a month. So, whenever he chooses a number for any purpose, he tries to choose numbers that are NOT perfect squares. Help Danny figure out how many numbers he can choose in a particular range.

The Problem

Given a low bound and a high bound, determine the number of integers within those bounds, inclusive, that are NOT perfect squares.

The Input

The first line of the input file will contain a number n $(1 \leq n \leq 10000)$ indicating the number of ranges to be evaluated. The following n lines will each contain two integers, a $(1 \leq a \leq 1000000)$ and b $(a \leq b \leq 1000000)$.

The Output

For range [a, b], output on a single line the number of integers in the range that are not perfect squares.

Sample Input

```
3
20 40
1 3
9999 10000
```

Sample Output

```
19
2
1
```