A - Diophantine equation

DO NOT SUBMIT IN C++ 4.3.2

Sometimes solving a Diophantine equation is very hard. But, for example, the equation $a+b^2+c^3+d^4=n$ has a trivial solution for every value of n. Your task is to determine the number of solutions of the equation for each given n, assuming that in the equation all the values a, b, c and d are non-negative integers.

1 - N

Input

The first line of input contains an integer T, representing the number of test cases (T<20).

The following T lines contain one non-negative integer n each, where $n < 10^9$.

Output

Output T lines, each containing the number of solutions of the respective equation for n.

Example

Input:

```
5
0
1
10
100
1000
```

1+01

Output:

```
1
4
19
148
1476
```

B - IQ test

Bob is preparing to pass IQ test. The most frequent task in this test is to find out which one of the given n numbers differs from the others. Bob observed that one number usually differs from the others in evenness. Help Bob — to check his answers, he needs a program that among the given n numbers finds one that is different in evenness.

Input

The first line contains integer n ($3 \leq n \leq 100$) — amount of numbers in the task. The second line contains n space-separated natural numbers, not exceeding 100. It is guaranteed, that exactly one of these numbers differs from the others in evenness.

Output

Output index of number that differs from the others in evenness. Numbers are numbered from 1 in the input order.

Sample Input

Input
```
5
2 4 7 8 10
```
Output
```
3
```
Input
```
4
1 2 1 1
```
Output
```
2
```

BAPS-BUBT NATIONAL PROGRAMMI
OCTOBER 06-10,2016

C - Flipping Game

...ub got bored, so he invented a game to be played on paper.

He writes n integers $a_1, a_2, ..., a_n$. Each of those integers can be either 0 or 1. He's allowed to do exactly one move: he chooses two indices i and j ($1 \le i \le j \le n$) and flips all values a_k for which their positions are in range $[i, j]$ (that is $i \le k \le j$). Flip the value of x means to apply operation $x = 1 - x$.

The goal of the game is that after exactly one move to obtain the maximum number of ones. Write a program to solve the little game of Iahub.

Input

The first line of the input contains an integer n ($1 \le n \le 100$). In the second line of the input there are n integers: $a_1, a_2, ..., a_n$. It is guaranteed that each of those n values is either 0 or 1.

Output

Print an integer — the maximal number of 1s that can be obtained after exactly one move.

Sample Input

Input
5
1 0 0 1 0
Output
4
Input
4
1 0 0 1
Output
4

100100111

n^2

$0 = 1 - 0$

1001
1101

110011001

1001
01111

Hint

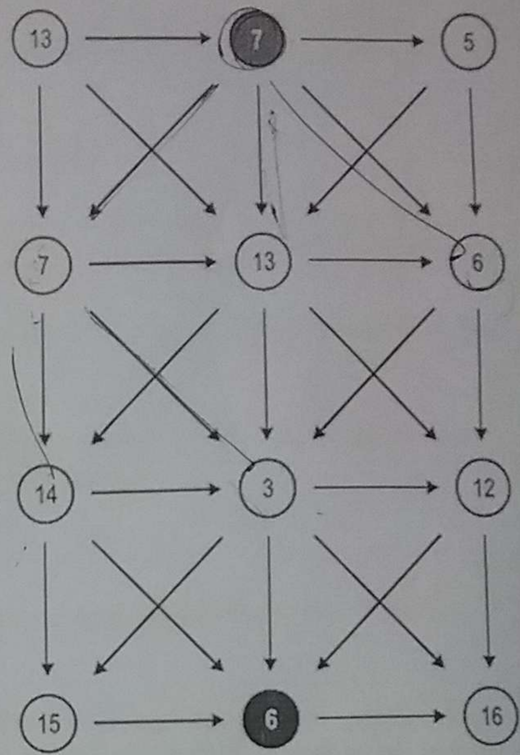
In the first case, flip the segment from 2 to 5 ($i = 2, j = 5$). That flip changes the sequence, it becomes: [1 1 1 0 1]. So, it contains four ones. There is no way to make the whole sequence equal to [1 1 1 1 1].

In the second case, flipping only the second and the third element ($i = 2, j = 3$) will turn all numbers into 1.

10010
11101

110011001

D -Tri graphs



..re's a simple graph problem: Find the shortest path from the top-middle vertex to the bottom-middle vertex in a given tri-graph.

A tri-graph is an acyclic graph of ($N \geq 2$) rows and exactly 3 columns. Unlike regular graphs, the costs in a tri-graph are associated with the vertices rather than the edges. So, considering the example on the right with $N = 4$, the cost of going straight down from the top to bottom along the middle vertices is $7 + 13 + 3 + 6 = 29$. The layout of the directional edges in the graph are always the same as illustrated in the figure.

Input

Your program will be tested on one or more test cases. Each test case is specified using $N + 1$ lines where the first line specifies a single integer ($2 \leq N \leq 100,000$) denoting the number of rows in the graph. N lines follow, each specifying three integers representing the cost of the vertices on the ith row from left to right. The square of each cost value is less than 1,000,000.

The last case is followed by a line with a single zero.

Output

For each test case, print the following line:

1000000

10^8

max

mf ml

k.␣*n*

Where *k* is the test case number (starting at one,) and *n* is the least cost to go from the top-middle vertex to the bottom-middle vertex.

for(i=f; i<=l; i++){

for(j=i; j<=l; j++){

for(tem

i

j

Sample Input

```
4
13 7 5
7 13 6
14 3 12
15 6 16
0
```

Sample Output

```
1. 22
```

E -Sum the Square

...ke any positive number, find the sum of the squares of its digits, repeat! You'll end up with an infinite sequence with an interesting property that we would like to investigate further. Starting with the number 5, the sequence is:

$$(5, 25, 29, 85, 89, 145, 42, 20, 4, 16, 37, 58,...)$$

The interesting part is in what comes after 58: $5^2 + 8^2 = 89$ which is a number that's already been seen in the sequence. In other words, after 58, the sequence will fall into the repeating cycle: 89, 145, 42, 20, 4, 16, 37, 58.

What's amazing is that this cycle will appear for many other numbers: 3, 18, 36, and 64 just to name a few. (see figure on the following page.)

For some numbers, the sequence will fall into another repeating cycle by reaching 1. (see second figure on the following page) For example, starting with 19, you'll end up with the sequence:

$$(19, 82, 68, 100, 1,...)$$

And that's about it. Any number you choose will end up falling into a repeating cycle: Either the 89, 145, ... cycle or the 1, ... cycle.

Given two numbers, your objective is to generate as few numbers in their sequences for the two sequences to intersect at one common number. For example, given 61 and 29, we can achieve what's required by generating the sequences: (61, 37, 58, 89) and (29, 85, 89). Similarly, for 19 and 100, the sequences would be (19, 82, 68, 100) and (100).

Input

Your program will be tested on one or more test cases. Each test case is specified on a single line having two integers ($0 < A, B < 10^9$).

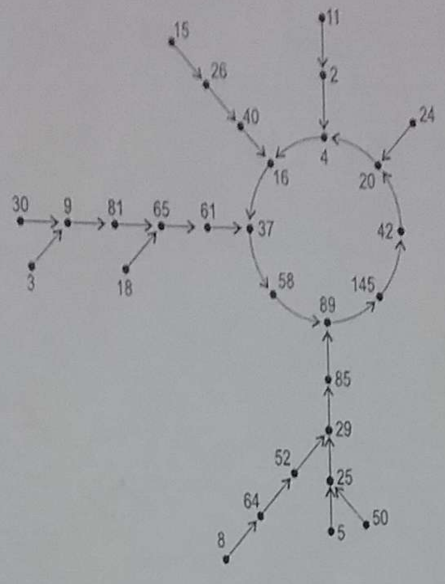
The last case is followed by a dummy line made of two zeros.

Output

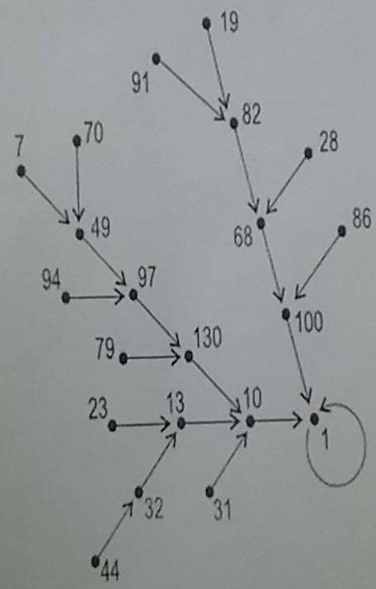
For each test case, print the following line:

$A \sqcup B \sqcup S$

Where A, B are as in the input and S is the (minimum) sum of the lengths of the two sequences. If the sequences starting at A and B do not intersect, then $S = 0$.



Few numbers falling into the 89, 145, 42, 20, 4, 16, 37, 58 cycle.



Few numbers falling into the 1 cycle.

Sample Input

```
89 89
19 100
61 19
0 0
```

Sample Output

```
89 89 2
19 100 5
61 19 0
```