

## Codeforces Round #303 (Div. 2)

### A. Toy Cars

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

Little Susie, thanks to her older brother, likes to play with cars. Today she decided to set up a tournament between them. The process of a tournament is described in the next paragraph.

There are  $n$  toy cars. Each pair collides. The result of a collision can be one of the following: no car turned over, one car turned over, both cars turned over. A car is good if it turned over in no collision. The results of the collisions are determined by an  $n \times n$  matrix  $A$ : there is a number on the intersection of the  $i$ -th row and  $j$ -th column that describes the result of the collision of the  $i$ -th and the  $j$ -th car:

- -1: if this pair of cars never collided. -1 occurs only on the main diagonal of the matrix.
- 0: if no car turned over during the collision.
- 1: if only the  $i$ -th car turned over during the collision.
- 2: if only the  $j$ -th car turned over during the collision.
- 3: if both cars turned over during the collision.

Susie wants to find all the good cars. She quickly determined which cars are good. Can you cope with the task?

#### Input

The first line contains integer  $n$  ( $1 \leq n \leq 100$ ) — the number of cars.

Each of the next  $n$  lines contains  $n$  space-separated integers that determine matrix  $A$ .

It is guaranteed that on the main diagonal there are -1, and -1 doesn't appear anywhere else in the matrix.

It is guaranteed that the input is correct, that is, if  $A_{ij} = 1$ , then  $A_{ji} = 2$ , if  $A_{ij} = 3$ , then  $A_{ji} = 3$ , and if  $A_{ij} = 0$ , then  $A_{ji} = 0$ .

#### Output

Print the number of good cars and in the next line print their space-separated indices in the increasing order.

#### Sample test(s)

input
<pre> 3 -1 0 0 0 -1 1 0 2 -1 </pre>
output
<pre> 2 1 3 </pre>
input
<pre> 4 -1 3 3 3 3 -1 3 3 3 3 -1 3 3 3 3 -1 </pre>
output
<pre> 0 </pre>

## B. Equidistant String

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

Little Susie loves strings. Today she calculates distances between them. As Susie is a small girl after all, her strings contain only digits zero and one. She uses the definition of Hamming distance:

We will define the distance between two strings  $s$  and  $t$  of the same length consisting of digits zero and one as the number of positions  $i$ , such that  $s_i$  isn't equal to  $t_i$ .

As besides everything else Susie loves symmetry, she wants to find for two strings  $s$  and  $t$  of length  $n$  such string  $p$  of length  $n$ , that the distance from  $p$  to  $s$  was equal to the distance from  $p$  to  $t$ .

It's time for Susie to go to bed, help her find such string  $p$  or state that it is impossible.

### Input

The first line contains string  $s$  of length  $n$ .

The second line contains string  $t$  of length  $n$ .

The length of string  $n$  is within range from 1 to  $10^5$ . It is guaranteed that both strings contain only digits zero and one.

### Output

Print a string of length  $n$ , consisting of digits zero and one, that meets the problem statement. If no such string exist, print on a single line "impossible" (without the quotes).

If there are multiple possible answers, print any of them.

### Sample test(s)

input
0001 1011
output
0011

  

input
000 111
output
impossible

### Note

In the first sample different answers are possible, namely — 0010, 0011, 0110, 0111, 1000, 1001, 1100, 1101.

## C. Woodcutters

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

Little Susie listens to fairy tales before bed every day. Today's fairy tale was about wood cutters and the little girl immediately started imagining the choppers cutting wood. She imagined the situation that is described below.

There are  $n$  trees located along the road at points with coordinates  $x_1, x_2, \dots, x_n$ . Each tree has its height  $h_i$ . Woodcutters can cut down a tree and fell it to the left or to the right. After that it occupies one of the segments  $[x_i - h_i, x_i]$  or  $[x_i, x_i + h_i]$ . The tree that is not cut down occupies a single point with coordinate  $x_i$ . Woodcutters can fell a tree if the segment to be occupied by the fallen tree doesn't contain any occupied point. The woodcutters want to process as many trees as possible, so Susie wonders, what is the maximum number of trees to fell.

### Input

The first line contains integer  $n$  ( $1 \leq n \leq 10^5$ ) — the number of trees.

Next  $n$  lines contain pairs of integers  $x_i, h_i$  ( $1 \leq x_i, h_i \leq 10^9$ ) — the coordinate and the height of the  $i$ -th tree.

The pairs are given in the order of ascending  $x_i$ . No two trees are located at the point with the same coordinate.

### Output

Print a single number — the maximum number of trees that you can cut down by the given rules.

### Sample test(s)

input
5 1 2 2 1 5 10 10 9 19 1
output
3

input
5 1 2 2 1 5 10 10 9 20 1
output
4

### Note

In the first sample you can fell the trees like that:

- fell the 1-st tree to the left — now it occupies segment  $[-1;1]$
- fell the 2-nd tree to the right — now it occupies segment  $[2;3]$
- leave the 3-rd tree — it occupies point 5
- leave the 4-th tree — it occupies point 10
- fell the 5-th tree to the right — now it occupies segment  $[19;20]$

In the second sample you can also fell 4-th tree to the right, after that it will occupy segment  $[10;19]$ .

## D. Queue

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

Little girl Susie went shopping with her mom and she wondered how to improve service quality.

There are  $n$  people in the queue. For each person we know time  $t_i$  needed to serve him. A person will be disappointed if the time he waits is more than the time needed to serve him. The time a person waits is the total time when all the people who stand in the queue in front of him are served. Susie thought that if we swap some people in the queue, then we can decrease the number of people who are disappointed.

Help Susie find out what is the maximum number of not disappointed people can be achieved by swapping people in the queue.

### Input

The first line contains integer  $n$  ( $1 \leq n \leq 10^5$ ).

The next line contains  $n$  integers  $t_i$  ( $1 \leq t_i \leq 10^9$ ), separated by spaces.

### Output

Print a single number — the maximum number of not disappointed people in the queue.

### Sample test(s)

input
5 15 2 1 5 3
output
4

### Note

Value 4 is achieved at such an arrangement, for example: 1, 2, 3, 5, 15. Thus, you can make everything feel not disappointed except for the person with time 5.

## E. Paths and Trees

time limit per test: 3 seconds  
memory limit per test: 256 megabytes  
input: standard input  
output: standard output

Little girl Susie accidentally found her elder brother's notebook. She has many things to do, more important than solving problems, but she found this problem too interesting, so she wanted to know its solution and decided to ask you about it. So, the problem statement is as follows.

Let's assume that we are given a **connected** weighted undirected graph  $G = (V, E)$  (here  $V$  is the set of vertices,  $E$  is the set of edges). The shortest-path tree from vertex  $u$  is such graph  $G_1 = (V, E_1)$  that is a tree with the set of edges  $E_1$  that is the subset of the set of edges of the initial graph  $E$ , and the lengths of the shortest paths from  $u$  to any vertex to  $G$  and to  $G_1$  are the same.

You are given a **connected** weighted undirected graph  $G$  and vertex  $u$ . Your task is to find the shortest-path tree of the given graph from vertex  $u$ , the total weight of whose edges is minimum possible.

### Input

The first line contains two numbers,  $n$  and  $m$  ( $1 \leq n \leq 3 \cdot 10^5$ ,  $0 \leq m \leq 3 \cdot 10^5$ ) — the number of vertices and edges of the graph, respectively.

Next  $m$  lines contain three integers each, representing an edge —  $u_i, v_i, w_i$  — the numbers of vertices connected by an edge and the weight of the edge ( $u_i \neq v_i$ ,  $1 \leq w_i \leq 10^9$ ). It is guaranteed that graph is connected and that **there is no more than one edge between any pair of vertices**.

The last line of the input contains integer  $u$  ( $1 \leq u \leq n$ ) — the number of the start vertex.

### Output

In the first line print the minimum total weight of the edges of the tree.

In the next line print the indices of the edges that are included in the tree, separated by spaces. The edges are numbered starting from 1 in the order they follow in the input. You may print the numbers of the edges in any order.

If there are multiple answers, print any of them.

### Sample test(s)

input
3 3 1 2 1 2 3 1 1 3 2 3
output
2 1 2

input
4 4 1 2 1 2 3 1 3 4 1 4 1 2 4
output
4 2 3 4

### Note

In the first sample there are two possible shortest path trees:

- with edges 1 — 3 and 2 — 3 (the total weight is 3);
- with edges 1 — 2 and 2 — 3 (the total weight is 2);

And, for example, a tree with edges 1 — 2 and 1 — 3 won't be a shortest path tree for vertex 3, because the distance from vertex 3 to vertex 2 in this tree equals 3, and in the original graph it is 1.