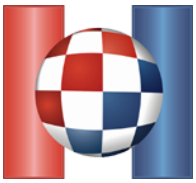

FINAL EXAM #1
23rd April 2014
Tasks overview

TASK	GTA	KOSTA
input data	standard input	
output data	standard output	
time limit	5 seconds	5 seconds
memory limit	256 MB	256 MB
points	**100**	**100**
	total 200	

Dubravka the chemist examines DNA molecules in her laboratory. Each DNA molecule is represented with a series of characters, where each characters is from the set {'A', 'C', 'G', 'T'}. Therefore, 'A', 'ATG' and 'GTA' are series of characters which represent different DNA molecules.

Dubravka can perform the following *mutations* (alterations) in any part of the DNA molecule:

- A $\leftrightarrow$ TC (therefore, the character 'A' can be replaced with'TC' and vice versa)
- C $\leftrightarrow$ AG
- G $\leftrightarrow$ CT
- T $\leftrightarrow$ GA

Dubravka tends to take a certain molecule and alter it by successively applying these mutations. This results in a different molecule, for example:

$$\text{AA} \rightarrow \text{TCA} \rightarrow \text{TAGA} \rightarrow \text{TAT}.$$

Dubravka currently has N molecules in her laboratory. Write a programme that will, for each pair of given molecules, determine whether it is possible to end up with the second molecule when starting from the first molecule by applying the mentioned mutations.

INPUT DATA

The first line of input contains the integer N ($2 \leqslant N \leqslant 100$), the number of molecules. The molecules are marked with numbers from 1 to N.
Each of the following N lines contains molecules – series of characters where each character is an uppercase letter 'A', 'C', 'G' or 'T'. Each series of characters consists of at least one and at most 50 000 characters.

OUTPUT DATA

Output exactly N characters in each of N lines. The j-th character in the i-th line should be '`1`' if it's possible to get molecule j from the molecule i, otherwise it should be '`0`'. You mustn't print spaces between individual characters in the same line.

SCORING

Each of the following points describes a test data worth 10 points.

- $N \leqslant 5$, the molecules are not longer than 4 characters.
- The molecules consist of exactly 2 characters.
- The molecules consist of exactly 3 characters.
- The molecules consist of exactly 4 characters.
- The molecules consist of exactly 5 characters.
- The molecules consist of exactly 6 characters.

Apart from test data mentioned here, there is additional test data (in total worth 40 points) for which no additional limitations apply.

SAMPLE TESTS

input	**input**	**input**
4 A C G T	4 AA TAT C CGTAC	4 AAA CCC TATA CACA
output	**output**	**output**
1000 0100 0010 0001	1100 1100 0011 0011	1111 1111 1111 1111

Grill master Kosta followed his entrepreneurial gut and opened N restaurants in Manhattan marked with numbers from 1 to N, respectively. It is well known that the streets of Manhattan are parallel to coordinate axes, which means that we can describe the locations of restaurants with points in a coordinate system, using integer coordinates. More specifically, restaurant A is located on point (X_A, Y_A). The distance between restaurants A and B is the sum of absolute differences of their coordinates: $|X_A - X_B| + |Y_A - Y_B|$.

Kosta is planning on buying *two at most* modern machines for automatic making of burgers. Each machine is going to be implemented *in an existing restaurant* and the burgers will be delivered by car every morning to all the remaining restaurants. Of course, when the burgers are delivered to the remaining restaurants, they are delivered from the closest restaurant that owns a burger machine. For restaurant C, we define D_C as the distance between restaurant C to its closest restaurant that owns a burger machine. It's bad to keep the burgers in the car for too long, so Kosta wants to pick the locations for the burger machines in a way that the furthest restaurant gets the freshest burgers as soon as possible. In other words, we want the maximum value D_C to be **the minimum possible**.

Write a programme that will, based on the locations of restaurants and the number of machines that Kosta is going to buy (one or two), calculate **the minimum possible value** D so that we can implement the burger machines in existing restaurants, and that the distance to **all the remaining restaurants** is at most D. Additionally, your programme should determine the optimal locations of the machines. If Kosta is buying two machines, it is allowed to implement both machines in the same restaurant.

INPUT DATA

The first line of input contains the integer K ($1 \leqslant K \leqslant 2$), the number of burger machines Kosta is going to buy.
The second line of input contains the integer N, the number of restaurants (limitations for number N depend on number K, see section "Scoring").
Each of the following N lines contains integers X_K and Y_K ($0 \leqslant X_K, Y_K \leqslant 10^6$) – coordinates of an individual restaurant. There will not exist two restaurants on the same location.

OUTPUT DATA

The first line of output must contain an integer – the required minimum possible value D.
The second line of output must contain K space-separated integers – restaurants in which Kosta is implementing the machines.

Please note: The solution doesn't need to be unique.

SCORING

If the first line (minimum possible value D) is correct, and the second line is either not printed or not correct, the competitor will get 60% of points for that sample test.

In test cases worth 4 points total, it will hold $K = 1$ and $3 \leqslant N \leqslant 1\,000$.
In test cases worth 16 points total, it will hold $K = 1$ and $1\,000 < N \leqslant 200\,000$.
In test cases worth 4 points total, it will hold $K = 2$ and $3 \leqslant N \leqslant 100$.
In test cases worth 24 points total, it will hold $K = 2$ and $100 < N \leqslant 3\,000$.
In test cases worth 52 points total, it will hold $K = 2$ and $3\,000 < N \leqslant 50\,000$.

SAMPLE TESTS

input

```
2
5
1 1
2 3
5 10
4 6
7 12
```

output

```
5
1 3
```

input

```
2
10
3 6
1 4
4 1
4 7
4 10
3 8
3 10
6 7
5 1
2 10
```

output

```
5
1 3
```

input

```
1
10
3 10
6 1
5 7
0 4
2 7
2 0
9 2
4 1
3 6
1 4
```

output

```
10
3
```