2017 Canadian Computing Olympiad
Day 1, Problem 1
Vera and Trail Building

Time Limit: 1 second

Problem Description

Vera loves hiking and is going to build her own trail network. It will consist of V places numbered from 1 to V, and E bidirectional trails where the i-th trail directly joins two distinct places a_i and b_i. Vera would like her network to be connected so it should be possible to hike between any two places using the trails. It is possible that there could be more than one trail directly joining the same pair of places.

Vera considers two places a, b with $a < b$ to form a *beautifully connected pair* if it is possible to hike using the trails from a to b then back to a without hiking on the same trail more than once. She thinks her trail network would be beautiful if it had exactly K beautifully connected pairs.

Vera does not want her network to be too large, so the network should satisfy $1 \leq V, E \leq 5000$.

Given K, help Vera find any beautiful trail network.

Input Specification

There is one line of input, which contains the integer K ($0 < K \leq 10^7$).

For 3 of the 25 available marks, $K \leq 1000$.

For an additional 6 of the 25 available marks, $K \leq 10^5$.

Output Specification

Print a beautiful network in the following format:

- the first line should contain the number of vertices, V, followed by one space, followed by the number of edges, E;
- each of the next E lines should contain two integers, a_i and b_i, separated by one space, indicating a trail between places a_i and b_i ($1 \leq i \leq E$).

The trails can be printed in any order. The two places of any trail can be printed in any order. If there are multiple beautiful trail networks, print any of them. It is guaranteed that a solution always exists.

Sample Input 1

```
2
```

Output for Sample Input 1

```
4 5
1 2
2 1
3 4
4 3
1 4
```

Explanation for Output for Sample Input 1

The two beautifully connected pairs are $1, 2$ and $3, 4$.

Sample Input 2

```
6
```

Output for Sample Input 2

```
4 4
1 2
2 3
3 4
4 1
```

Explanation for Output for Sample Input 1

All pairs of places form a beautifully connected pair.

2017 Canadian Computing Olympiad
Day 1, Problem 2
Cartesian Conquest

Time Limit: 2 seconds

Problem Description

Long ago, in the land of Cartesia, there ruled the Rectangle Empire. The Empire was large and prosperous, and it had great success with expanding its territory through frequent conquests. The citizens of this ancient civilization followed many curious customs. Unfortunately, the significance of these are now shrouded in mystery.

The Rectangle Empire operated under a system of rectangular districts. These districts were carefully managed to meet three special criteria.

1. The Empire's territory is divided into districts such that each piece of land controlled by the Empire belongs to exactly one district.
2. The boundaries of the districts, when viewed on a map, must be rectangles such that the length of the longer side of the rectangle is twice the length of the shorter side.
3. The side lengths of the districts must be integers, when measured in Ξ (note that Ξ was the primary unit of length in the Rectangle Empire).

When the empire was first established, it consisted of a single district. Since then, the empire has gained additional districts through conquest of neighbouring regions. Whenever the empire gained control over a new region of land, they always established a single new district using that exact land. This means that the empire was always mindful about the geometric properties of the land they were hoping to conquer. You can assume that no two of these conquests occurred at the same time.

The addition of new districts was the only way that the boundaries of the empire ever changed. Furthermore, each district, once added, was never modified or merged with another.

The final, most important tradition of the Rectangle Empire was to make sure that the overall territory of the empire was always a rectangle, though it did not necessarily need to satisfy the 2:1 ratio for the side lengths that individual districts satisfy.

Recently, archeologists have discovered that at one point in time, the empire had dimensions N by M (measured in Ξ). You need not be alarmed if these numbers are very large; after all, Cartesia is an infinite plane. Your task is to estimate the number of districts in the empire when it was at this size. Over all possible ways that the empire was founded and expanded, what is the minimum and maximum number of districts?

Input Specification

The input will be a single line, containing two integers N and M ($1 \leq N, M \leq 10^8$).

For 5 of the 25 marks available, $N, M \leq 1000$.

For an additional 8 of the 25 marks available, $N, M \leq 10^6$.

Output Specification

Output a single line, containing the minimum number of districts, followed by a space, followed by the maximum number of districts.

Sample Input

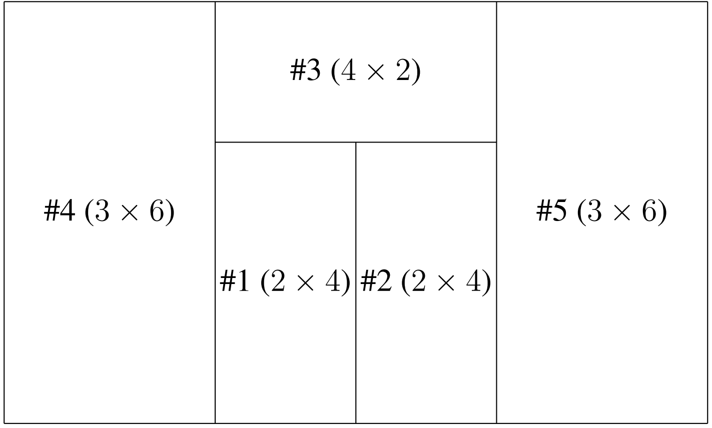
```
10 6
```

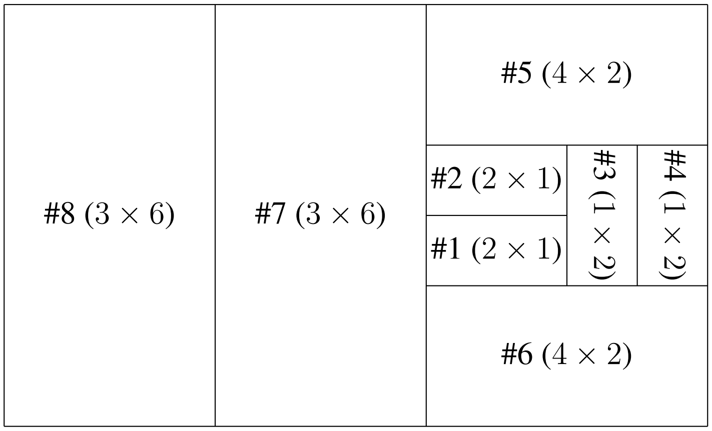
Output for Sample Input

```
5 8
```

Explanation of Output for Sample Input

The illustrations below show how this minimum and maximum number of districts could have been achieved. Districts are labelled #1, #2, #3, ... giving the order in which they were added to the region. The dimensions of each district is shown in brackets as $(k \times 2k)$ or $(2k \times k)$:





2017 Canadian Computing Olympiad
Day 1, Problem 3
Vera and Modern Art

Time Limit: 4 seconds

Problem Description

After being inspired by the great painter Picowso, Vera decided to make her own masterpiece. She has an empty painting surface which can be modeled as an infinite 2D coordinate plane. Vera likes powers of two $(1, 2, 4, 8, 16, \ldots)$ and will paint some some points in a repeated manner using step sizes which are a power of two.

Vera will paint N times. The i-th time can be described by three integers x_i, y_i, v_i. Let a_i be the largest power of two not greater than x_i and let b_i be the largest power of two not greater than y_i. Vera will add a paint drop with colour v_i to all points that are of the form $(x_i + a_i p, y_i + b_i q)$, where p, q are non-negative integers. A point may have multiple paint drops on it or have multiple drops of the same colour.

Then Vera will ask Q questions. For the j-th question she wants to know the colour at the point (r_j, c_j). The colour at a point is equal to the sum of the colours of all paint drops at that point. If there are no paint drops at a point, the colour of that point is 0.

Since you are forced to be her art assistant, you will have to answer Vera's questions.

Input Specification

The first line contains two integers N, Q, separated by one space ($1 \leq N, Q \leq 2 \cdot 10^5$).

The next N lines each contain three space-separated integers, x_i, y_i, v_i representing the paint drops of colour v_i ($1 \leq i \leq N$; $1 \leq v_i \leq 10\,000$; $1 \leq x_i, y_i \leq 10^{18}$).

The next Q lines each contain two space-separated integers r_j, c_j, representing the Q questions about the point (r_j, c_j) ($1 \leq j \leq Q; 1 \leq r_j \leq 10^{18}; 1 \leq c_j \leq 10^{18}$).

For 5 of the 25 available marks, $N, Q \leq 2000$.

For an additional 5 of the 25 available marks, $y_i = 1$ ($1 \leq i \leq N$).

For an additional 5 of the 25 available marks, $N, Q \leq 10^5$ and $1 \leq r_j, c_j \leq 10^9$ ($1 \leq j \leq Q$).

Output Specification

The output will be Q lines. The j-th line ($1 \leq j \leq Q$) should have one integer, which is the colour of point (r_j, c_j).

Sample Input

```
5 6
1 2 1
3 4 2
4 5 3
6 3 4
7 1 5
2 6
7 8
5 9
11 2
10 7
4 5
```

Output for Sample Input

```
1
8
0
6
4
3
```

Explanation of Output for Sample Input

Let colour 1, 2, 3, 4, 5 be red, blue, green, orange, purple respectively.

Let p, q be non-negative integers, then:

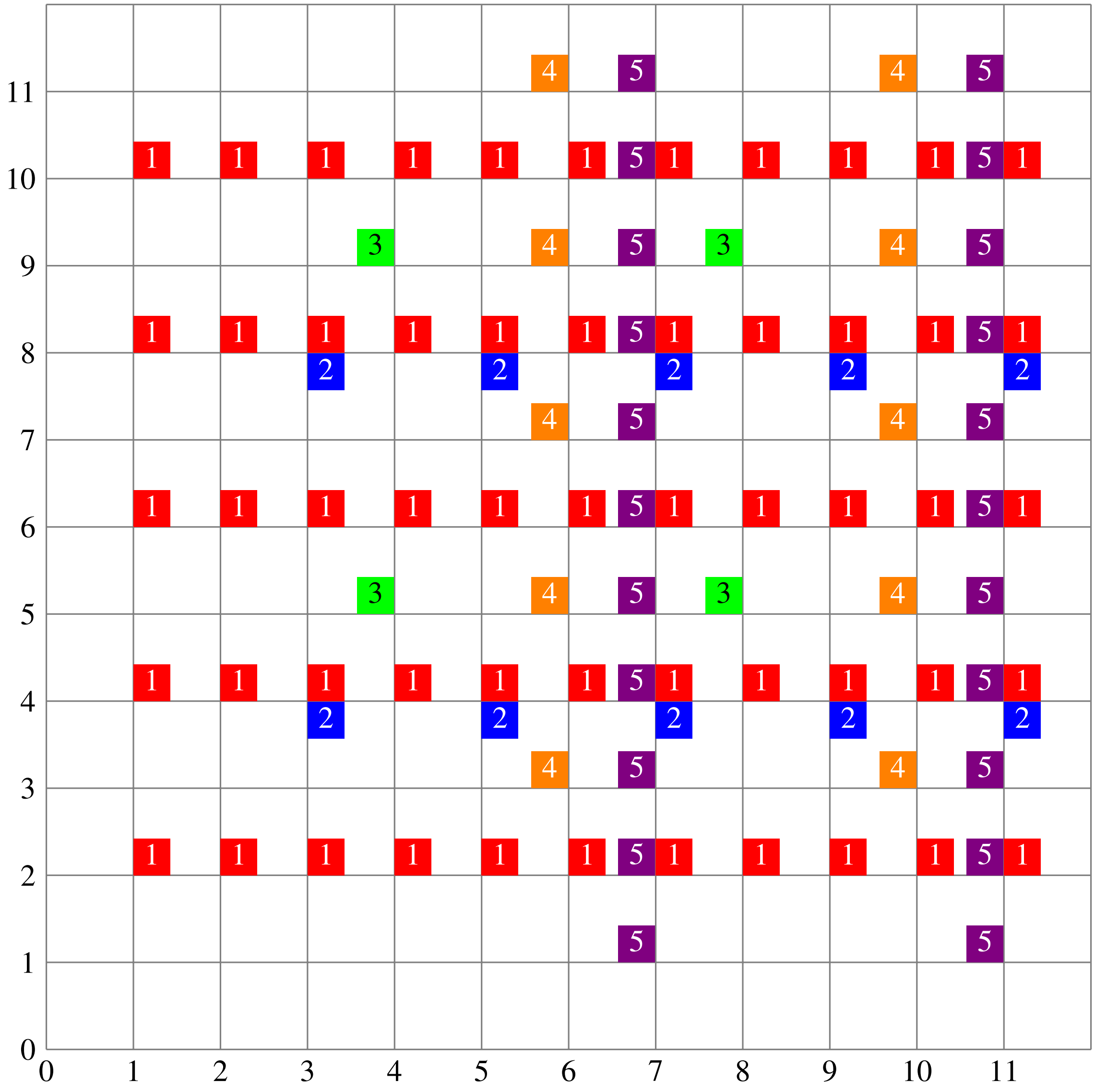
- Points $(1 + p, 2 + 2q)$ have a red paint drop.
- Points $(3 + 2p, 4 + 4q)$ have a blue paint drop.
- Points $(4 + 4p, 5 + 4q)$ have a green paint drop.
- Points $(6 + 4p, 3 + 2q)$ have a orange paint drop.
- Points $(7 + 4p, 1 + q)$ have a purple paint drop.

The painting from $(0, 0)$ to $(11, 11)$ is shown on the next page:

We can see that:

- $(2, 6)$ has a red paint drop, so it has colour 1.
- $(7, 8)$ has a red, blue and purple paint drop, so it has colour $1 + 2 + 5 = 8$.

- $(5, 9)$ has no paint drops, so it has colour 0.
- $(11, 2)$ has a red and purple paint drop, so it has colour $1 + 5 = 6$.
- $(10, 7)$ has a orange paint drop, so it has colour 4.
- $(4, 5)$ has a green paint drop, so it has colour 3.



2017 Canadian Computing Olympiad
Day 2, Problem 1
Rainfall Capture

Time Limit: 1 second

Problem Description

It was a dark and stormy night. It also rained, and rained, and rained.

Lucy wants to capture some of the rain, but she only has limited materials. She has a collection of pillars, of various heights, which she can configure to capture the rain. Each pillar is an integer height and has length and width of 1. Once Lucy has her configuration of pillars, she has enough other siding material to enclose the front and back to allow rain to fill the all the available space in between pillars. There is more than enough rain and any excess rain will overflow and get absorbed into the earth.

For example, if Lucy has pillars of height 1, 5, 2, 1, 4, she could configure them as follows (all configurations are illustrated from the side):

```
 *
 *  *
 *  *
 ** *
*****
```

which would capture 5 units of rain (R) as follows:

```
 *
 *RR*
 *RR*
 **R*
*****
```

For this first collection of pillars (1, 5, 2, 1, 4), she could also capture 6 units of rain as follows:

```
 *
 *RR*
 *RR*
**RR*
*****
```

As another example, if the collection of pillars was 5, 1, 5, 1, 5, Lucy could capture 8 units of rain as follows:

```
⋆R⋆R⋆
⋆R⋆R⋆
⋆R⋆R⋆
⋆R⋆R⋆
⋆⋆⋆⋆⋆
```

Finally, this configuration of (5, 1, 4, 1, 5) captures 9 units of rain:

```
⋆RRR⋆
⋆R⋆R⋆
⋆R⋆R⋆
⋆R⋆R⋆
⋆⋆⋆⋆⋆
```

Lucy has N pillars ($2 \leq N \leq 500$) with heights h_1, $h_2 \ldots h_N$ ($1 \leq h_i \leq 50$). She would like to know, of all possible configurations of pillars, what are all of the obtainable volumes of rainfall that she can capture using these N pillars.

Input Specification
The first line contains the integer N ($2 \leq N \leq 500$) signifying the number of pillars. The next line contains the integers h_i ($1 \leq h_i \leq 50$, $1 \leq i \leq N$), representing the ith pillar height.

For 5 of the 25 marks available, $N \leq 10$.

For an additional 10 of the 25 marks available, $N \leq 50$.

Output Specification
On one line, output a space-separated list of all possible obtainable integer volumes of rain captured, in increasing order.

Sample Input 1

```
5
1 5 2 1 4
```

Output for Sample Input 1

```
0 1 2 3 4 5 6 8
```

Explanation for Output for Sample Input 1
This is the first given example.

Sample Input 2

```
5
5 1 5 1 5
```

Output for Sample Input 2

```
0 4 8
```

Explanation for Output for Sample Input 2

This is the second given example.

Sample Input 3

```
5
5 1 4 1 5
```

Output for Sample Input 3

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

Explanation for Output for Sample Input 3

This is the third given example.

2017 Canadian Computing Olympiad
Day 2, Problem 2
Professional Network

Time Limit: 2 seconds

Problem Description
Kevin is developing his professional network within a certain community. Unfortunately, he has not connected with anybody yet. But he has his eyes on N potentially valuable connections, numbered from 1 to N. He is determined to connect with them all.

However, few people in this community are willing to friend an outsider. Each of the N people Kevin wants to connect with has similar, but different criteria for determining who is an outsider and who isn't. Person i is willing to friend Kevin if he either has at least A_i connections within the community already, or if Kevin gives this person B_i Internet Points.

Kevin likes his Internet Points very much, and so he doesn't want to give away too many. Now it is your job to help Kevin give away the least number of Internet Points while still making connections with each of the N people.

Input Specification
The first line will contain the integer N $(1 \leq N \leq 200\,000)$. Each of the next N lines will contain integers A_i and B_i $(1 \leq i \leq N; 0 \leq A_i \leq N; 0 \leq B_i \leq 10\,000)$.

For 2 of the 25 available marks, $B_i = 1$ for all i.
For an additional 4 of the 25 available marks, $N \leq 10$.
For an additional 7 of the 25 available marks, $N \leq 1000$.

Output Specification
Output one integer on a single line, the minimum number of Internet Points Kevin has to give away.

Sample Input 1

```
4
3 3
1 2
0 5
3 4
```

Output for Sample Input 1

```
3
```

Explanation for Output for Sample Input 1

Kevin can connect with person 3 immediately, and with this connection made, he can also connect with person 2. He doesn't have enough connections to connect with person 1 or person 4, so he gives 3 Internet Points to person 1 to acquire 3 total connections which enables him to connect with person 4.

Sample Input 2

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

Output for Sample Input 2

```
0
```

Explanation for Output for Sample Input 2

It is possible that Kevin can connect with everyone without giving away any Internet Points.

Sample Input 3

```
3
0 6
2 7
3 8
```

Output for Sample Input 3

```
8
```

Explanation for Output for Sample Input 3

Kevin should connect with person 1 immediately, then give 8 Internet Points to person 3 to connect with them, then connect with person 2.

2017 Canadian Computing Olympiad
Day 2, Problem 3
Shifty Grid

Time Limit: 2 seconds

Problem Description
You are given a rectangular grid of numbered tiles, with no empty spaces. This grid can only be manipulated using a sequence of *shift* operations. A shift involves either moving an entire row left or right by some number of units, or moving an entire column up or down by some number of units. Tiles which move outside of the rectangular boundaries wrap around to the opposite side of the grid. For example, in the grid

```
0   1   2   3
4   5   6   7
8   9   10  11
12  13  14  15
```

a vertical shift downwards by one applied to the second column has the following result:

```
0   13  2   3
4   1   6   7
8   5   10  11
12  9   14  15
```

Notice that a left shift by K units is the same a right shift by $N - K$ units. Similarly, an upward shift by K units is a downward shift by $M - K$ units. Thus, without loss of generality, we will restrict the shift directions to be only to the right or down.

In a grid with N rows and M columns, there are NM tiles in total. You may assume that the tiles are numbered with distinct integers from 0 to $NM - 1$.

You may have noticed that in the first example given above, the tiles are in a very organized formation. We call such arrangements *solved*. That is, a grid of tiles is solved when the first row contains the numbers from 0 to $M-1$ in order, the second row has the numbers from M to $2M-1$ in order, and so on, with the last row having the number $(N-1)M$ to $NM-1$ in order.

Find a sequence of shift operations that restores a scrambled grid to a solved state.

Input Specification
The first line will contain two space-separated integers N and M ($2 \leq N, M \leq 100$). The next N lines will contain M space-separated integers, representing the grid.

Note that both N and M will always be even, and there will be a solution requiring at most 10^5 shift operations.

For 5 of the available 25 marks, $N \cdot M \leq 8$.

For an additional 10 of the available 25 marks, the puzzle is solvable in at most 2 moves.

Output Specification
Output any sequence of moves that solves the puzzle, in the following format:

- The first line of output should contain a single integer K ($1 \leq K \leq 10^5$), representing the number of moves in the sequence.
- The next K lines should be either of the form `1` i r $(1 \leq i \leq N, 0 \leq r < M)$ representing a right shift of the i-th row by r, or of the form `2` j d $(1 \leq j \leq M, 0 \leq d < N)$ representing a down shift of the j-th column by d.

Sample Input 1

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

Output for Sample Input 1

```
2
2 1 1
1 1 1
```

Explanation for Output for Sample Input 1
We shift the first column down by one to obtain

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

then shift the first row right by one to reach the state

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

which is solved.

Sample Input 2

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

Output for Sample Input 2

```
7
1 1 1
2 1 1
1 2 1
1 3 1
2 1 2
1 1 1
2 1 1
```

Explanation for Output for Sample Input 2

The sequence of shifts, starting from the input is:

```
2 3     3 2     6 2     6 2     6 2     1 2     2 1     0 1
5 0 ->  5 0 ->  3 0 ->  0 3 ->  0 3 ->  4 3 ->  4 3 ->  2 3
4 1     4 1     5 1     5 1     1 5     6 5     6 5     4 5
6 7     6 7     4 7     4 7     4 7     0 7     0 7     6 7
```