Hong Kong Olympiad in Informatics 2018/19 Junior Group

Task Overview

ID	Name	Time Limit	Memory Limit	Subtasks
J191	Alice and Wings	1.000 s	256 MB	4 + 12 + 20 + 28 + 36
J192	Bigger, Better	1.000 s	256 MB	18 + 27 + 21 + 19 + 15
J193	Hyper Knight II	1.000 s	256 MB	16 + 19 + 11 + 12 + 16 + 26
J194	Graffiti	0.500 s	256 MB	5 + 9 + 5 + 16 + 19 + 21 + 25

Notice:
Unless otherwise specified, inputs and outputs shall follow the format below:

- One space between a number and another number or character in the same line.
- No space between characters in the same line.
- Each string shall be placed in its own separate line.
- Outputs will be automatically fixed as follows: Trailing spaces in each line will be removed and an end-of-line character will be added to the end of the output if not present. All other format errors will not be fixed.

C++ programmers should be aware that using C++ streams (`cin` / `cout`) may lead to I/O bottlenecks and substantially lower performance.

For some problems 64-bit integers may be required. In Pascal it is `int64`. In C/C++ it is `long long` and its token for `scanf` /`printf` is `%lld`.

All tasks are divided into subtasks. You need to pass all test cases in a subtask to get points.

J191 - ALICE AND WINGS

Time Limit: 1.000 s / Memory Limit: 256 MB

Alice once loved eating Wings. However, since last year someone wrote a program for her to find the optimum way to order them, she has become less interested. One day, an idea came up, and she decided to create a startup business selling Wings.

Purchasing raw Wings from the supplier is different from buying fried Wings from a fast food chain. Alice feels disappointed because she cannot use the code written last year, what an unfortunate. She also foresees that the longer the restaurant has run, the more customers she will have, and therefore the more Wings she will order.

She first fixes a number M. On Day i since the start of her business, the supplier will send $(i+1)$ boxes to Alice's restaurant, with the first box containing a fixed amount of B (possibly zero) Wings, and the remaining boxes containing A Wings each. Then she gathered all the Wings and group them into packs of M. These packs will be moved to the storage room. Alice will bring the remaining fewer-than-M (possibly zero) Wings home.

Suppose $A = 2$, $B = 3$ and $M = 6$.
On Day 1 there will be 1 box with 2 Wings and 1 box with 3 Wings. There are a total of 5 Wings. Since there are fewer than $M = 6$ Wings, Alice will bring 5 Wings home.
On Day 2 there will be 2 boxes with 2 Wings and 1 box with 3 Wings. There are a total of 7 Wings. Alice will put 6 Wings into the storage room and take 1 Wing home.
On Day 3 there will be 3 boxes with 2 Wings and 1 box with 3 Wings. There are a total of 9 Wings. Alice will put 6 Wings into the storage room and take 3 Wings home.

You, as her roommate, has observed and recorded the number of number of Wings she brought back home the first three days (Day 1 - 3). You are to write a program to determine the numbers A, B and M. It is known that A and B are both non-negative integers less than M.

There might be chances that there exist no valid solutions for A, B and M. This may be due to several reasons, such as the wrong amount of Wings was delivered from the supplier, or Alice might have already eaten some of them (who knows), but you need not make these assumptions. You only need to compute from what you have recorded.

INPUT

The only line of input consists three integers: W_1, W_2, W_3, indicating the number of Wings Alice brought back home on the first, second and third day respectively.

OUTPUT AND SCORING

If there is no solution, output `No solution`.
Otherwise, the only line of output containing three integers, A, B, M. The output has to satisfy $0 \leq A, B < M$

In case there may be multiple solutions, the set of answers with minimal M is more desirable.

You can get 100% for any correct output with minimal M or `No solution`, and 50% for any other correct output with $M \leq 10^{18}$.

You score for each subtask is the lowest score among all test cases within that subtask.

SAMPLE TESTS

	Input	Output
1	`2 4 6`	`2 0 7`

This sample scores 100% of the points

	Input	Output
2	`2 4 6`	`2 0 2018`

This sample only scores 50% of the points

	Input	Output
3	`5 1 3`	`2 3 6`

This corresponds to the example in the problem statement.

	Input	Output
4	`2 6 9`	`No solution`
5	`3 5 0`	`2 1 7`
6	`6 0 13`	`13 12 19`
7	`27 57 87`	`30 85 88`

SUBTASKS

For all cases: $0 \leq W_1, W_2, W_3 \leq 3 \times 10^8$

	Points	Constraints
1	4	$W_1 = W_2$
2	12	$0 \leq W_1 \leq W_2 \leq W_3$ or $W_1 \geq W_2 \geq W_3 \geq 0$
3	20	$0 \leq W_1, W_2, W_3 \leq 100$
4	28	$0 \leq W_1, W_2, W_3 \leq 3000$
5	36	No additional constraints

J192 - BIGGER, BETTER

Time Limit: 1.000 s / Memory Limit: 256 MB

In 2078, every people loves big numbers, Hitomi lives in 2078 and therefore she loves big numbers. As a Mahoutsukai (aka. witch), she can perform some magic on some numbers to change their value.

Recently, she has just learnt a new skill called "Mark Every Definite Integral As Number". Given an array $A_1, A_2, \ldots, A_N$ with N numbers (not necessarily distinct) and K, she can pick K elements (not necessarily continuous) from the array, and change all of them to their median (see the definition below). After performing the magic, she wants the sum of all numbers to be as large as possible. Her grandma always knows how to choose those K numbers optimally but Hitomi doesn't. Her grandma knows the magic of time and sent her back to 2018 and not letting Hitomi back to 2078 until she knows the solution.

Hitomi is very despair. However, she found that the HKOI finalist in 2018 is very clever and may help her solves the problem. Can you help her?

The median of K numbers is defined as follow:

- If K is odd, then the median is the $\frac{K+1}{2}^{th}$ smallest element of those K numbers, e.g. the median of $[1, 3, 4]$ is 3.
- If K is even, then the median is the average of the $\frac{K}{2}^{th}$ smallest element and the $(\frac{K}{2}+1)^{th}$ smallest element, e.g. the median of $[1, 5, 8, 9]$ is $\frac{5+8}{2} = 6.5$

INPUT

The first line of the input consists of two integers N and K.

The second line of the input consists of N integers, the i^{th} one is A_i.

OUTPUT

Your output should consist of K integers (in arbitrary order), those to be chosen such that the sum of all numbers after performing "Mark Every Definite Integral As Number" is maximized.

If there are more than one optimal solution, you can output any of them.

SAMPLE TESTS

	Input	Output
1	5 3 8 1 2 3 4	1 3 4

The median of $[1, 3, 4]$ is 3.
After performing "Mark Every Definite Integral As Number", the array becomes $[8, 3, 2, 3, 3]$, the sum is 19.

	Input	Output
2	4 4 1 2 2 1	2 1 1 2

The median of $[1, 2, 2, 1]$ is 1.5.
After performing "Mark Every Definite Integral As Number", the array becomes $[1.5, 1.5, 1.5, 1.5]$, the sum is 6.

SUBTASKS

For all cases:
$1 \le K \le N \le 5 \times 10^5$
$1 \le A_i \le 5 \times 10^5$ for all $1 \le i \le N$

	Points	Constraints
1	18	$3 \le N \le 100$ $K = 3$
2	27	$1 \le K \le N \le 4000$ K is odd
3	21	$1 \le K \le N \le 4000$
4	19	K is odd
5	15	No additional constraints

J193 - HYPER KNIGHT II

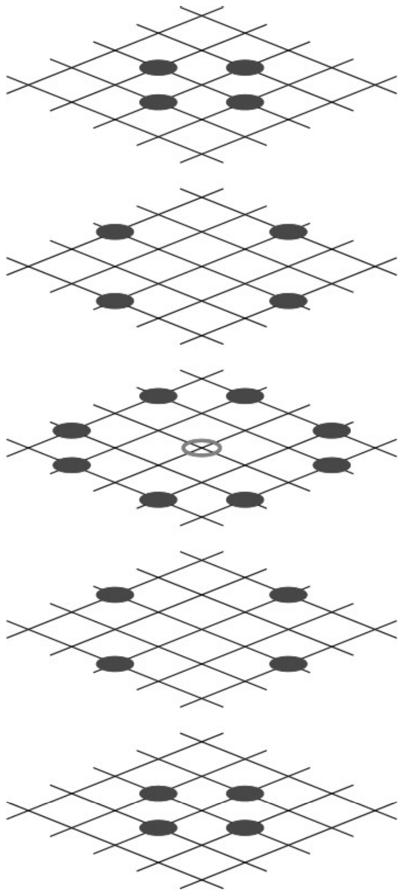
Time Limit: 1.000 s / Memory Limit: 256 MB

Bob is a knight living in a 3-dimensional world. Now, he is drunk and lost his horse. Thus, he can only move with some special restrictions:

- The coordinates of the source and the destination of each step must be integers. (They can be negative.)
- The step size must be $\sqrt{5}$ units since Bob lost control to his legs when his feet were $\sqrt{5}$ units apart from each other.

Therefore, in one step Bob needs to move by 2 units in one dimension, and 1 unit in one of the other two dimensions. In other words, if Bob moves from (x_a, y_a, z_a) to (x_b, y_b, z_b) in one step, then the set $\{|x_a - x_b|, |y_a - y_b|, |z_a - z_b|\} = \{0, 1, 2\}$.

The coordinates of Bob's current location is (X, Y, Z) and his home is at $(0, 0, 0)$. Going home is not easy for a drunk man. Help Bob find a way to go home with limited number of steps. You are not required to find the way with minimum number of steps.



INPUT

The first line contains three integers, X, Y and Z. (X, Y, Z) are the coordinates of Bob's current location.

The second line contains one integer, K, the maximum allowed number of steps.

OUTPUT

On the first line, output one integer, S, the number of steps used in Bob's route.

Then output $S+1$ lines. The i^{th} of the $S+1$ lines contains 3 integers, x_i, y_i and z_i. (x_i, y_i, z_i) are the coordinates of the i^{th} location Bob is going to step on.

Note that your output should satisfy the following:

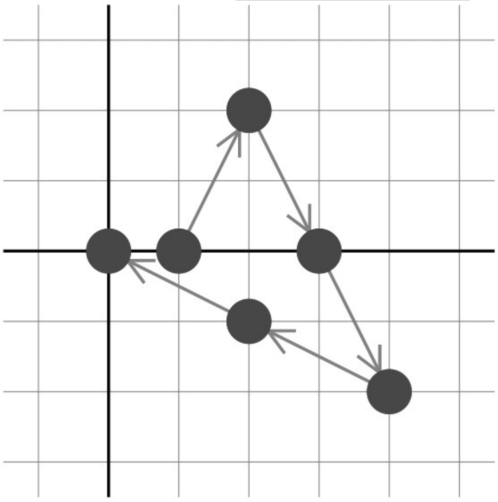
- $0 \leq S \leq K$
- $(x_1, y_1, z_1) = (X, Y, Z)$
- $(x_{S+1}, y_{S+1}, z_{S+1}) = (0, 0, 0)$
- $\sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2 + (z_i - z_{i+1})^2} = \sqrt{5}$ for all $i = 1, 2, \ldots, S$

It can be shown that there always exists at least one possible way satisfying the restrictions. If there are several possible ways, output any of them.

SAMPLE TESTS

	Input	Output
1	1 0 2 128000	1 1 0 2 0 0 0

	Input	Output
2	1 1 -2 128000	2 1 1 -2 1 2 0 0 0 0
3	1 0 0 128000	5 1 0 0 2 2 0 3 0 0 4 -2 0 2 -1 0 0 0 0



SUBTASKS

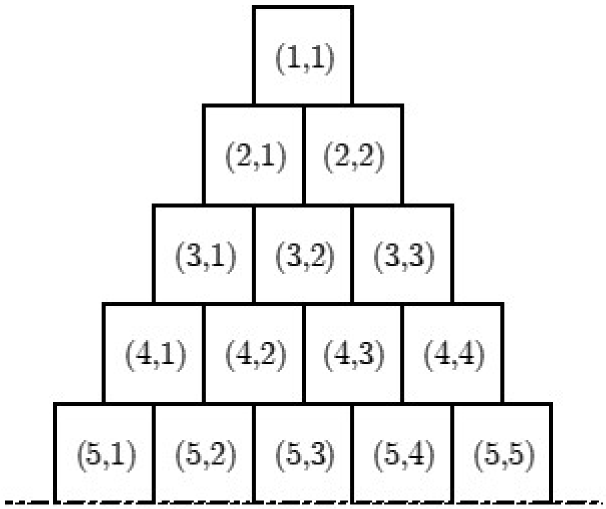
For all cases: $-12000 \leq X, Y, Z \leq 12000$

	Points	Constraints
1	16	$K = 128000$
2	19	$K = 25600$ $Y = Z = 0$
3	11	$K = 25600$ $Z = 0$
4	12	$K = 25600$
5	16	$K = 14400$ or 25600
6	26	$K = 12800$ or 14400 or 25600

J194 - GRAFFITI

Time Limit: 0.500 s / Memory Limit: 256 MB

Bob has built a triangular wall of height N with his cubic bricks. He numbered the rows as 1 to N from top to bottom, so the row on the top is row 1, and the row touching the ground is row N. For all $1 \le i \le N$, on row i, there are exactly i bricks, numbered as 1 to i from left to right. We denote the brick c on row r as (r, c). The wall is built in a way such that each brick not touching the ground is placed on two of the bricks from the row below it. In other words, for all $1 \le j \le i < N$, brick (i, j) is placed on the bricks $(i+1, j)$ and $(i+1, j+1)$.



(A wall of height 5)

Initially, all the bricks are white. Alice has painted K distinct bricks black. She gets tired and asks Bob to paint some (possibly zero) other bricks black, such that the wall is *harmonious*. A wall is *harmonious* if and only if **BOTH** of the following conditions are met for all bricks (i, j) not on the last row:

- if brick (i, j) is black, then **BOTH OF THE TWO** bricks right below it (i.e. $(i+1, j)$ and $(i+1, j+1)$) must be black.
- if brick (i, j) is white, then **AT LEAST ONE** of the two bricks right below it (i.e. $(i+1, j)$ and $(i+1, j+1)$) must be white.

Please help Bob find out the minimum number of bricks that he has to paint black, so that the wall is harmonious.

INPUT

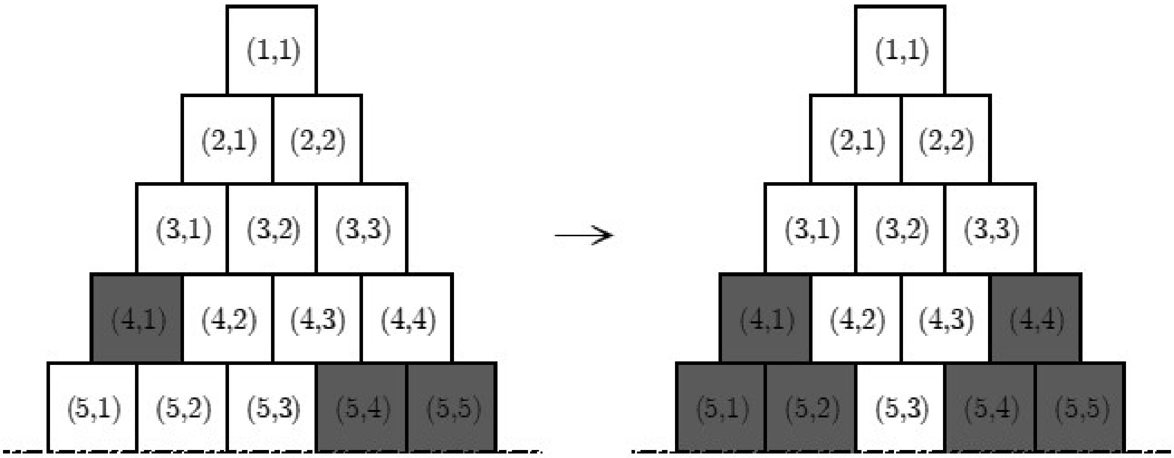
The first line contains two integer N and K.
Each of the next K lines consists of two integers x and y, stating that brick (x, y) is painted black by Alice. It is guaranteed that $1 \le y \le x \le N$, and the input bricks are distinct.

OUTPUT

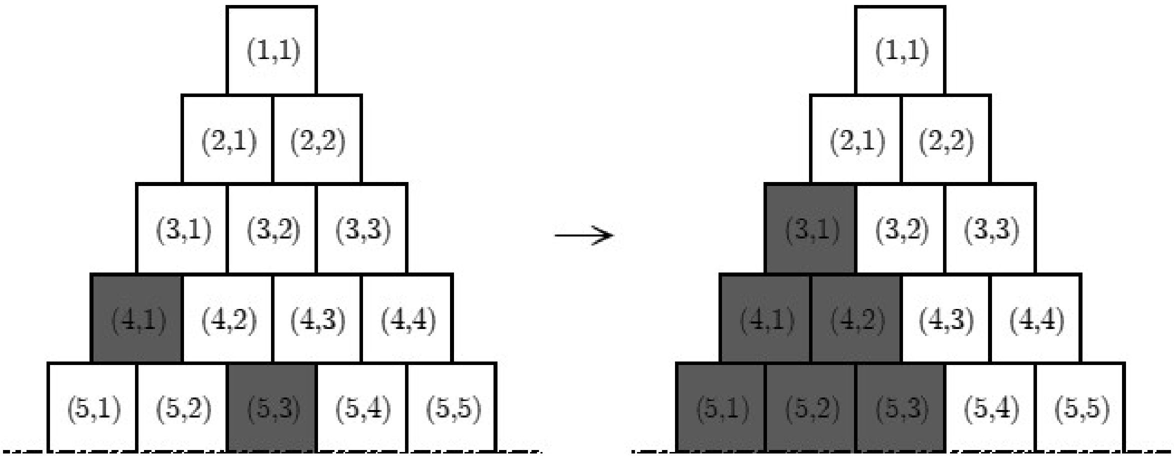
Output a single integer, the minimum number of bricks that have to be painted black by Bob.

SAMPLE TESTS

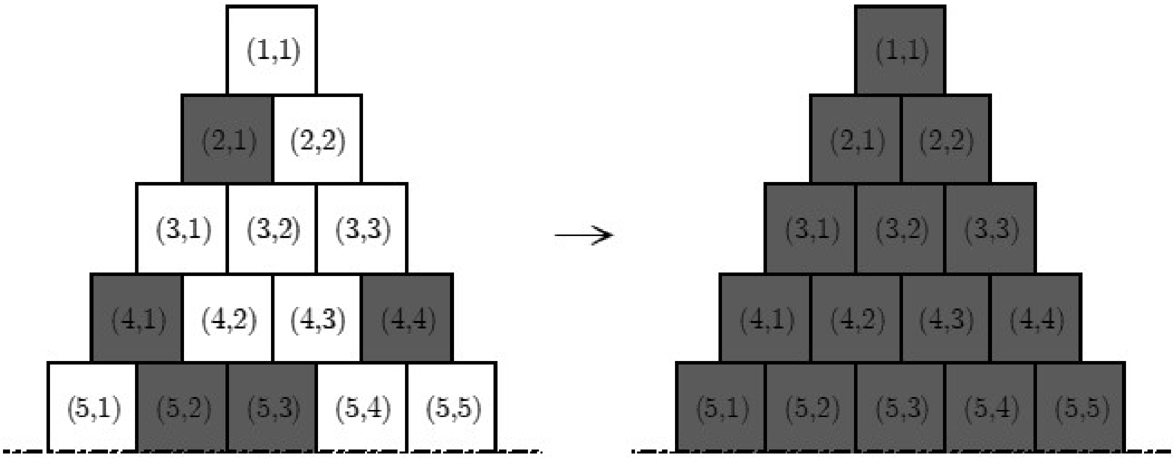
	Input	Output
1	5 3 4 1 5 4 5 5	3



	Input	Output
2	5 2 4 1 5 3	4



3	5 5 2 1 4 1 4 4 5 2 5 3	10



SUBTASKS

For all cases:

$1 \le N \le 2 \times 10^9$

$1 \le K \le \min(\frac{N \times (N+1)}{2}, 1000)$

	Points	Constraints
1	5	$K = 1$
2	9	$K = 2$
3	5	$N = 2$
4	16	$N \le 20$
5	19	$N \le 3000$
6	21	$N \le 10^6$
7	25	No additional constraints

HINT

64-bit integer type is needed.