Mock ZCO #1 (by saarang123)

A. Error Bottles

1 second, 256 megabytes

Paras is going out for a hike and wants to carry exactly L litres of water. However, he has only 2 bottles with capacity A and B litres respectively. Both bottles are empty initially. Water shortage is a crisis in today's world so Paras's mom lets him perform the following operations **atmost** K times:

- Fill either of the bottles completely

- Empty any of the bottles

- Pour the contents of one bottle into the other, stopping when the former becomes empty or the latter becomes full (whichever of these happens first).

While performing the operations, Paras realized he may not be able to achieve exactly L litres. Since Paras is too busy training for IOI, he requests you help him find the **minimum error** between the amount he can achieve and the amount Paras wants to achieve. That is, if the total water in both the bottles at the end of the operations is L', compute the minimum value of $|L - L'|$

Input
The first, and only line of input, contains A, B, K, L.

$1 \leq A, B \leq 10^5$

$1 \leq K \leq 10^5$

$1 \leq L \leq 2 \times 10^5$

Output
Output the minimal error Paras can produce. (Note: In real life, Paras is too good to make any errors!)

Scoring
The input is divided into multiple subtasks. You will get the points for a subtask if you solve all the testcases for a subtask correctly within the time limit.

- Subtask 1 [3 points]: $K = 1$

- Subtask 2 [8 points]: $A = B$

- Subtask 3 [14 points]: $1 \leq K \leq 8$

- Subtask 4 [21 points]: $1 \leq A, B, K \leq 100$

- Subtask 5 [26 points]: $1 \leq A, B, K \leq 1000$

- Subtask 6 [28 points]: Original Constraints

input
14 50 2 32
output
18

In two steps Paras can be left with the following amounts in his bottles:

$(0, 0) = 0$ litres, $(14, 0) = 14$ litres, $(0, 50) = 50$ litres, $(0, 14) = 14$ litres, $(14, 36) = 50$ litres, $(14, 50) = 64$ litres.

The closest we can come to 32 litres is 14 for a difference of 18. It would require an extra step to pour out the first pail to end up with $(0, 36)$.

B. Problem Setting Challenge

3 seconds, 256 megabytes

Saarang and Shiven have each set N problems. Each of the $2N$ problems have difficulty rating according to Saarang and a difficulty rating according to Shiven (they may be different).

Saarang wants to challenge Shiven with one of his problems. If Shiven receives a problem from Saarang, he will want to challenge Saarang with a problem as well. Since Shiven wants to appear as a strong problem setter, he will pick a problem **atleast as difficult** as the problem he received (according to Shiven) but no more than K units harder. If Shiven cannot give Saarang such a problem, he shall quit problemsetting!

If Shiven does return Saarang a problem, Saarang will similarly try to give Shiven a problem which is at least as hard but no more than K units harder (according to Saarang). If this is impossible, Saarang too will quit! Otherwise, he continues to give Shiven a problem. The cycle will continue until one of the problem setters quit, an unhappy outcome, or one of the problem setters receive a problem which he accords a difficulty value of 0, in which case the exchange shall end and both problem setters will be happy.

Note that a problem may not be given twice, nor can either setter return a problem given to him.

For each of the N problems Saarang could select as his initial challenge to Shiven, determine the **minimum number** of problems that could possibly be exchanged before the setters are happy. If there is no possible happy outcome, output -1.

Input
The first line of input consists of 2 integers - N and K.

The next $2N$ lines contain 2 space-separated integers each, respectively denoting the difficulty value of a particular problem according to Saarang, and the value of that problem according to Shiven.

The first N lines refer to Saarang's problems, and the remaining N lines refer to Shiven's problems.

$1 \leq N \leq 10^5$

$1 \leq K \leq 10^9$

All difficulty values are in the range $[0, 10^9]$.

Output
There should be N lines in the output.

Line i should contain a single integer: the minimum number of problems that could be exchanged in a happy problem challenge started with Saarang's problem i. If no problem exchange starting with problem i is happy, then line i should contain the single integer -1 instead.

Scoring
The input is divided into multiple subtasks. You will get the points for a subtask if you solve all the testcases for a subtask correctly within the time limit.

- Subtask 1 [6 points]: All difficulty values are in the range $[1, 10^9]$.

- Subtask 2 [12 points]: $K = 0$ and $1 \leq N \leq 2000$

- Subtask 3 [8 points]: $K = 0$ and all $4N$ difficulty values are distinct.

- Subtask 4 [14 points]: $K = 0$

- Subtask 5 [13 points]: $1 \leq N \leq 2000$

- Subtask 6 [17 points]: There is exactly 1 problem which has a difficulty rating of 0.

- Subtask 7 [8 points]: All of Shiven's difficulty ratings are 0 (because he is a geniosity :o)

- Subtask 8 [22 points]: Original Constraints

input
2 1 1 1 5 0 4 2 1 4

output
3 1

If Saarang challenges Shiven with problem 1 of difficulty 1 (according to Shiven), Shiven will retaliate with his problem 1 which has difficulty 2 according to Shiven. ($2 - 1 \leq K$). Saarang has now received a problem of difficulty 4 according to him and will return his problem 2 which has difficulty 5 ($5 - 4 \leq K$). Since Shiven has now received a problem with 0 difficulty in his eyes, both setters are happy. Hence the answer is 3 for this case.

If Saarang challenges Shiven with problem 2 of difficulty 0 (according to Shiven), both setters will be happy. Hence the minimum number is 1 for this case.