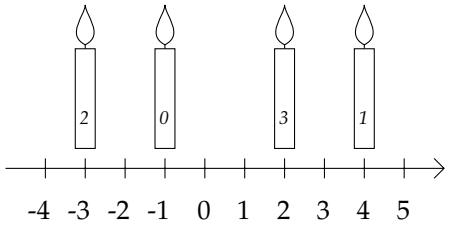
Candles

Mouse Binna loves candles and has decided to use them instead of lightbulbs in her room: they're prettier and smell good. She's placed N candles along a linear wall. The i-th candle is placed exactly x_i meters to the right of the middle of the wall. There may be several candles on the same coordinate, and x_i is negative if the i-th candle is on the left. In the visualisation below, $N = 4$ and $x_0 = -1, x_1 = 4, x_2 = -3, x_3 = 1$.



The issue is that it takes a lot of time to light the candles when she wakes up. She needs to light K candles to have enough light in the room. She wants to do it as quickly as possible, so she wants to walk as short a distance as possible, starting in front of her bed at coordinate s. She only travels on one axis along the wall, and can light candle i when she stands on the same coordinate x_i. Can you help her determine the shortest distance she needs to walk to light K candles?

Input

On the first line, there are three integers, N, the number of candles, K, the number of candles that Binna wants to light, and s, the coordinate of Binna's bed.

On the second line, there are N integers $x_0, \ldots, x_{N-1}$, the coordinates of each of the N candles.

Output

Print a single integer, the minimal distance Binna has to walk to light K candles starting from s.

Limits

There are five subtasks. In every subtask, $1 \leq N \leq 10^6$, $1 \leq K \leq N$, $-10^9 \leq x_i \leq 10^9$, and $-10^9 \leq s \leq 10^9$.

- In subtask 1, worth 5 points, we have $N = K = 1$.
- In subtask 2, worth 40 points, we have $N \leq 10^4$.
- In subtask 3, worth 10 points, we have $x_i - x_{i-1} = c$ for some integer constant $c \geq 0$ and for all $1 \leq i < N$. Additionally, for some $0 \leq j < N$, $x_j = s$.
- In subtask 4, worth 5 points, we have $x_i - x_{i-1} = c$ for some integer constant $c \geq 0$ and for all $1 \leq i < N$.
- In subtask 5, worth 40 points, there are no further restrictions.

Examples

Input	Output
6 3 9 11 20 6 13 2 16	7

Binna starts from position 9 *and lights candles on positions* 11, 13, *and* 16. *She can move in a straight line and the total distance is* 7.

Input	Output
6 4 0 -3 7 5 -7 2 -8	12

Binna start from position 0*, lights the candle on position* 2*, goes back and lights the candles on positions* -3*,* -7*, and* -8*. The total distance is* 12*.*

Order of Departure

Mouse Stofl had a big and fancy party at his big and fancy house. He invited N of his mouse friends (numbered from 0 to $N-1$) and of course they all showed up. They arrived in a specific order. At the party, there is a big pile of fancy cheese balls. Whenever a mouse arrives at the party, it is gifted one cheese ball by each of its friends that were already at the party at the time of its arrival.

Note that some pairs of Stofl's friends may not be friends yet themselves. Therefore, some mice might not receive cheese balls from everyone that is already at the party. However, they always receive at least one cheese ball, from Mouse Stofl.

Unfortunately, even the best party has to end. However, Mouse Stofl's friends do not want to leave unless the order in which they depart is fair: the mice should depart in the same order in which they arrived.

There is only one problem: all of the mice at the party have long forgotten in which order they arrived! The only thing they (of course) remember is how many cheese balls they were gifted upon their arrival.

Therefore, Mouse Stofl came up with a clever scheme: he can just pretend that he knows the order in which his friends arrived. He only has to make sure that, had his friends actually arrived in that order, they would have received the correct number of cheese balls.

Given the number of cheese balls received by each of Mouse Stofl's friends, determine a valid order in which they could have arrived.

Input

The first line of the input contains a single integer N – the number of Stofl's friends at the party.

The second line of the input contains N integers between 1 and N, the i-th integer is the number of cheese balls received by the i-th friend.

The third line of the input contains a single integer M – the number of friendships among Mouse Stofl's friends.

The j-th of the next M lines of the input contains two integers a_j and b_j ($0 \leq a_j < b_j < N$), indicating that Mouse a_j and Mouse b_j are friends with each other. Each pair (a_j, b_j) appears at most once.

Note that Mouse Stofl is friends with all other mice, but those friendships do not appear in the input.

The final line of the input contains a single integer L that is either 0 or 1 (explained in the output section below).

Output

Print a single line containing the first N non-negative integers in some order in which Stofl's friends may have arrived at the party. If $L = 0$ and there are multiple solutions, you may print any of them. However, if $L = 1$, you will only score points if your solution is *lexicographically minimal*.

(A valid solution $p_0, p_1, \ldots, p_{N-1}$ is said to be lexicographically minimal if for any other valid solution $q_0, q_1, \ldots, q_{N-1}$ there is some index i such that for all indices $j < i$, we have $p_j = q_j$ and $p_i < q_i$. I.e., you should output the *smallest* sequence, where sequences are compared according to the *first index where they differ*.)

Limits

There are 5 subtasks with the following point distribution:

- In subtask 1, worth 30 points, we have $1 \leq N \leq 1\,000$, $M = N(N-1)/2$ and you may print any solution (i.e., $L = 0$).

- In subtask 2, worth 20 points, we have $1 \leq N \leq 1\,000$, $0 \leq M \leq N(N-1)/2$ and you may print any solution (i.e., $L = 0$).
- In subtask 3, worth 20 points, we have $1 \leq N \leq 100\,000$, $0 \leq M \leq 500\,000$ and you may print any solution (i.e., $L = 0$).
- In subtask 4, worth 15 points, we have $1 \leq N \leq 1\,000$, $0 \leq M \leq N(N-1)/2$ and your solution must be lexicographically minimal (i.e., $L = 1$).
- In subtask 5, worth 15 points, we have $1 \leq N \leq 100\,000$, $0 \leq M \leq 500\,000$ and your solution must be lexicographically minimal (i.e., $L = 1$).

In all test cases, it is guaranteed that there is at least one solution. In particular, the number of cheese balls received by each of Stofl's friends is between 1 and the total number of their friends attending the party.

Examples

Input	Output
5 1 4 3 5 2 10 0 1 0 2 0 3 0 4 1 2 1 3 1 4 2 3 2 4 3 4 0	0 4 2 1 3

This sample satisfies the constraints for subtask 1.

According to the input, Mouse 0 got 1 cheese ball, Mouse 1 got 4 cheese balls, Mouse 2 got 3 cheese balls, Mouse 3 got 5 cheese balls and Mouse 4 got 2 cheese balls.

Furthermore, all pairs of mice are friends.

We have $L = 0$, so the solution does not necessarily have to be lexicographically minimal.

The output describes a scenario where Mouse 0 arrives first, then Mouse 4, then Mouse 2, then Mouse 1, and finally Mouse 3.

This output is a valid solution, because the mice would indeed obtain the correct number of cheese balls, had they arrived in this order:

When Mouse 0 arrives, it gets 1 cheese ball from Mouse Stofl.

When Mouse 4 arrives, it gets 2 cheese balls, one from Mouse Stofl and one from Mouse 0.

When Mouse 2 arrives, it gets 3 cheese balls from Mouse Stofl, Mouse 0, and Mouse 4.

When Mouse 1 arrives, it gets 4 cheese balls from Mouse Stofl, Mouse 0, Mouse 4 and Mouse 2.

When Mouse 3 arrives, it gets 5 cheese balls from everyone else.

Input	Output
5 1 4 2 3 1 6 2 3 0 1 1 2 0 2 1 4 3 4 0	0 4 2 1 3

This sample satisfies the constraints for subtasks 2 and 3.

According to the input, Mouse 0 got 1 *cheese ball, Mouse 1 got* 4 *cheese balls, Mouse 2 got* 2 *cheese balls, Mouse 3 got* 3 *cheese balls, and Mouse 4 got* 1 *cheese ball.*

In this example, not all pairs of mice are friends: Mouse 0 is not friends with Mouse 3 and Mouse 4. Furthermore, Mouse 1 is not friends with Mouse 3 and Mouse 2 is not friends with Mouse 4.

We still have $L = 0$*, so the solution does not necessarily have to be lexicographically minimal.*

The example output is the same as for the previous example.

This output is a valid solution, because the mice would indeed obtain the correct number of cheese balls, had they arrived in this order:

When Mouse 0 arrives, it gets 1 *cheese ball from Mouse Stofl.*

When Mouse 4 arrives, it gets 1 *cheese ball, from Mouse Stofl (it is not friends with Mouse* 0*).*

When Mouse 2 arrives, it gets 2 *cheese balls from Mouse Stofl and Mouse* 0 *(it is not friends with Mouse* 4*).*

When Mouse 1 arrives, it gets 4 *cheese balls from Mouse Stofl, Mouse* 0*, Mouse* 4*, and Mouse* 2*.*

When Mouse 3 arrives, it gets 3 *cheese balls from Mouse Stofl, Mouse* 2*, and Mouse* 4 *(it is not friends with Mouse* 0 *and Mouse* 1*).*

Input	Output
5 1 4 2 3 1 6 2 3 0 1 1 2 0 2 1 4 3 4 1	0 2 4 1 3

This sample satisfies the constraints for subtasks 4 and 5.

The input is the same as for the previous example, except now $L = 1$. This means the output has to be lexicographically minimal.

Note that if we swap the order of arrival for Mouse 2 and Mouse 4 in the output from the previous sample, this does not change the number of cheese balls obtained by any mouse. This is because they arrive right after one another and they do not gift a cheese ball to each other as they are not friends.

Therefore, this new order is also valid. In fact, for this example, this is the lexicographically minimal solution. In subtasks 4 and 5, this is the only output that will score any points.

Because $L = 1$, this sample is scored like subtasks 4 and 5, so be aware that you may fail this sample if your solution cannot solve those subtasks.

Timber!

Mouse Binna joined the **F**ellowship of **E**thical **L**umberjacks and **L**umberjanes (FELL), an organization concerned about environmentally friendly practices for felling trees.

The key beliefs of the FELL-ows (which is how the members of FELL like to call themselves) are:

1. If you are felling trees, bring back at least s wood. Making large harvests means the ecosystem is disturbed very rarely.
2. Try to space out the felled trees as much as possible. Let d be the minimal distance between two trees that have been felled. Find the largest d possible.
3. When a tree falls, you must yell "timber!"

Mouse Binna knows everything about point 3, but can't seem to figure out how to satisfy points 1 and 2. So she would like you to help her.

The forest can be modelled as n trees on a line. You know their heights $h_0, h_1, \ldots, h_{n-1}$ and are given a parameter s. You can either fell a tree or let it stand. The sum of heights of all trees you have felled should be at least s. Compute the largest d such that any two felled trees are at least d apart and their sum of heights is at least s.

Input

The first line of the input contains two integers n and s, the number of trees and the minimal sum of heights of the trees that are felled.

The second line contains n numbers $h_0, h_1, \ldots, h_{n-1}$, the heights of the trees.

Output

If it is impossible to get s amount of tree out of the forest, print a single line containing "`Impossible`".

If it is possible to reach s with zero or one trees, print a single line containing "`Infinity`".

Otherwise print a single line with an integer d – the largest d such that it is possible to fell trees that are at least d apart and their sum is at least s.

Limits

There are 7 subtasks. In all subtasks, we have $1 \leq h_i \leq 10^9$ and $0 \leq s \leq 10^{16}$.

- In subtask 1, worth 10 points, we have $1 \leq n \leq 10$.
- In subtask 2, worth 10 points, we have $1 \leq n \leq 10\,000$ and all trees have distinct height and are sorted increasingly (i.e. $h_0 < h_1 < \ldots < h_{n-1}$).
- In subtask 3, worth 10 points, we have $1 \leq n \leq 10\,000$.
- In subtask 4, worth 10 points, we have $1 \leq n \leq 10^5$ and all trees have equal height (i.e. $h_0 = h_1 = \ldots = h_{n-1}$).
- In subtask 5, worth 20 points, we have $1 \leq n \leq 10^5$ and all trees have distinct height and are sorted increasingly (i.e. $h_0 < h_1 < \ldots < h_{n-1}$).
- In subtask 6, worth 20 points, we have $1 \leq n \leq 10^5$.
- In subtask 7, worth 20 points, we have $1 \leq n \leq 10^6$.

Examples

Input	Output
8 14 4 5 2 2 3 1 1 5	2

Felling the trees at positions $[0, 2, 4, 7]$ *yields exactly* $4 + 2 + 3 + 5 = 14$ *wood.*

Input	Output
7 27 1 2 3 5 8 13 21	3

Felling the trees at positions $[0, 3, 6]$ *yields exactly* $1 + 5 + 21 = 27$ *wood. This would be a valid input for subtasks 2 and 5.*

Input	Output
11 13 5 6 1 1 4 1 1 1 1 4 1	4

Input	Output
5 6 5 5 5 5 5	4

Input	Output
5 6 1 1 1 1 1	Impossible

Even if you fell all trees there is not enough wood.

Input	Output
5 5 5 5 5 5 5	Infinity

It is already enough to fell one tree, therefore the distance to another felled tree is infinity.

Input	Output
4 4 1 1 1 1	1

RPG Gameplay

Mouse Stofl has been playing a lot of RPGs (computer games where you control a character) recently. He has invested hundreds of hours into this game called "Effect of the Primordial Gods". All of Stofl's friends are worried that Stofl might not leave his room anymore. Can you help Stofl finish the final level of the game so he can leave his room again?

In the game, Stofl has two attacks: He can attack using his left arm and using his right arm. If he uses his left arm, he needs to wait l seconds until he can use it to attack again. If he uses his right arm, he needs to wait r seconds until he can use it again. (This is usually called a "cooldown".) Note that the two arms don't affect each other, meaning that even if Stofl is waiting for his right arm to be ready again, he can use his left arm if it's ready, and vice-versa.

Stofl currently wants to clear a level with n monsters, where the i-th monster appears t_i seconds after he starts the level (the monsters are given in non-decreasing order of time). When a monster appears, Stofl can hit it with one of his two arms. If he does, the monster dies. Otherwise, the monster hits Stofl and then disappears.

Stofl starts the room with some (positive integer) amount of hitpoints h, symbolising how much health he has. Every time a monster hits Stofl, his hitpoints get reduced by 1. If his hitpoints reach 0, Stofl dies.

What is the minimal number of hitpoints (the minimal value of h) Stofl needs to have in the beginning of the level for him to survive it?

Remark: Note that the memory limit is a bit lower than usual.

Input

On the first line, there are 3 integers n, l and r, denoting the number of monsters, and the cooldown of his left and his right attack. On the second line, there are n integers t_i, denoting the time the i-th monster appears. The monsters are given in non-decreasing order of time (meaning that if $i < j$ then $t_i \leq t_j$).

Output

Print a single integer, the minimal amount of hitpoints Stofl needs in the beginning to survive the level.

Limits

There are five subtasks. In every subtask, $1 \leq n \leq 10^6$, $0 \leq t_0 \leq t_1 \leq \ldots \leq t_{n-1} \leq 10^9$, and $1 \leq l, r \leq 10^9$.

- In subtask 1, worth 10 points, we have $t_i < 10^9$ and $r = 10^9$.
- In subtask 2, worth 15 points, we have $l = r$.
- In subtask 3, worth 10 points, we have $n \leq 16$.
- In subtask 4, worth 20 points, we have $n \leq 3 \cdot 10^3$.
- In subtask 5, worth 15 points, we have $n \leq 2 \cdot 10^4$.
- In subtask 6, worth 30 points, there are no further constraints.

Examples

Input	Output
6 3 5 1 2 4 5 6 11	2

This sample satisfies the constraints for subtasks 3, 4, 5, and 6.

If Stofl starts with 2 *hitpoints, he can use his right arm to hit the first, the fifth and the sixth monster, and his left arm to hit the second and the fourth monster. Then he only gets hit by the third monster, reducing his hitpoints to* 1*, but that doesn't kill him. Thus Stofl survives.*

Input	Output
5 5 3 1 2 4 7 14	1

This sample satisfies the constraints for subtasks 3, 4, 5, and 6.

If Stofl starts with 1 *hitpoint, he can use his right arm to hit the first, the third and the fifth monster, and his left arm to hit the second and the fourth monster. Then Stofl doesn't get hit by any monster and survives.*

Input	Output
7 4 1000000000 0 4 8 10 12 14 16	2

This sample satisfies the constraints for subtasks 1, 3, 4, 5, and 6.

Input	Output
9 4 4 1 2 3 6 7 7 9 11 13	4

This sample satisfies the constraints for subtasks 2, 3, 4, 5, and 6.