

Problem A. BFS by Foxpower

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 128 mebibytes

Fox Ciel went to JAG Kingdom by bicycle, but she forgot a place where she parked her bicycle. So she needs to search it from a bicycle-parking area before returning home.

The parking area is formed as a unweighted rooted tree T with n vertices, numbered 1 through n . Each vertex has a space for parking one or more bicycles. Ciel thought that she parked her bicycle near the vertex 1, so she decided to search it from there by the breadth-first search. That is, she searches it at the vertices in the increasing order of their distances from the vertex 1. If multiple vertices have the same distance, she gives priority to the vertices in the order of searching at their parents. If multiple vertices have the same parent, she searches at the vertex with minimum number at first.

Unlike a computer, she can't go to a next vertex by random access. Thus, if she goes to the vertex j after the vertex i , she needs to walk the distance between the vertices i and j . BFS by fox power perhaps takes a long time, so she asks you to calculate the total moving distance in the worst case starting from the vertex 1.

Input

The first line of the input contains an integer n ($1 \leq n \leq 10^5$), which is the number of vertices on the unweighted rooted tree T . The second line contains $n - 1$ integers p_i ($1 \leq p_i < i$), which are the parent of the vertex i . The vertex 1 is a root node, so p_1 does not exist.

Output

Print the total moving distance in the worst case in one line.

Examples

standard input	standard output
4 1 1 2	6
4 1 1 3	4
11 1 1 3 3 2 4 1 3 2 9	25

Problem B. Cube Coloring

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 128 mebibytes

Great painter Cubic is famous for using cubes for his art. Now, he is engaged in a new work of art. He tries to form an $X \times Y \times Z$ rectangular parallelepiped by arranging and piling up many $1 \times 1 \times 1$ cubes such that the adjacent surfaces of cubes are fully shared.

Of course, he won't finish his work only arranging and piling up cubes. The position of each cube is denoted by $(0, 0, 0)$ through $(X - 1, Y - 1, Z - 1)$ as by the ordinary coordinate system, and Cubic calls the cube (A, B, C) the origin cube. Then, he paints a pattern on the rectangular parallelepiped with different colors according to the distance between each cube and the origin cube. He paints all cubes regardless of whether or not a cube is visible externally. This is his artistic policy. He uses Manhattan distance as the distance between cubes. The Manhattan distance between two cubes (x_1, y_1, z_1) and (x_2, y_2, z_2) is defined as $|x_1 - x_2| + |y_1 - y_2| + |z_1 - z_2|$.

On the current work, Cubic decides to use N colors, which are numbered from 1 to N . He paints a cube with the $(i + 1)$ -th color if the distance D between the cube and the origin cube satisfies $D = i(\text{mod}N)$.

Cubic wants to estimate the consumption of each color in order to prepare for the current work. He asks a great programmer, you, to write a program calculating the number of cubes that will be painted by each color.

Input

The input contains seven integers X, Y, Z, A, B, C , and N . All integers are in one line and separated by a single space. Three integers X, Y , and Z ($1 \leq X, Y, Z \leq 10^6$) represent the length of each side of the rectangular parallelepiped that Cubic tries to form. Three integers A, B , and C ($0 \leq A < X, 0 \leq B < Y, 0 \leq C < Z$) represent the position of the origin cube. The number of kinds of colors is denoted by N ($1 \leq N \leq 1,000$).

Output

The output contains N integers in one line. The i -th integer ($1 \leq i \leq N$) represents the number of the cubes that will be painted by the i -th color. All integers must be separated by a single space.

Examples

standard input	standard output
2 2 2 0 0 0 5	1 3 3 1 0
4 3 3 1 1 1 3	13 10 13
2000 2000 2000 1000 1000 1000 1	8000000000

Problem C. Decoding Ancient Messages

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 128 mebibytes

Professor Y's work is to dig up ancient artifacts. Recently, he found a lot of strange stone plates, each of which has N^2 characters arranged in an $N \times N$ matrix. Further research revealed that each plate represents a message of length N . Also, the procedure to decode the message from a plate was turned out to be the following:

1. Select N characters from the plate one by one so that any two characters are neither in the same row nor in the same column.
2. Create a string by concatenating the selected characters.
3. Among all possible strings obtained by the above steps, find the lexicographically smallest one. It is the message represented by this plate.

NOTE: The order of the characters is defined as the same as the order of their ASCII values (that is, $A < B < \dots < Z < a < b < \dots < z$).

After struggling with the plates, Professor Y gave up decoding messages by hand. You, a great programmer and Y's old friend, was asked for a help. Your task is to write a program to decode the messages hidden in the plates.

Input

The first line contains an integer N ($1 \leq N \leq 50$). Each of the subsequent N lines contains a string of N characters. Each character in the string is an uppercase or lowercase English letter.

Output

Print the message represented by the plate in a line.

Examples

standard input
3
aab
czc
baa
standard output
aac

standard input	
36	<pre> ii ii ii ii ii ii ii iiiiiiiiiiiQiiiiiiiiiiiiiiiiiiiQiiiiiiii iiiiiiiiiiiQQQiiiiiiiiiiiQQQiiiiiiii iiiiiiiiiiiQQQQiiiiiiiiiiQQQQiiiiiiii iiiiiiiiiiiQQQQQQQQQQQQQQQiiiiiiii iiiiiiiiiiiQQQQQQQQQQQQQQQiiiiiiii iiiiiiiiiiiQQQQQQQQQQQQQQQiiiiiiii iiiiiiiiiiiQQQQQQQQQQQQQQQiiiiiiii iiiiiiiiiiiQQQQQQQQQQQQQQQiiiiiiii iiiiiiiiiiiQQQQQQQQQQQQQQQiiiiiiii iiiiiiiQiiiiQQQQQQQQQQQQQQQQQiiiiQiiii iiiiiiiQQiQQQQQQQQQQQQQQQQQQQiiQQiiii iiiiiiiQQQQQQQQQQQQQQQQQQQQQiiQQiiii iiiiiiiiQQQQQQQQQQQQQQQQQQQQQiiQQiiii iiiiiiiiQQQQQQQQQQQQQQQQQQQQQiiiiiiii iiiiiiiiQQQQQQiiQQQQQQQQiiQQQQQiiiiii iiiiiiiQQQQQQQiiiiQQQQQiiiiQQQQQiiiiii iiiiiiQQQQQQQQQiiiiQQQQQiiiiQQQQQiiiiii iiiiiiQQQQQQQQQiiiiQQQQQiiQQQQQQQQiQiii iiiQQQQQQQiiQiiiiQQQQQiiQiiQQQQQQQii iQQQQQQQQQiiiiiiQQQQQQQiiiiiiQQQQQQQii iiQQQQQQQiiiiiiQQQQQQQiiiiiiiiiQQQiiii iQQQiiiiiiiiiiiQQQQQQQQQiiiiiiiiQQQii iiiiiiiiiiiiiiiiiiQQQQQQQQQiiiiiiiiiiii iiiiiiiiiiiiiiiiiiQQQQQQQQQiiiiiiiiiiii iiiiiiiiiiiiiiiiiiQQQQQQQQQiiiiiiiiiiii iiiiiiiiiiiiiiiiiiQQQQQQQQQiiiiiiiiiiii iiiiiiiiiiiiiiiiiiiiQQQQQQQQQiiiiiiiiiiii iiiiiiiiiiiiiiiiiiiiQQQQQQQQQiiiiiiiiiiii iiiiiiiiiiiiiiiiiiiiQQQQQQQQQiiiiiiiiiiii iiiiiiiiiiiiiiiiiiiiQQQQQQQQQiiiiiiiiiiii iiiiiiiiiiiiiiiiiiiiQQQQQQQQQiiiiiiiiiiii iiiiiiiiiiiiiiiiiiiiQQQQQQQQQiiiiiiiiiiii </pre>
standard output	
<pre> QQQQQQQQQQQQQQQQQQQQQQQQQQQiiiiiiiiiii </pre>	
standard input	
Acm	
aCm	
acM	
standard output	
ACM	

Problem D. LR

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 128 mebibytes

JAG Kingdom will hold a contest called ICPC (Interesting Contest for Producing Calculation).

At the contest, you are given a string composed of ?-s and usable characters. You should replace all “?” in the string with the usable characters to make the string valid mathematical expression, before submitting it. The usable characters are ‘L’, ‘R’, ‘(’, ‘)’, ‘,’, and digits from 0 to 9.

For example, suppose that you are given the string “R(??3,??1?78??1?)?”, then you can submit “R(123,L(1678,213))” as an example.

The submitted string will be scored as follows.

Let L and R be functions defined by $L(x, y) = x$, $R(x, y) = y$, where x and y are non-negative integers.

The submitted string will be regarded as a mathematical expression, whose value will be the score. For example, the score of the string “R(123,L(1678,213))” is $R(123, L(1678, 213)) = R(123, 1678) = 1678$.

If the string cannot be evaluated as a mathematical expression about the functions L and R, the string will be rejected. For example, “R”, “R(3)”, “R(3,2”, “R(3,2,4)” and “LR(3,2)” are all invalid.

And strings that contain numbers with extra leading zeros, will be rejected. For example, “R(04,18)” is invalid, while “R(0,18)” is valid.

The winner of the contest will be the person getting the highest score. Your friend Jagger, who is going to join the contest, wants to be the winner. You are asked by Jagger to make the program finding the possible highest score for the input string.

Input

The input contains one string in a line, whose length N is between 1 and 50, inclusive.

You can assume that each element in the string is one of ‘?’, ‘L’, ‘R’, ‘(’, ‘)’, ‘,’, or digits from 0 to 9.

Output

Display the possible highest score in a line for the given string.

If it’s impossible to get valid strings for the input string, print “invalid” in a line.

Examples

standard input
R?????,2?)
standard output
29

JAG Spring 2014 Contest
Japanese Alumni Group, April 2014

[illegible]

Problem E. Parentheses

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 128 mebibytes

You are given n strings $str_1, str_2, \dots, str_n$, each consisting of ‘(’ and ‘)’. The objective is to determine whether it is possible to permute the n strings so that the concatenation of the strings represents a valid string.

Validity of strings are defined as follows:

- The empty string is valid.
- If A and B are valid, then the concatenation of A and B is valid.
- If A is valid, then the string obtained by putting A in a pair of matching parentheses is valid.
- Any other string is not valid.

For example, “()()” and “(())” are valid, while “())” and “((()” are not valid.

Input

The first line of the input contains an integer n ($1 \leq n \leq 100$), representing the number of strings. Then n lines follow, each of which contains str_i ($1 \leq |str_i| \leq 100$). All characters in str_i are ‘(’ or ‘)’.

Output

Output a line with “Yes” (without quotes) if you can make a valid string, or “No” otherwise.

Examples

standard input	standard output
3 ()()(())()()()()())	Yes
2))()())((()((No

Problem F. Polygon Guards

Input file: *standard input*
Output file: *standard output*
Time limit: 3 seconds
Memory limit: 128 mebibytes

You are an IT system administrator in the Ministry of Defense of Polygon Country.

Polygon Country's border forms the polygon with N vertices. Drawn on the 2D-plane, all of its vertices are at the lattice points and all of its edges are parallel with either the x -axis or the y -axis.

In order to prevent enemies from invading the country, it is surrounded by very strong defense walls along its border. However, on the vertices, the junctions of walls have unavoidable structural weaknesses. Therefore, enemies might attack and invade from the vertices.

To observe the vertices and find an invasion by enemies as soon as possible, the ministry decided to hire some guards. The ministry plans to locate them on some vertices such that all the vertices are observed by at least one guard. A guard at the vertex A can observe a vertex B if the entire segment connecting A and B is inside or on the edge of Polygon Country. Of course, guards can observe the vertices they are located on. And a guard can observe simultaneously all the vertices he or she can observe.

To reduce the defense expense, the ministry wants to minimize the number of guards. Your task is to calculate the minimum number of guards required to observe all the vertices of Polygon Country.

Input

The first line contains an even integer N ($4 \leq N < 40$). The following N lines describe the vertices of Polygon Country. Each of the lines contains two integers, X_i and Y_i ($1 \leq i \leq N$, $|X_i| \leq 1,000$, $|Y_i| \leq 1,000$), separated by one space. The position of the i -th vertex is (X_i, Y_i) .

If i is odd, $X_i = X_{i+1}$, $Y_i \neq Y_{i+1}$. Otherwise, $X_i \neq X_{i+1}$, $Y_i = Y_{i+1}$. Here, we regard that $X_{N+1} = X_1$, and $Y_{N+1} = Y_1$. The vertices are given in counterclockwise order under the coordinate system that the x -axis goes right, and the y -axis goes up. The shape of Polygon Country is simple. That is, each edge doesn't share any points with other edges except that its both end points are shared with its neighbor edges.

Output

Print the minimum number of guards in one line.

Examples

standard input	standard output
8 0 2 0 0 2 0 2 1 3 1 3 3 1 3 1 2	1
12 0 0 0 -13 3 -13 3 -10 10 -10 10 10 -1 10 -1 13 -4 13 -4 10 -10 10 -10 0	2

Problem G. Proportional Representation

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 128 mebibytes

An election selecting the members of the parliament in JAG Kingdom was held. The only system adopted in this country is the party-list proportional representation. In this system, each citizen votes for a political party, and the number of seats a party wins will be proportional to the number of votes it receives. Since the total number of seats in the parliament is an integer, of course, it is often impossible to allocate seats exactly proportionally. In JAG Kingdom, the following method, known as the D'Hondt method, is used to determine the number of seats for each party.

Assume that every party has an unlimited supply of candidates and the candidates of each party are ordered in some way. To the y -th candidate of a party which received x votes, assign the value x/y . Then all the candidates are sorted in the decreasing order of their assigned values. The first T candidates are considered to win, where T is the total number of seats, and the number of seats a party win is the number of its winning candidates.

The table below shows an example with three parties. The first party received 40 votes, the second 60 votes, and the third 30 votes. If the total number of seats is $T = 9$, the first party will win 3 seats, the second 4 seats, and the third 2 seats.

# of votes	$x/1$	$x/2$	$x/3$	$x/4$	$x/5$	$x/6$	$x/7$
40	40	20	13.33...	10	8	6.66...	5.71...
60	60	30	20	15	12	10	6.57...
30	30	15	10	7.5	6	5	3.28...

When selecting winning candidates, ties are broken by lottery; any tied candidates will have a chance to win. For instance, in the example above, if $T = 5$ then two candidates tie for the value 20 and there are two possible outcomes:

- The first party wins 2 seats, the second 2 seats, and the third 1 seat.
- The first party wins 1 seat, the second 3 seats, and the third 1 seat.

You have just heard the results of the election on TV. Knowing the total number of valid votes and the number of seats each party won, you wonder how many votes each party received.

Given N , M , and S_i ($1 \leq i \leq M$), denoting the total number of valid votes, the number of parties, and the number of seats the i -th party won, respectively, your task is to determine for each party the minimum and the maximum possible number of votes it received. Note that for some cases there might be no such situation with the given N , M , and S_i .

Input

The first line of the input contains two integers N ($1 \leq N \leq 10^9$) and M ($1 \leq M \leq 3 \cdot 10^4$), where N is the total number of valid votes and M is the number of parties. M lines follow, the i -th of which contains a single integer S_i ($0 \leq S_i \leq 3 \cdot 10^4$), representing the number of seats the i -th party won. You can assume that there exists i with $S_i \neq 0$.

Output

If there is no situation with the given N , M , and S_i , display the word “impossible”. Otherwise, output M lines, each containing two integers. The first integer and the second integer in the i -th line should be the minimum and the maximum possible number of votes the i -th party received, respectively.

Examples

standard input	standard output
10 2 2 1	5 7 3 5
5 6 0 2 0 2 6 0	0 0 1 1 0 0 1 1 3 3 0 0
2000 5 200 201 202 203 204	396 397 397 399 399 401 401 403 403 404
15 10 1 1 1 1 1 1 1 1 1 13	impossible
1000000000 9 12507 16653 26746 21516 29090 10215 28375 21379 18494	67611619 67619582 90024490 90033301 144586260 144597136 116313392 116323198 157257695 157269050 55221291 55228786 153392475 153403684 115572783 115582561 99976756 99985943

Problem H. RLE Replacement

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 128 mebibytes

In JAG Kingdom, ICPC (Intentionally Compressible Programming Code) is one of the common programming languages. Programs in this language only contain uppercase English letters and the same letters often appear repeatedly in ICPC programs. Thus, programmers in JAG Kingdom prefer to compress ICPC programs by Run Length Encoding in order to manage very large-scale ICPC programs.

Run Length Encoding (RLE) is a string compression method such that each maximal sequence of the same letters is encoded by a pair of the letter and the length. For example, the string “RRRRLEEE” is represented as “R4L1E3” in RLE.

Now, you manage many ICPC programs encoded by RLE. You are developing an editor for ICPC programs encoded by RLE, and now you would like to implement a replacement function. Given three strings A , B , and C that are encoded by RLE, your task is to implement a function replacing the first occurrence of the substring B in A with C , and outputting the edited string encoded by RLE. If B does not occur in A , you must output A encoded by RLE without changes.

Input

The input consists of three lines. The lines represent strings A , B , and C that are encoded by RLE, respectively. Each of the lines has the following format: $c_1l_1c_2l_2\ldots c_nl_n\$$.

Each c_i ($1 \leq i \leq n$) is an uppercase English letter and l_i ($1 \leq i \leq n, 1 \leq l_i \leq 10^8$) is an integer which represents the length of the repetition of c_i . The number n of the pairs of a letter and an integer satisfies $1 \leq n \leq 10^3$. A terminal symbol “\$” indicates the end of a string encoded by RLE. The letters and the integers are separated by a single space. It is guaranteed that $c_i \neq c_{i+1}$ holds for any $1 \leq i \leq n - 1$.

Output

Replace the first occurrence of the substring B in A with C if B occurs in A , and output the string encoded by RLE. The output must have the same RLE format as the input.

Examples

standard input	standard output
R 100 L 20 E 10 \$ R 5 L 10 \$ X 20 \$	R 95 X 20 L 10 E 10 \$
A 3 B 3 A 3 \$ A 1 B 3 A 1 \$ A 2 \$	A 6 \$

Problem I. Tokyo Olympics Center

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 128 mebibytes

You are participating in the Summer Training Camp for Programming Contests. The camp is held in an accommodation facility called Tokyo Olympics Center. Today is the last day of the camp. Unfortunately, you are ordered to check if all the participants have properly cleaned their rooms.

The accommodation facility is a rectangular field whose height is H and width is W . The field is divided into square cells. The rows are numbered 1 through H from top to bottom and the columns are numbered 1 through W from left to right. The cell in row i , column j is denoted by (i, j) . Two cells are adjacent if they share an edge.

Each cell is called either a wall cell or a floor cell. A wall cell represents a wall which no one can enter. A floor cell is a part of the inside of the accommodation facility. Everybody can move between two adjacent floor cells. The floor cells are divided into several units. Each unit is assigned an uppercase English letter (A, B, C, etc). A floor cell adjacent to exactly one floor cell is called a room. Otherwise the floor cell is called an aisle. For example, the accommodation facility can be shown as the following figure. We denote a wall cell by ‘.’ (single period).

```
.....  
.....AAABBBBBBB.....  
...A.AA.A...B.B..B.  
..AAAAAAAABBBBBBBB.  
...A..A.A.....B....  
.....A.....BBBB.  
....A.AA..C.C...B..  
...AAAAACCCCCBBBB.  
...A..A...C.C...B..  
.....
```

In the above figure, there are 7 rooms in unit A, 4 rooms in unit B, and 4 rooms in unit C.

Because the accommodation facility is too large to explore alone, you asked the other participants of the camp to check the rooms. For simplicity's sake, we call them staffs. Now, there are K staffs at the cell (s, t) . You decided to check the rooms according to the following procedure.

1. First, you assign each staff to some of units. Every unit must be assigned to exactly one staff. Note that it is allowed to assign all the units to one staff or to assign no units to some staffs.
2. Then, each staff starts to check the rooms in the assigned units at the same time. The staffs can move between two adjacent floor cells in T_{move} time. To check the room at (i, j) , the staffs must move to the cell (i, j) and spend T_{check} time there. Each staff first determines the order of the units to check and he or she must check the rooms according to the order. For example, suppose that there is a staff who is assigned units A, C, and E. He may decide that the order is $E \rightarrow A \rightarrow C$. After that, he must check all the rooms in unit E at first. Then, he must check all the rooms in unit A, and so on. The staffs can pass any floor cells.

However, the staffs cannot check rooms that are not assigned to them. Further, the staffs cannot check rooms against the order of units that they have decided.

3. After checking all the assigned rooms, the staffs must return to the cell (s, t) .
4. When every staff returns to the cell (s, t) , the task is done.

Because you do not have enough time before the next contest, you want to minimize the total time to check all the rooms.

Input

The first line of the input contains three integers H , W , and K ($1 \leq H \leq 50$, $1 \leq W \leq 50$, $1 \leq K \leq 12$). The second line contains four integers s , t , T_{move} , and T_{check} ($1 \leq s \leq H$, $1 \leq t \leq W$, $1 \leq T_{move}, T_{check} \leq 10^4$). Each of the following H lines contains exactly W characters. Each character represents a cell in the accommodation facility. The j -th character in the i -th line of these lines is '.' if the cell (i, j) is a wall cell. Otherwise, the character is an uppercase English letter ('A'-'L') representing the unit to which the cell (i, j) belongs.

You can assume that the following conditions are satisfied.

- The number of rooms in each unit is between 1 and 12, inclusive.
- The cell (s, t) is guaranteed to be an aisle.
- Floor cells in each unit are connected.
- Floor cells in the field are connected.
- Every unit contains at least two cells.

Output

Output the minimum required time to check all the rooms in one line.

Examples

standard input	standard output
3 3 1 1 1 10 10 AAA A.. A..	100
3 3 2 1 1 10 10 ABB A.. A..	50

JAG Spring 2014 Contest
Japanese Alumni Group, April 2014

standard input	standard output
5 10 3 3 6 1 100 ...G.H.A.. .AAGAHAABB FFAAAAAAA.. .EEAAADACC ..E...D...	316
10 19 2 6 15 3 10AAABBBBBBB..... ...A.AA.A...B.B..B.. ..AAAAAAAABBBBBBBB.. ...A..A.A.....B.....A.....BBBB..A.AA..C.C...B.. ...AAAAACCCCCBBBB.. ...A..A...C.C...B..	232
27 36 6 24 19 616 1933B.....B..BBB.....BBBB..BBBBB.....BBBB..BBBBBBBBBBBBBBBB..BBBBBBBBBBBBBBBB..BBBBBBBBBBBBBBBB..BBBBBBBBBBBBBBBB..BBBBBBBBBBBBBBBB..BBBBBBBBBBBBBBBB..B...BBBBBBBBBBBBBBBB...B..BB.BBBBBBBBBBBBBBBBBBB..BB..BBBBBBBBBBBBBBBBBBBBBB.BB..BBBBBBBBBBBBBBBBBBBBBB..BBBBBB..BBBBBBB..BBBBB..BBBBBB..BBBBB..BBBBBB..BBBBBBB..BBBBB..BBBBBBB.B.. ...BBBBBBB..B...BBBBB..B..BBBBBBB.. .BBBBBBBB.....BBBBBB.....BBBBBB.. ..BBBBBBB.....BBBBBB.....BBB.. .BBBB.....BBBBBBBBB.....BBB..BBBBBBBBB..BBBBBBBBB..BBBBBBB..BBBBBBB..BBBBBBB..	137071

Problem J. Unfair Game

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 128 mebibytes

Rabbit Hanako and Fox Jiro are great friends going to JAG primary school. Today they decided to play the following game during the lunch break.

This game is played by two players with N heaps of some number of stones. The players alternatively take their turn to play the game. Jiro is a kind gentleman, so he yielded the first turn to Hanako. In each turn, the player must take some stones, satisfying the following conditions:

- If the player is Hanako, she must take between 1 to A stones, inclusive, from a heap.
- If the player is Jiro, he must take between 1 to B stones, inclusive, from a heap.

The winner is the player who takes the last stone. Jiro thinks it is rude to go easy on her because he is a perfect gentleman. Therefore, he does his best. Of course, Hanako also does so.

Jiro is worried that he may lose the game. Being a cadet teacher working at JAG primary school as well as a professional competitive programmer, you should help him by programming. Your task is to write a program calculating the winner, assuming that they both play optimally.

Input

The first line contains three integers N , A , and B . N ($1 \leq N \leq 10^5$) is the number of heaps. A and B ($1 \leq A, B \leq 10^9$) are the maximum numbers of stones that Hanako and Jiro can take in a turn, respectively. Then N lines follow, each of which contains a single integer S_i ($1 \leq S_i \leq 10^9$), representing the number of stones in the i -th heap at the beginning of the game.

Output

Output a line with text “Hanako” if Hanako wins the game or “Jiro” in the other case.

Examples

standard input	standard output
3 5 4 3 6 12	Hanako
4 7 8 8 3 14 5	Jiro