Problem A

Just Terraffic!

Time limit: 3 seconds

The local council is recording traffic flow using a pressure pad laid across the road. The pressure pad tracks whenever the wheels on an axle of a vehicle cross the pressure pad. The only vehicles using the road are cars with two axles. Each vehicle may or may not have a single-axle trailer attached to it. When a car crosses the pressure pad, two times are recorded: one when the front wheels cross and another when the rear wheels cross. If the car is towing a trailer an additional time is recorded when the trailer wheels cross. Given a sequence of times from the recorder, the council wishes to know how many cars without trailers crossed the pad and how many cars with trailers crossed it.

Obviously, there is some ambiguity. For example, a sequence of 6 recordings could be three cars without trailers or two cars with trailers. To reduce such ambiguity, we will make the following two assumptions:

1. Any two successive times with a difference less than or equal to 1000 ms must belong to the same vehicle.
2. Any two successive times with a difference greater than or equal to 2000 ms must be from different vehicles.

Given a sequence of times, determine the number of cars with and without a trailer.

Input

The first line of the input contains a single integer n ($1 \leq n \leq 300\,000$), which is the number of times the pressure pad was triggered. The second line contains n distinct integers $t_1, \ldots, t_n$ ($0 \leq t_i < 2^{30}$) in increasing order, the times that the pressure pad was triggered. The times are in milliseconds.

Output

Display the number of cars with and without trailers. If the number of cars of each type can be uniquely determined, then display two lines of the form

```
Cars without trailers: X
Cars with trailers: Y
```

If there is no interpretation of the times that is consistent with the assumptions, then display `Impossible`. If there are multiple interpretations of the times that give different numbers of cars with and without trailers, then display `Ambiguous`.

Sample Input 1

```
7
10 200 5000 6100 7200 8300 9400
```

Sample Output 1

```
Cars without trailers: 2
Cars with trailers: 1
```

Sample Input 2

```
6
0 1100 2200 3300 4400 5500
```

Sample Output 2

```
Ambiguous
```

Sample Input 3

```
4
0 1000 2000 3001
```

Sample Output 3

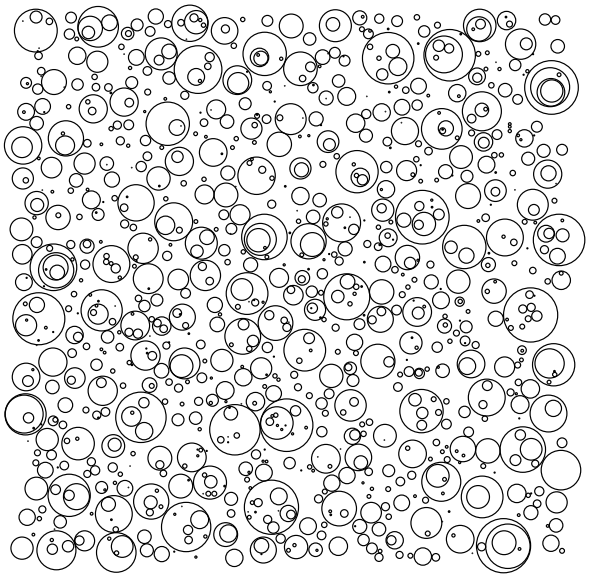
```
Impossible
```

Problem B Circles

Input file:	`circles.in`
Output file:	`circles.out`
Time limit:	2 seconds
Memory limit:	256 megabytes

Note: This problem uses file I/O.

There are n circles located on the plane. Circles may have common points, but for any two circles their intersection is either a point, or one of the two circles.



Find the total area covered by at least one circle.

Input

The first line of the input file contains integer number n ($1 \le n \le 100\,000$). The following n lines contain three integers each and describe circles. The i-th circle is described by coordinates of its center x_i and y_i and its radius r_i ($-10^6 \le x_i, y_i \le 10^6$, $1 \le r_i \le 10^6$).

Output

Output one real number: the total area covered by at least one circle. Your answer must have absolute or relative error of at most 10^{-9}.

Example

circles.in	circles.out
4 2 2 2 2 2 1 5 2 1 5 5 2	28.2743338823081391

C Deck of Cards

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	64 megabytes

While on a break from making the problems for Regional Finals, the South Pacific Programming Judges frequently likes to play their favourite card game, Uno. In the current game, there are just two players left, Malcolm and Richard, who are trying to figure out the optimal strategy to win. In this version of Uno, each player has a hand consisting of cards, each of which has a colour: red (R), green (G), blue (B) or yellow (Y) and a numeric value $1 - 9$.

The players alternate taking turns to play one card from their hand. On any turn, the card played must either be the same colour as the previous card that was played or have the same numeric value, or both (there may be duplicates of the same card). The first player who does not have any cards that can be played (nothing in their hand has the same colour or numeric value as the previous card that was played) loses. Since the judges prefer games of skill to games of chance, they have decided to play the game with a rule such that every player can see every player's hand. It is currently Malcolms's turn to play. Given the current card on the table (the most recently played card), determine who will win the game if both Malcolm and Richard play optimally.

Input

The first line of input contains two integers m ($1 \leq m \leq 1\,000$), which is the number of cards in Malcolm's hand and r ($1 \leq r \leq 1\,000$), which is the number of cards in Richard's hand.

The second line describes Malcolm's hand. This line contains m strings describing the cards in Malcolm's hand. Each card has a colour (either `R`, `G`, `B` or `Y`) and a numeric value (either `1`, ..., `8` or `9`).

The third line describes Richard's hand. This line contains r strings describing the cards in Richard's hand. Each card is described in the same way as above.

The fourth line of input contains a single string which describes the most recently played card in the same format as above.

Output

Display the winner of the game assuming both players play optimally.

Examples

standard input	standard output
3 2 B4 B5 Y2 B3 B7 B2	Malcolm
4 5 G1 Y3 R3 R9 B3 B9 G4 R8 Y1 R5	Richard
1 1 B1 G7 R5	Richard

D Juggling Errands

Time limit: 15 seconds

Zachary has just flown to Melbourne, and has a large number of errands to run at RMIT before he can go to the VCPC. In fact, he has exactly E errands. RMIT is a planned university and is built on a square $N \times N$ grid of locations. The grid is 4-connected, so Zachary can only travel north, south, east, or west to the directly adjacent grid location from any grid location.

Unfortunately, due to a festival, some grid locations have become impassable. Zachary would like to complete his errands extra fast. Can you write a program to help him? To complete an errand, Zachary will need to travel from some location in the university to another.

Input

The first line of input contains two integers N ($1 \leq N \leq 250$), which is the size of the grid in RMIT, and E ($1 \leq E \leq 100\,000$), which is the number of errands that Zachary must do.

The next N lines contain the map of RMIT of size $N \times N$. The character `X` indicates that the grid location is blocked. The character `.` indicates that you are free to move in and out of this grid location. The rows numbered 1 to N from top-to-bottom and the columns are numbered 1 to N from left-to-right.

The final E lines contain Zachary's errands. Each of these lines contain 4 integers r_1 ($1 \leq r_1 \leq N$), c_1 ($1 \leq c_1 \leq N$), r_2 ($1 \leq r_2 \leq N$) and c_2 ($1 \leq c_2 \leq N$), which indicates that Zachary has an errand from (r_1, c_1) to (r_2, c_2).

Output

Display a single line for each errand containing a single integer, which is the minimum distance Zachary must travel from (r_1, c_1) to (r_2, c_2). If there is no path for an errand, then output -1. Note that if either the start or destination locations are impassable, then there is no path.

Example

test	answer
5 6 ..X.. ..X.. XX... ..... XXXX. 1 1 2 2 1 1 3 3 2 2 2 2 3 3 5 5 1 2 5 5 1 3 5 5	2 -1 0 4 -1 -1

E Collecting Stars

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

You are planning to speedrun the latest Super Mario game. In this game, there are n stars that can be collected and your objective is to collect any k of them. Each star takes a certain amount of time to get. After collecting a star, Mario will reappear at the start of the game, so the time taken to collect any sequence of stars is the same regardless of the order that they are collected in. However, some stars do not become available for collection until a certain quantity of stars has already been collected.

Given a description of the stars, determine the fastest time in which you could collect k of them or determine that it is impossible to do so.

Input

The input starts with a line containing two integers n ($1 \leq n \leq 200\,000$), which is the number of stars, and k ($1 \leq k \leq n$), which is the number of stars you must collect.

The following n lines describe the stars. Each of these lines contains two integers t ($1 \leq t \leq 10^9$), which is the amount of time it will take to collect the star, and d ($0 \leq d < n$), which is the number of stars that must be collected before the star is available.

Output

Display the minimum amount of time to collect k stars. If you cannot collect k stars, display `IMPOSSIBLE`.

Examples

standard input	standard output
5 4 1 0 2 1 3 1 2 3 4 0	8
3 3 1 0 1 2 4 2	IMPOSSIBLE

F Holiday

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	256 megabytes

Your best friend from America is coming to visit you in Melbourne, Australia. "I hope the weather is nice," he says unaware of what Melbourne has in store for him. Your friend has identified n consecutive days over which he could be in Melbourne, but has not yet decided how long to stay. You have looked up the weather report for those n days and noted the temperature of each one. Your friend would like to experience all of the different temperatures of Melbourne.

Determine the minimum number of consecutive days your friend must stay in order to experience every temperature. That is, what is the smallest value of t such that no matter which of the first $n-t+1$ days your friend arrives, they will experience every different temperature at some point during their t day trip?

Input

The first line contains an integer n ($1 \leq n \leq 200\,000$), which is the number of days.

The second line describes the temperatures. This line contains n integers $t_1, t_2, \ldots, t_n$ ($0 \leq t_i \leq 10^9$), which are the temperatures on each day in order.

Output

Display the minimum number of consecutive days your friend must stay.

Examples

standard input	standard output
5 40 40 40 40 40	1
6 10 40 20 30 50 4	6
5 40 10 40 10 40	2
4 1 2 2 1	3