Codeforces Beta Round #94 (Div. 1 Only)

A. Statues

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

In this task Anna and Maria play a game with a very unpleasant rival. Anna and Maria are in the opposite squares of a chessboard (8×8): Anna is in the upper right corner, and Maria is in the lower left one. Apart from them, the board has several statues. Each statue occupies exactly one square. A square that contains a statue cannot have anything or anyone — neither any other statues, nor Anna, nor Maria.

Anna is present on the board as a figurant (she stands still and never moves), and Maria has been actively involved in the game. Her goal is — to come to Anna's square. Maria and statues move in turn, Maria moves first. During one move Maria can go to any adjacent on the side or diagonal cell in which there is no statue, or she can stay in the cell where she is. The statues during their move must go one square down simultaneously, and those statues that were in the bottom row fall from the board and are no longer appeared.

At that moment, when one of the statues is in the cell in which the Maria is, the statues are declared winners. At the moment when Maria comes into the cell where Anna has been waiting, Maria is declared the winner.

Obviously, nothing depends on the statues, so it all depends on Maria. Determine who will win, if Maria does not make a strategic error.

Input

You are given the 8 strings whose length equals 8, describing the initial position on the board. The first line represents the top row of the board, the next one — for the second from the top, and so on, the last line represents the bottom row. Each character string matches a single cell board in the appropriate row, and the characters are in the same manner as that of the corresponding cell. If the cell is empty, the corresponding character is "`.`". If a cell has Maria, then it is represented by character "`M`". If a cell has Anna, it is represented by the character "`A`". If a cell has a statue, then the cell is represented by character "`S`".

It is guaranteed that the last character of the first row is always "`A`", the first character of the last line is always "`M`". The remaining characters are "`.`" or "`S`".

Output

If Maria wins, print string "`WIN`". If the statues win, print string "`LOSE`".

Sample test(s)

input
```
.......A
........
........
........
........
........
........
M.......
```
output
```
WIN
```

input
```
.......A
........
........
........
........
........
SS......
M.......
```
output
```
LOSE
```

input
```
.......A
........
........
........
........
.S......
S.......
MS......
```

output
LOSE

B. String

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

One day in the IT lesson Anna and Maria learned about the lexicographic order.

String x is lexicographically less than string y, if either x is a prefix of y (and $x \neq y$), or there exists such i $(1 \leq i \leq min(|x|, |y|))$, that $x_i < y_i$, and for any j $(1 \leq j < i)$ $x_j = y_j$. Here $|a|$ denotes the length of the string a. The lexicographic comparison of strings is implemented by operator < in modern programming languages .

The teacher gave Anna and Maria homework. She gave them a string of length n. They should write out all substrings of the given string, including the whole initial string, and the equal substrings (for example, one should write out the following substrings from the string "aab": "a", "a", "aa", "ab", "aab", "b"). The resulting strings should be sorted in the lexicographical order. The cunning teacher doesn't want to check all these strings. That's why she said to find only the k-th string from the list. Help Anna and Maria do the homework.

Input

The first line contains a non-empty string that only consists of small Latin letters ("a"-"z"), whose length does not exceed 10^5. The second line contains the only integer k $(1 \leq k \leq 10^5)$.

Output

Print the string Anna and Maria need — the k-th (in the lexicographical order) substring of the given string. If the total number of substrings is less than k, print a string saying "No such line." (without the quotes).

Sample test(s)

input
aa 2
output
a

input
abc 5
output
bc

input
abab 7
output
b

Note

In the second sample before string "bc" follow strings "a", "ab", "abc", "b".

C. Games with Rectangle

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

In this task Anna and Maria play the following game. Initially they have a checkered piece of paper with a painted $n \times m$ rectangle (only the border, no filling). Anna and Maria move in turns and Anna starts. During each move one should paint inside the last-painted rectangle a new lesser rectangle (along the grid lines). The new rectangle should have no common points with the previous one. Note that when we paint a rectangle, we always paint only the border, the rectangles aren't filled.

Nobody wins the game — Anna and Maria simply play until they have done k moves in total. Count the number of different ways to play this game.

Input

The first and only line contains three integers: n, m, k ($1 \le n, m, k \le 1000$).

Output

Print the single number — the number of the ways to play the game. As this number can be very big, print the value modulo 1000000007 ($10^9 + 7$).

Sample test(s)

input
3 3 1
output
1

input
4 4 1
output
9

input
6 7 2
output
75

Note

Two ways to play the game are considered different if the final pictures are different. In other words, if one way contains a rectangle that is not contained in the other way.

In the first sample Anna, who performs her first and only move, has only one possible action plan — insert a 1×1 square inside the given 3×3 square.

In the second sample Anna has as much as 9 variants: 4 ways to paint a 1×1 square, 2 ways to insert a 1×2 rectangle vertically, 2 more ways to insert it horizontally and one more way is to insert a 2×2 square.

D. Numbers

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

One day Anna got the following task at school: to arrange several numbers in a circle so that any two neighboring numbers differs exactly by 1. Anna was given several numbers and arranged them in a circle to fulfill the task. Then she wanted to check if she had arranged the numbers correctly, but at this point her younger sister Maria came and shuffled all numbers. Anna got sick with anger but what's done is done and the results of her work had been destroyed. But please tell Anna: could she have hypothetically completed the task using all those given numbers?

Input

The first line contains an integer n — how many numbers Anna had ($3 \le n \le 10^5$). The next line contains those numbers, separated by a space. All numbers are integers and belong to the range from 1 to 10^9.

Output

Print the single line "YES" (without the quotes), if Anna could have completed the task correctly using all those numbers (**using all of them is necessary**). If Anna couldn't have fulfilled the task, no matter how hard she would try, print "NO" (without the quotes).

Sample test(s)

input
4 1 2 3 2
output
YES

input
6 1 1 2 2 2 3
output
YES

input
6 2 4 1 1 2 2
output
NO

E. Birthday

time limit per test: 4 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Anna's got a birthday today. She invited many guests and cooked a huge (nearly infinite) birthday cake decorated by n banana circles of different sizes. Maria's birthday is about to start in 7 minutes too, and while Anna is older, she decided to play the boss a little. She told Maria to cut the cake by k straight-line cuts (the cutting lines can intersect) to divide banana circles into banana pieces.

Anna has many guests and she wants everyone to get at least one banana piece. That's why she told Maria to make the total number of banana pieces maximum. It's not a problem if some banana pieces end up on the same cake piece — the key is to make the maximum number of banana pieces. Determine what result Maria will achieve.

Input

The first line contains two integers n and k — the number of banana circles and the number of cuts Maria should perform ($1 \le n \le 1000$, $1 \le k \le 10^5$). Next n lines contain the positions and sizes of the banana circles (all banana circles are round). On the cake the Cartesian coordinate system is defined. Each line contains three integers x, y and r — the coordinates of the center of the corresponding banana piece and its radius ($-1000 \le x, y \le 1000$, $1 \le r \le 1000$).

It is guaranteed that the banana circles do not intersect, do not touch each other and do not overlap with each other.

Pretest 10 is big test with $n = k = 1000$.

Output

Print the only integer — the largest number of banana pieces that Maria can get after she performs the k straight-line cuts.

Please do not use the `%lld` specificator to read or write 64-bit integers in C++. It is preferred to use the `cin`, `cout` streams or the `%I64d` specificator.

Sample test(s)

input
1 1 0 0 1
output
2

input
3 1 0 0 1 3 0 1 6 0 1
output
6

input
1 3 0 0 1
output
7