Problem A. Amoeba

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Monte-Carlo is an amoeba. Amoebas can feed on gel: whenever an amoeba encounters a piece of gel that is exactly as big as the amoeba, the amoeba will consume the gel and thus double its size.

Initially, the size of Monte-Carlo was some unknown positive integer. During its lifetime, Monte-Carlo encountered several gels and consumed the ones it could.

You are given a sequence a. The elements of a are the sizes of gels Monte-Carlo encountered, in chronological order.

Let S be the set of all possible final sizes of Monte-Carlo. Compute the number of positive integers that do not belong into S.

Input

The first line contains integer n — the number of gels Monte-Carlo encountered ($1 \leq n \leq 200$).

Next line consists of n integers and contains the sequence a ($1 \leq a_i \leq 10^9$).

Output

Output the number of positive integers that do not belong into S. It is possible to prove that the answer for any test case is finite and fits into a 32-bit signed integer type.

Examples

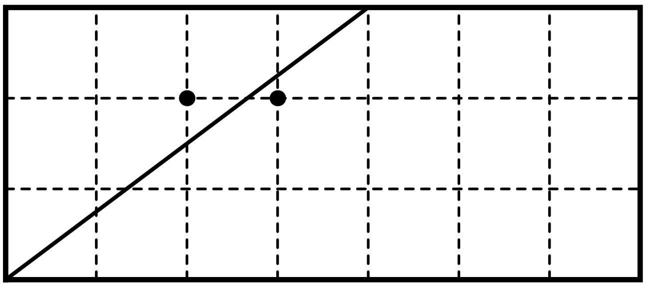
standard input	standard output
3 3 2 1	2
10 2 2 2 2 2 2 4 2 2 2	2
11 1 2 4 8 16 32 64 128 256 1024 2048	11

Problem B. Anniversary Cake

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Two students, Adam and Anton, are celebrating two-year anniversary of not passing their Math Logic exam. After very careful search in a local supermarket, they bought a rectangular cake with integer dimensions and two candles.

Later in the campus Adam put the candles into different integer points of the cake and gave a knife to Anton to cut the cake. The cut should start and end at integer points at the edges of the cake, and it should not touch the candles. Also each piece should have exactly one candle at it. Please, help Anton to find the starting and ending points of the cut.



A 7×3 cake and two candles at $(2,2)$ and $(3,2)$.
Anton can cut this cake through $(0,0)$ and $(4,3)$.

Input

The single line of the input contains six integers: w, h — cake dimensions; a_x, a_y — x and y coordinates of the first candle; b_x, b_y — the coordinates of the second candle ($3 \le w, h \le 10^9$; $0 < a_x, b_x < w; 0 < a_y, b_y < h$; $a_x \neq b_x$ or $a_y \neq b_y$).

Output

Output four integers s_x, s_y, e_x, and e_y — the starting and ending coordinates of the cut. Both starting and ending point of the cut should belong to the sides of the cake.

If there are several solutions, output any of them.

Example

standard input	standard output
`7 3 2 2 3 2`	`0 0 4 3`

Problem C. Ants Meet

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Magical Girl Lein is observing ants.

There are n ants. At first, they have integer coordinates in the Cartesian plane. More precisely, ant i starts at the point (x_i, y_i). All ants move at the same speed. Each ant moves in one of the four basic directions. (I.e., either parallel to the x axis or parallel to the y axis.) When 2 or more ants meet at the same time, these ants disappear.

You are given coordinates of the ants and their directions. Find the number of ants that still exist after the last meeting occurs.

Input

First line contains integer n — the number of ants ($1 \le n \le 50$). Next n lines describe the position and direction of each of the ants. Each of these lines contains two integers x_i and y_i ($|x_i|, |y_i| \le 1000$) — coordinates of i-th ant location, and a character 'N', 'E', 'S' or 'W', which encodes the direction in which ant i is going:

- 'N' means north (y coordinate increases),
- 'E' means east (x coordinate increases),
- 'S' means south (y coordinate decreases),
- 'W' means west (x coordinate decreases).

All ant locations are distinct.

Output

Output an integer: the number of ants that still exist after the last meeting occurs.

Examples

standard input	standard output
4 0 0 N 10 10 W 20 20 N 30 30 E	2
4 -10 0 N 0 -10 E 0 10 W 10 0 S	0
9 -1 0 E 1 0 E 0 0 N -1 -1 W 1 -1 W 0 1 S 0 -1 W -1 1 S 1 1 E	5

Problem D. Baby-step-giant-step

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

You are standing at the point $(0, 0)$ of the plane. In one step you can move from point (x_f, y_f) to any point of the plane (x_t, y_t) such that Euclidean distance between these points($\sqrt{(x_f - x_t)^2 + (y_f - y_t)^2}$) is either a or b. Find the minimum number of steps you have to make in order to get in point $(d, 0)$.

Input

The first line contains one integer t denoting the number of test cases you have to process.

Each of the next t lines contains three integers a, b and d denoting the lengths of steps you can make and the x-coordinate of your target.

Constraints

$1 \le t \le 10^5$

$1 \le a < b \le 10^9$

$0 \le d \le 10^9$

Output

Output t lines containing the minimum number of steps for each test.

Example

standard input	standard output
3 2 3 1 1 2 0 3 4 11	2 0 3

Note

In the first case one of optimal paths is $(0, 0)$, $(\frac{1}{2}, \frac{\sqrt{15}}{2})$, $(0, 1)$.

In the second case we can make no steps.

In the second case one of optimal ways is $(0, 0)$, $(0, 4)$, $(0, 8)$, $(0, 11)$.

Problem E. Backspace

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

There is a text field which initially contains string s. You can press a button with lowercase English symbol or backspace button. In the first case the symbol is appended to the end of the string. In the second case, if the string is empty then nothing happens, otherwise the last symbol of the string is deleted.

You want to obtain string t in exactly k button presses. Determine whether you can do this or not.

Input

The first line contains string s denoting the initial string in text field. The second line contains string t denoting the string you want to get. Both strings consist only of lowercase English symbols. The third line contains single integer k denoting the number of button presses.

Constraints

$1 \le |s| \le 100$

$1 \le |t| \le 100$

$1 \le k \le 100$

Output

Output "Yes"if you can obtain string t or "No"otherwise.

Examples

standard input	standard output
hackerearth hackerrank 9	Yes
aba aba 7	Yes

Note

Sample 0

In this test you can press backspace 5 times and then type *rank* which is exactly 9 presses.

Sample 1

In this test you can press backspace 4 times and then type *aba* which is exactly 7 presses.

Problem F. Concatenation

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Famous programmer Gennady likes to create new words. One way to do it is to concatenate existing words. That means writing one word after another. For example, if he has words "cat" and "dog", he would get a word "catdog", that could mean something like the name of a creature with two heads: one cat head and one dog head.

Gennady is a bit bored of this way of creating new words, so he has invented another method. He takes a non-empty prefix of the first word, a non-empty suffix of the second word, and concatenates them. For example, if he has words "tree" and "heap", he can get such words as "treap", "tap", or "theap". Who knows what they could mean?

Gennady chooses two words and wants to know how many different words he can create using his new method. Of course, being a famous programmer, he has already calculated the answer. Can you do the same?

Input

Two lines of the input file contain words chosen by Gennady. They have lengths between 1 and 100 000 characters and consist of lowercase English letters only.

Output

Output one integer — the number of different words Gennady can create out of words given in the input file.

Examples

standard input	standard output
cat dog	9
tree heap	14

Problem G. iFactory

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

iMagine yourself as a loyal fan of one popular company that creates all kind of gadgets. After all of these years "that company" has created n different mobile phones, enumerated from 1 to n and powering devices, enumerated the same way. Due to the production policy, phones can be charged only by devices with phone's version not exceeding charger's version. Moreover, phone's and device's total version has to be equal or less than k. For example, if phone's version is a and charger's is b, then the phone can be charged if $a \leq b$ and $a + b \leq k$.

You are so loyal, you have already bought every device — all n phones and n chargers! Moreover, you are fond of tea and taking photos with your cool mug, but with your effort phone discharges in a couple of hours, and you have to take a phone and a charger with you to the tea-shop and, of course, a pair of different ones.

Your task is to find the number of days, you can visit the tea-shop with a different pair of phone and a charger, which you haven't brought before, so that you can charge the phone, according to the rules mentioned above.

Input

The first line of input contains one integer t — number of test cases ($1 \leq t \leq 10\,000$).

For each case, single line contains two integers n and k — number of mobile phones and versions' maximum sum, respectively ($1 \leq n, k \leq 2 \cdot 10^9$).

Output

For each case, output on a separate line a single integer — number of different pairs of a phone and a charger, satisfying the rules described in the statement.

Scoring

Subtask	Points	Constraints n, k, t
1	37	$1 \leq n, k \leq 1\,000$, $1 \leq t \leq 100$
2	31	$1 \leq n, k \leq 10^6$, $1 \leq t \leq 100$
3	32	$1 \leq n, k \leq 2 * 10^9$, $1 \leq t \leq 10\,000$

Example

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

Note

Let the first number in a pair be phone's version and second — charger's.

Then, for the first test case, there are four days, you can go to the tea-shop with a different pair of phone and a charger, which you haven't brought before, so that you can charge the phone, according to the rules mentioned in the statement:

1. 2, 2
2. 1, 2
3. 1, 3
4. 1, 1

Next, for the second test case, versions' maximum sum is 1, therefore no pair of a charger and a phone exists.

Finally, for the third test case, there are two days:

1. 1, 2
2. 1, 1

Problem H. Easy Arithmetic

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Eva is a third-grade elementary school student. She has just learned how to perform addition and subtraction of arbitrary-precision integers. Her homework is to evaluate some expressions. It is boring, so she decided to add a little trick to the homework. Eva wants to add some plus and minus signs to the expression to make its value as large as possible.

Input

The single line of the input file contains the original arithmetic expression. It contains only digits, plus ('+') and minus ('-') signs.

The original expression is correct, that is:

- numbers have no leading zeroes;
- there are no two consecutive signs;
- the last character of the expression is a digit.

The length of the original expression does not exceed 1000 characters.

Output

Output a single line — the original expression with some plus and minus signs added. Output expression must satisfy the same correctness constraints as the original one. Its value must be as large as possible.

Examples

standard input	standard output
10+20-30	10+20-3+0
-3-4-1	-3-4-1
+10	+10

Problem I. Jokewithpermutation

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	64 megabytes

Joey had saved a permutation of integers from 1 to n in a text file. All the numbers were written as decimal numbers without leading spaces.

Then Joe made a practical joke on her: he removed all the spaces in the file.

Help Joey to restore the original permutation after the Joe's joke!

Input

The input file contains a single line with a single string — the Joey's permutation without spaces.

The Joey's permutation had at least 1 and at most 50 numbers.

Output

Write a line to the output file with the restored permutation. Don't forget the spaces!

If there are several possible original permutations, write any one of them.

Example

standard input	standard output
`4111109876532`	`4 1 11 10 9 8 7 6 5 3 2`

Problem J. Knight circuit 2

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

The knight is a chess piece that moves by jumping: two cells in one direction, one in the other. Formally, a knight standing on the cell (x, y) can move to any of the following eight cells: $(x+2, y+1)$, $(x+2, y-1)$, $(x-2, y+1)$, $(x-2, y-1)$, $(x+1, y+2)$, $(x+1, y-2)$, $(x-1, y+2)$, and $(x-1, y-2)$. Of course, if the knight is close to the edge of the board, some of these cells might not be on the board. It is not allowed to jump to a cell that is not on the board.

A knight circuit is a sequence of cells on a chessboard that starts and ends with the same cell. Each consecutive pair of cells in the knight circuit must correspond to a single jump of the knight. The knight circuit may visit each cell arbitrarily many times. The size of a knight circuit is the number of different cells visited by the circuit.

Calculate the maximum knight circuit size that can be obtained on the given board of size $w \times h$. You are free to choose any cell as the start of your circuit.

Input

Single line contains two integers w and h ($1 \leq w, h \leq 45000$).

Output

Output single number — answer for the problem.

Examples

standard input	standard output
`1 1`	`1`
`15 2`	`8`
`100 100`	`10000`

Problem K. Little Elephant and Strings

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Little Elephant from the Zoo of Lviv likes strings.

You are given a string a and a string b of the same length. In one turn Little Elephant can choose any character of a and move it to the beginning of the string (i.e., before the first character of a). Print the minimal number of turns needed to transform a into b. If it's impossible, print "`-1`" instead.

Input

The first line contains two strings a and b ($1 \leq |a|, |b| \leq 50$, $|a| = |b|$). a and b will consist of uppercase letters ('`A`' — '`Z`') only.

Output

Print one integer.

Examples

standard input	standard output
`ABC CBA`	`2`
`A B`	`-1`
`AAABBB BBBAAA`	`3`
`ABCDEFGHIJKLMNOPQRSTUVWXYZ` `ZYXWVUTSRQPONMLKJIHGFEDCBA`	`25`
`A A`	`0`
`DCABA DACBA`	`2`

Note

Test 1: The optimal solution is to make two turns. On the first turn, choose character '`B`' and obtain string "`BAC`". On the second turn, choose character '`C`' and obtain "`CBA`".

Test 2: In this case, it's impossible to transform a into b.

Problem L. ORSolitaire

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Manao is playing a solitaire game called OR-solitaire. In this game, the player starts with a number $x = 0$ and should obtain the number g in one or more moves. The set of valid moves is determined by an array a. In each move, the player chooses some element of a and replaces x with the bitwise OR of x and the chosen element.

Fox Ciel wants Manao to stop playing OR-solitaire and move on with his life. She decided to erase some of the elements from a in such a way that it becomes impossible to complete the game. Find out the minimum number of elements that need to be removed to achieve this.

Input

First line contains two integers m and g: the size of a and the number Manao should obtain ($1 \le m \le 50$, $1 \le g \le 10^9$).

Next line contains m integers a_i: ($1 \le a_i \le 10^9$).

Output

Print the minimum number of elements that need to be removed to achieve Ciel wish.

Examples

standard input	standard output
`3 7` `1 2 4`	`1`
`5 7` `1 2 4 7 8`	`2`
`3 1` `12571295 2174218 2015120`	`0`
`11 11` `5 2 4 52 62 9 8 3 1 11 6`	`3`

Note

First test: The goal of the game is to obtain $x = 7$ from $x = 0$. The possible moves are to replace x with bitwise OR of x and 1, bitwise OR of x and 2 and bitwise OR of x and 4. $x = 7$ can be obtained only by using each of the three moves at least once, so removing any single element from a will make the game impossible to finish.

Second test: In this example, Fox Ciel should remove the number 7 and one of the numbers 1, 2, 4.

Third test: There is no need to remove elements from a, since the game cannot be completed in its initial version.

Problem M. Rooks on board

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Daria likes playing chess. She has an extraordinary chessboard of size $n \times m$. Her favorite figure is rook.

Daria also likes coming up with puzzles and solving them. The one she came up with lately is about rooks. She has $n \cdot m$ identical rooks and his extraordinary chessboard. She wants to place all the rooks one by one on the chessboard occupying each cell. The only restriction is that for each rook he places, there are even number of rooks already on the chessboard, which can capture the new rook.

Help Daria to solve the puzzle. Given n and m, find the order of cells, which Daria should place the rooks in.

Input

Input contains two integers n and m.

Output

If there is an order of cells solving the puzzle, first line should contain "`YES`" and the next $n \cdot m$ lines should contain pairs of integers x_i and y_i $(1 \le x_i \le n;\ 1 \le y_i \le m)$ — cell coordinates.

If no solution exists, output "`NO`".

Examples

standard input	standard output
`1 1`	`YES` `1 1`
`1 2`	`NO`

Problem N. Stone Division

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

There is a pile with n stones on the table. There is also a set S of m distinct integers $\{a_0, \ldots, a_{m-1}\}$. Two players make turns splitting one of the piles on the table into k piles of equal size if it is possible and if $k \in S$. If a player can't make any valid turn, he loses the game. Determine, which player wins a game if both of them play optimally.

Input

The first line contains two integers n and m denoting the size of the initial pile and the size of the set.

The second line contains m distinct integers a_i.

Constraints

$1 \le n \le 10^{18}$

$1 \le m \le 10$

$2 \le a_i \le 10^{18}$

Output

Output "First"if the first player wins the game and "Second"otherwise.

Example

standard input	standard output
15 3 5 2 3	Second

Note

If the first player splits a pile into 5 piles of size 3, in the next 5 turns players will split these piles. So the first player will lose after 6 turns.

If the first player splits a pile into 3 piles of size 5, in the next 3 turns players will split these piles. So the first player will lose after 4 turns.

Thus, the second player wins the game.

Problem O. The Division Game

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Manao likes to play the Division Game with his friends. This two-player game is played with some collection of natural numbers s. Manao plays first and the players alternate in making a move. A move is choosing some number x from s and a natural number $y > 1$ such that y divides x. Then, x is replaced by $\frac{x}{y}$ in the collection. Note that at any moment the collection may contain multiple copies of the same number. The game proceeds until no more moves can be made. The player who managed to make the last move is declared the winner.

Since hot debates arise on what numbers should be in s, the friends decided to regularize their choice. They always choose a contiguous interval of numbers $[a, b]$ to be the initial collection s. That is, at the beginning of the game, the collection s contains each of the integers a through b, inclusive, exactly once. Manao knows that a and b will satisfy the condition $L \le a \le b \le R$. Count the number of such intervals for which Manao will win the Division Game given that both players play optimally.

Input

Input contains two integers L and R ($2 \le L \le 10^9$; $L \le R \le L + 10^6$).

Output

Output single integer: the number of such intervals for which Manao will win the Division Game given that both players play optimally.

Examples

standard input	standard output
9 10	2
2 5	9
2 6	13
2 100	4345

Problem P. Least common multiple

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Gennady and Artem are discussing the solutions of different problems. Gennady told Artem about number theory problem he solved the day before. One of the steps to solve the problem was to calculate the least common multiple of all integers from 1 to n, inclusive. This problem inspired Gennady to come up with another problem about LCMs: Given n, find greatest $m \leq n$, such that $\text{LCM}(1, 2, \ldots, n) = \text{LCM}(m, m+1, \ldots, n)$.

We have no doubt Artem will solve the problem, but will you?

Input

Input contains single integer n.

Output

Output the only integer m.

Examples

standard input	standard output
5	3
10	7