Problem A. Acronyms

Input: Standard
Output: Standard
Author(s):   Manuel Felipe Pineda Loaiza - UTP Colombia

There are several interesting developments in the history that were named in honor of their inventors, for example:

- RSA is made of the initial letters of the surnames of Ron Rivest, Adi Shamir, and Leonard Adleman.

- BPS is made of the initial letters of Bogomolnyi, Prasad, and Sommerfield.

Sometimes, people change the name of those developments with other words (not necessarily meaningful), for example:

- RSA: Randomized Super Algorithm.

- BPS: Beyond Perturbative States.

In this task you must check if a name may have come from certain authors. A name comes from certain authors if the initials of the authors are the same as the initials of the words in the name. Note that the initials should match in the same order that they are given in the input.

Input

The input consists of several test cases. Each test case consists of two lines:

- The name of the authors separated by spaces.

- The name to check separated by spaces.

All the words consist of lowercase english letters.

The number of authors is greater than 1 and less than 10. This number may be different from the number of words in the name to check.

Output

For each test case print "yes", if the name may have come from the given authors. Print "no" otherwise.

Example

Input	Output
rivest shamir adleman	yes
randomized super algorithm	yes
bogomolnyi prasad sommerfield	no
beyond perturbative states	no
arthur jack sean	
amazing jurisprudential victory	
rivest adleman shamir	
randomized super algorithm	

Problem B. Tobby Bones

Input:	Standard
Output:	Standard
Author(s):	Santiago Gutiérrez Alzate - UTP Colombia and Google

Tobby is a very intelligent dog who loves to collect bones. In his house Tobby has a large shelf in which he stores his bones. The shelf is divided in N slots and each of those slots can hold at most one bone. The only difference between bones is their size, which is represented by an integer. Tobby is constantly buying new bones to fill his shelf and to replace other bones from his shelf.

Tobby occasionally takes some of his bones for exhibitions around the country. Since Tobby is a lazy dog, he decides which bones to carry to an exhibition in the following way: he chooses two indexes i and j with $i \leq j$ and he carries to the exhibition all bones located between positions i and j of his shelf that have a size of at most S.

Given a log of bones purchases and of exhibitions that Tobby needs to make between purchases, your task is to tell Tobby how many bones he should bring to each exhibition.

Initially all the slots in Tobby's shelf are empty.

Input

The input contains several test cases. Each test case begins with one line, in which there are two positive integers N ($1 \leq N \leq 6 * 10^5$) and M ($1 \leq M \leq 2 * 10^6$) that represent, respectively, the number of slots in Tobby's shelf and the number of queries in the log. Then M lines follow, each containing a log entry. Each log entry starts with a number Q:

If Q is 0, then the log entry indicates that Tobby bought a bone. In that case two numbers follow: I ($0 \leq I \leq N - 1$) and B ($1 \leq B \leq 6 * 10^5$), indicating the index within the shelf in which the bone will be stored and the size of the bone. If there is already a bone stored in index I, then Tobby just replaces the bone in that index with the new bone.

If Q is 1, then the log entry indicates that Tobby wants to know how many bones he should bring to an exhibition. In that case three numbers follow: I, J ($0 \leq I \leq J \leq N - 1$) and S ($1 \leq S \leq 6 * 10^5$), indicating that Tobby wants to make an exhibition with all bones located between I and J (inclusive) of his shelf that have size less than or equal to S.

Output

For each exhibition query in the input, you should print a single line with the number of bones that Tobby should bring to that exhibition.

Example

Input	Output
5 10	1
0 0 5	2
0 1 6	2
0 2 7	4
1 0 2 5	
1 0 2 6	
0 3 5	
0 4 6	
1 0 4 5	
0 0 8	
1 0 4 7	

Use fast I/O methods

Problem C. Tobby and Query

Input:	Standard
Output:	Standard
Author(s):	Jhon Jiménez - UTP Colombia

In his free time Tobby is always searching for interesting things. This time Tobby created the following problem: given a sequence of n integer numbers, Tobby would like to know how many different numbers are in the range $[l, r]$ ($r \geq l$)

Input

The input has several test cases. The first line of each test case contains an integer n ($1 \leq n \leq 10^5$), the size of the sequence of numbers. The next line contains n values a_i ($0 \leq ai \leq 9$), the numbers in the sequence. The next line contains an integer q ($1 \leq q \leq 10^4$), the amount of queries. Then there are q lines, each line contains a query: two integers l and r ($1 \leq l, r \leq n$).

Output

For each test case print q integers, representing the amount of different numbers in the range $[l, r]$ for each query in the input.

Example

Input	Output
7	1
0 2 3 3 7 5 2	2
3	4
1 1	1
2 4	1
2 7	
5	
7 7 7 7 7	
2	
4 5	
1 5	

Use fast I/O methods

Problem D. Standard Deviation

Input:	`standard`
Output:	`standard`
Author(s):	Hugo Humberto Morales Peña - UTP Colombia

In mathematics, the standard deviation of a set of n integer numbers is defined as:

$$S = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1}}$$

where $\overline{x}$ is the average of the set of n integer numbers for which the standard deviation is being calculated. That average is calculated as:

$$\overline{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i$$

The task is to calculate, **in an efficient way**, the standard deviation of the first n odd positive integer numbers.

Input

There are several test cases in the input. Each test case consists of a single line containing a positive integer number n ($2 \leq n \leq 10^6$) which indicates the amount of consecutive odd numbers (starting from one) that should be considered when calculating the standard deviation. The last test case has a value of 0, for which you shouldn't generate any response.

Output

For each test case, you should print a single line containing a floating point number: the standard deviation of the first n odd positive numbers. The absolute error of your answer should not be greater than 10^{-6}.

Example

Input	Output
10	6.055301
100	58.022984
1000	577.638872
10000	5773.791360
100000	57735.315593
1000000	577350.557865
0	

Use fast I/O methods

Problem E. Tobby and the LED display

Input:	Standard
Output:	Standard
Author(s):	Carlos Arias Londoño - UTP Colombia

Tobby got his first job! Being a lazy puppy, the work that he has found is not very demanding and it consists of observing a LED display during T minutes and reporting the status of the LED display once this time runs out. The LED display is capable of displaying N characters and the text can move in two directions, Left or Right. In every minute the character that occupies the i_{th} position moves to the $(i-1)_{th}$ or $(i+1)_{th}$ position, depending on the direction in which the text moves in the LED display.

The LED display works in a circular way, therefore, if the character that occupies the $i = 1$ position moves to the left its new position will be $i = N$, moreover, if the character that occupies the $i = N$ position moves to the right its new position will be $i = 1$.

i.e. if Tobby got the board shown below where $N = 10$, $T = 3$ and the direction in which the text moves is Right, the following will happen:

```
Minute 0
+------------------+
| | | |T|o|b|b|y|!|!|
+------------------+

Minute 1
+------------------+
|!| | | |T|o|b|b|y|!|
+------------------+

Minute 2
+------------------+
|!|!| | | |T|o|b|b|y|
+------------------+
```

After 3 minutes, Tobby should report the board shown below.

```
+------------------+
|y|!|!| | | |T|o|b|b|
+------------------+
```

As it has been said, Tobby is very lazy and wont spends his time on this boring task, that's why he is willing to give you a bone from his first payment as reward :).

Input

The input consists of several test cases and must be read until EOF.

The first line of each test case contains two integers N, T ($1 \leq N \leq 50000$, $1 \leq T \leq 10^{14}$), and one character D ($D =$ 'L' or $D =$ 'R'), here N indicates the number of characters that the LED display can show, T shows the number of minutes that Tobby must wait to report the LED display state and D is the direction in which the LED display will work $L = Left$ and $R = Right$.

Then, there will be 3 lines and each one has $(2 * N) + 1$ characters. The first and third line are the upper edge and the lower edge respectively of the LED display, The second line shows the initial content of the LED display.

Output

For each test case the output must consist of 3 lines each one will have $(2 * N) + 1$ characters. The first and third line are the upper edge and the lower edge respectively of the LED display and the second one will show the LED display state after T minutes.

Example

Input	Output
10 3 R	+------------------+
+------------------+	|y|!|!| | | |T|o|b|b|
| | | |T|o|b|b|y|!|!|	+------------------+
+------------------+	

Use fast I/O methods

Problem F. Triangular Test II

Input:	Standard
Output:	Standard
Author(s):	Alejandro Moreno Agudelo - UTP Colombia

Hugov Molotov, an evil computer scientist, is obsessed by triangular numbers and how to use them to control the world. Molotov wants to choose a new henchman, so he has prepared a hard task, to test if you are worthy. The task consist in, given a positive integer number N, tell what is the minimum amount of triangular numbers whose sum is equal to N.

Note: a triangular number is given by:

$$\sum_{i=1}^{X} i = 1 + 2 + 3 + 4 + \ldots + X$$

where X is any positive integer number.

Input

The input may contain several test cases. Each test case is presented on a single line, and contains one positive integer N ($1 \leq N \leq 10^7$). The end of the test cases is given by the end of file (EOF).

Output

For each test case, your program must print one positive integer denoting the minimum amount of triangular numbers, whose sum is equal to N. Each test case must generate one output line.

Example

Input	Output
1	1
2	2
6	1
10	1

Use fast I/O methods

Problem G. Tobby and the line game

Input:	Standard
Output:	Standard
Author(s):	Diego Agudelo España - UTP Colombia

Tobby the pet enjoys a lot going to the circus and he likes to take part in the games available for smart pets like him. This time a new circus has just arrived to Tobby's town and they are promoting a new game called the line game.

The line game is a very simple game played over a rectangular field whose lower left corner is $\mathbf{p}_L = (x_L,\ y_L)$ and its upper right corner is $\mathbf{p}_R = (x_R,\ y_R)$. In one game there are multiple players and each of them have to throw two coins over the game field. The game's goal is to make the line joining the places where the two coins landed as large as possible. The player who manages to make the longest line from his threw coins is the winner of the game and therefore the winner of a big prize as well.

Tobby has been watching some rounds of the line game and he is wondering which are the chances for him to win this game. He knows that he can barely throw the coins with his dog's paws so it can be assumed that all the points over the game field are equally probable for the coins to land and it can also be assumed that the two throws are independent. However, Tobby is so skilled that you can be sure his coins will always land inside the game field or over its boundaries. Tobby wants to compute the expected value of the squared length of the resulting line since this result can help him to make a choice about whether or not he should take part in the line game. Can you help him?

For this problem the coin landing locations can be assumed to be 2D points over a continuous plane. Suppose that the first coin landed at $\mathbf{p}_1 = (x_1,\ y_1)$ and the second at $\mathbf{p}_2 = (x_2,\ y_2)$ so Tobby is interested in computing the expectation over $\|\mathbf{p}_1 - \mathbf{p}_2\|^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$.

Input

The input file contains several test cases. Each test case comprises 4 integer numbers, namely x_L, y_L, x_R and y_R which represent the rectangular field as described above. The end of the input file will be given by EOF.

Constraints:

- $-500 \le x_L,\ y_L,\ x_R,\ y_R \le 500$

- $x_L < x_R$

- $y_L < y_R$

Output

Print one number, the required expected value. The absolute error of your answer should not be greater than 10^{-5}.

Example

Input	Output
-100 -50 23 400	36271.5
0 0 500 1	41666.833333333314

Problem H. Painting the Wall

Input:	Standard
Output:	Standard
Author(s):	Sebastián Gómez - UTP Colombia

In his years of youth rebellion, Humbertov Moralov decided to paint the walls of this university. But unlike the rest of the revel young people, Moralov invented a robot to paint the wall for him.

The robot can execute two simple instructions to draw lines, namely:

- hline r c1 c2: draw a horizontal line in the row r between columns c1 and c2.

- vline c r1 r2: draw a vertical line in the column c between rows r1 and r2.
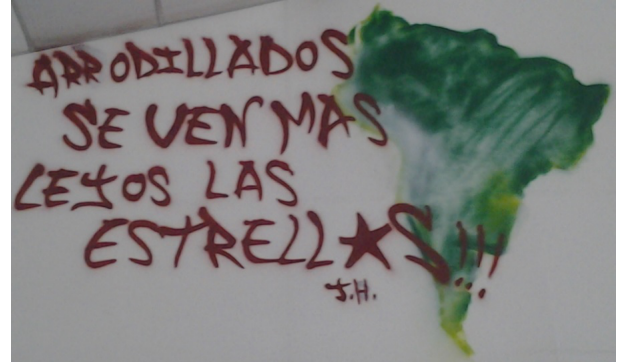
Now Moralov wants you to write a program that given the piece of rebel art to paint, outputs a program for the robot with the minimum number of instructions to paint the wall.

Hint: Some algorithms that you may find useful to solve this problem may have a theoretical complexity analysis that seems too high for the given input size. Remember that these complexity analysis are usually done thinking on the worst case, and many algorithms run much faster in practice. The input file for this problem was not constructed to break any particular solution in terms of complexity.

Input

The input consists of several test cases. Each test case begins with a line with two integers R and C corresponding to the number of rows and columns of the design to paint in the wall. Then R lines follow each with C characters. A character '*' means that the cell needs to be painted, and '.' means that the cell should be left unpainted.

- $1 \leq R, C \leq 800$

Output

For each test case print in one line a single integer n corresponding to the minimum number of instructions to paint the wall. Then n lines should follow with the instructions to actually paint the wall. If there are multiple solutions output any of them.

Example

Input	Output
5 7	3
.*...*.	vline 2 1 5
.*...*.	vline 6 1 5
.*****.	hline 3 2 6
.*...*.	
.*...*.	

Problem I. Tobby on Tree

Input:	Standard
Output:	Standard
Author(s):	Manuel Felipe Pineda Loaiza - UTP Colombia

Do you remember Tobby? He is a very cute dog who lives in tobbyland.

Tobby lives in a city where any two houses are connected by exactly one path, and that path is bidirectional. All the houses are identified by an integer $0 \leq id < N$, and have another integer indicating its corresponding value.

Tobby found on internet an interesting game called, "the subtraction game" that goes as follow:

Given a sequence of N elements: a_0, a_1, …, a_{n-1}:

```
If all the numbers are equal, the game ends.
Otherwise
    Select two numbers which are unequal
    Subtract the smaller number from the larger number
    Replace the larger number with the result from above.
```

The game **always** ends and no matter how you play, it will always terminate **on the same value**. The result of the game is that value.

The game is so cool for Tobby that he decides to play it as much as possible. For this reason each day Tobby visits two friends who live in the houses u and v and records all the values of the houses in the path from u to v (inclusive). At the end of the day he plays the substraction game with the sequence that he wrote.

The value of a house can change sometimes, but Tobby will always know the new value.

In order to help Tobby, he asks to answer the following types of queries:

1. given u, v: Find the result of the subtraction game on the path from u to v (inclusive).

2. given u, x: Change the value of the house u by x.

Input

The input has several test cases. The first line contains an integer $N \leq 50000$, the number of houses. The next line contains N integers, the i-th number is the value of the i-th house $1 \leq value \leq 10000$.

The next $N - 1$ lines contain two integers $0 \leq u, v \leq N - 1$ indicating a connection between the houses u and v.

The next line contains an integer $1 \leq q \leq 10000$ followed by q lines each containing a query. The first number of each of those lines is the query type and the rest of the line contains the query parameters.

Output

For each query of type 1, print the result of the subtraction game in the path from u to v.

Example

Input	Output
5	5
5 15 20 15 9	3
0 2	
0 3	
3 1	
3 4	
3	
1 2 1	
2 3 3	
1 1 4	

Use fast I/O methods

Problem J. Tobby Stones

Input:	Standard
Output:	Standard
Author(s):	Santiago Gutiérrez Alzate - UTP and Google

Tobby is an intelligent Boston Terrier who has a garden. In his garden he has N stones which, like Tobby, are either black or white. The stones are organized in a row and numbered from 0 to $N - 1$.

Tobby likes to change his stones quite a lot and he does so by taking a contiguous part of his stones, from integer indexes I to J, and making some changes to all stones located between those two indexes (including the stones at I and J).

Given a list of changes that Tobby makes to his garden, your task is to answer queries about the number of black and white stones in his garden.

Initially all stones in Tobby's garden are white.

Input

The input contains several test cases. Each test case starts with a line in which there are two positive integers N ($1 \leq N \leq 10^6$) and M ($1 \leq M \leq 10^6$) that represent, respectively, the number of stones in Tobby's garden and the number of changes and queries to answer. Then M lines follow, each containing either a change or a query. Each of those lines starts with an integer Q ($0 \leq Q \leq 3$):

If Q is 0, then the line is a query, in that case two numbers follow: I ($0 \leq I < N$) and J ($I \leq J < N$), indicating the start and end positions to consider to answer the query.

If Q is 1, then the line is a change of type reverse, in that case two integers follow: I ($0 \leq I < N$) and J ($I \leq J < N$), indicating that Tobby wants to reverse the color of all stones between those indexes so that the color of stone $I + K$ becomes the color of stone $J - K$ and the color of stone $J - K$ becomes the color of stones $I + K$ for all integer K such that $0 \leq K$ and $I + K < J - K$.

If Q is 2, then the line is a change of type flip color, in that case two integers follow: I ($0 \leq I < N$) and J ($I \leq J < N$), indicating that Tobby wants to change the color of all stones between those indexes so that all white stones become black stones and all black stones become white stones.

If Q is 3, then the line is a change of type set color, in that case three integers follow: I ($0 \leq I < N$), J ($I \leq J < N$) and C ($0 \leq C \leq 1$) indicating that Tobby wants to set the color of all stones between indexes I and J. If C is 0, then all stones between those two indexes should become black, while if it is 1 they should become white.

Output

For each query in the input, you should print a single line with two integers: the number of black stones and the number of white stones between the indexes I and J given in the query.

Example

Input	Output
10 7	0 10
0 0 9	5 0
3 0 4 0	5 0
0 0 4	0 7
1 0 9	0 51
0 5 9	
2 5 9	
0 3 9	
100 1	
0 0 50	

Use fast I/O methods

Problem K. Tobby and Seven

Input:	Standard
Output:	Standard
Author(s):	Jhon Jiménez - UTP Colombia

Tobby has a very special number, when this number is divided by 7 the remainder is 0. Every time that Tobby thinks about this property, he says: WOW, this is a wonderful number!

As everybody knows Tobby is not a common pet, he is smart and curious, so he decides to modify the original number as follows: Tobby writes the number in its binary representation, then he chooses two indexes and make a swap operation of the values of those positions, and then he does several other similar operations for a while. Tobby does not remember the total number of operations that he performed, he just remembers the positions in the binary representation that were swapped at some point and he also remembers that the most significant bit was never swapped.

Now Tobby wants to recover the original number, but Tobby is a busy pet and so he does not have time for that. Can you help him?

Input

The input has several test cases. Each test case has three lines, the first line contains a single integer n ($7 \leq n \leq 2^{60}$), the modified number. The second line contains an integer k ($2 \leq k \leq 20$) representing the amount of positions that were swapped and the third line contains k different integers in increasing order, the positions (relative to the least significant bit) of the number that were swapped (zero-indexed).

Output

Print a single integer: the recovered number which is divisible by 7. If there are several answers, then print the maximum possible value.

Example

Input	Output
79	91
5	28
0 1 2 4 5	130816
21	1152921504606846975
2	
0 3	
65791	
16	
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	
1152921504606846975	
5	
13 39 40 58 59	

Explication

In the first sample the original number (without swaps) is 91

Tobby remembers that in sometime the positions 0, 1, 2, 4 and 5 were swapped.

Suppose that Tobby did the follow swap operations:

6 5 4 3 2 1 0 = indexes

1 0 1 1 0 1 1 = 91

$swap(0, 2, 1011\overline{0}1\overline{1}) \rightarrow 1\ 0\ 1\ 1\ 1\ 1\ 0 = 94$

$swap(4, 5, 1\overline{0}\ \overline{1}1110) \rightarrow 1\ 1\ 0\ 1\ 1\ 1\ 0 = 110$

$swap(0, 1, 11011\overline{1}\ \overline{0}) \rightarrow 1\ 1\ 0\ 1\ 1\ 0\ 1 = 109$

$swap(2, 4, 11\overline{0}1\overline{1}01) \rightarrow 1\ 1\ 1\ 1\ 0\ 0\ 1 = 121$

$swap(0, 1, 111100\overline{0}\ \overline{1}) \rightarrow 1\ 1\ 1\ 1\ 0\ 1\ 0 = 122$

$swap(2, 4, 11\overline{1}1\overline{0}10) \rightarrow 1\ 1\ 0\ 1\ 1\ 1\ 0 = 110$

$swap(0, 5, 1\overline{1}01111\overline{0}) \rightarrow 1\ 0\ 0\ 1\ 1\ 1\ 1 = 79$

Note that Tobby can swap the same index many times, note also that the position 6 was never swapped.

Problem L. Tobby and Prime Sum

Input:	Standard
Output:	Standard
Author(s):	Carlos Arias Londoño - UTP Colombia

Tobby has learned to calculate the sum of the digits of a given number X, but this is very easy for him. Lately he has been studying about prime numbers and he has found a more challenging question:

How many integer numbers in the range from L to R (inclusive) exist such that the sum of their digits is a prime number?.

Tobby is an smart puppy but he can only count to 100, can you help him to solve this problem?

Input

The input consists of several test cases and must be read until EOF. The first line of each test case contains two integers L and R ($1 \leq L \leq R \leq 10^{500}$).

Output

For each test case the output consists of one number X indicating how many numbers in the range L, R meet the property previously mentioned. Because this amount can be very large you also should print the answer modulo $10^9 + 7$.

Example

Input	Output
1 10	4
20 46	11