Problem A. Print String

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

Print `"Hello, ProCo."` on a single line to standard output.

Input

There is no input for this problem.

Output

A single line containing the string `"Hello, ProCo."` without any trailing spaces or the surrounding quotes.

Problem B. Mental Sums

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

Mnemosyne likes to perform feats of mental arithmetic to show off her amazing memory. Today, she has invented a new mental exercise. She begins with a number x, written in base 10, squares each individual digit, and computes their sum. She repeats this process with the result until the number is only a single digit. Then, she recalls all her previous computations and reports the number of times she had to repeat this process.

For example, if the initial number is 103, the resulting number is $1^2 + 0^2 + 3^2 = 10$, and the next number is $1^2 + 0^2 = 1$, which is a single digit. Hence, her answer is 2. Amazingly, she does this all in her mind, and really quickly. Can you do better than Mnemosyne with a computer?

Let $f(x)$ be the answer to Mnemosyne's new puzzle if we begin with the number x. You are given two integers a, b, which define Q integers $x_1, \ldots, x_Q$, where $x_i = (a + b \times i) \mod (10^9 + 7)$. Find the answer $f(x_i)$ for each instance x_i of the puzzle.

Input

The first line contains a three integers Q, a, b, where $1 \leq q \leq 10^5$, and $0 \leq a, b \leq 10^9$.

Output

Output Q lines. Each line should contain a single integer, $f(x_i)$.

Examples

standard input	standard output

`3 9 99`	`9`
	`10`
	`12`

`1 2018 0`	`13`

Problem C. Necklace Constructions

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

Hephaestus has n straight metal rods of various lengths not necessarily distinct, and he's trying to use them to construct the largest necklace possible, for some notions of "large." He makes a necklace by taking some of the rods and attaching every end of every rod to exactly one other end of another rod until the rods "loop around" to form a single, connected polygon. Although the rods are infinitely thin, they cannot occupy the same space, i.e. the polygon must be non-degenerate.

Let the length of a necklace be the sum of the lengths of its constituent rods. Let S be the set of necklaces that he can construct. Help Hephaestus find the answer to the following two problems:

1. Find the maximum length of any necklace in S (output -1 if S is empty).
2. Find the maximum number of rods of any necklace in S (output -1 if S is empty).

Input

The first line contains a single integer n ($3 \le n \le 5 \times 10^5$), the number of rods.

For $1 \le i \le n$, the $i+1$-th line contains a single integer s_i ($1 \le s_i \le 10^9$), the length of the i-th rod.

Output

Two space-separated integers, where the first one is the maximum necklace length and the second one is the maximum number of rods in a necklace.

Examples

standard input	standard output
3 3 4 5	12 3
3 1 1 3	-1 -1

Problem D. Shrine

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

Within the ProColosseum, there is a shrine shaped like a $n \times n \times n$ lattice cube. An ant can walk on the surface of the cube only from one lattice point to an adjacent lattice point in one step (we do not allow diagonal movements).

Given two points on the surface of the cube, compute the minimum number of steps the ant needs to take in order to walk from one point to the other.

Input

The first line contains a single integer n ($1 \leq n \leq 5 \cdot 10^8$), the size of the cube.

The second line contains three space-separated integers x_1, y_1, z_1 ($0 \leq x_1, y_1, z_1 \leq n$). These are the coordinates of the starting point.

The third line contains three space-separated integers x_2, y_2, z_2 ($0 \leq x_2, y_2, z_2 \leq n$). These are the coordinates of the destination point.

Output

Print a single integer, the minimum number of steps for the ant to walk from the starting point to the destination point.

Examples

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

Problem E. Celestial Weapons

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

Hephaestus recently forged many Celestial bronze weapons, and he is selling them to demigods who are preparing for a war against monsters, first-come first-serve.

There are n demigods in a line entering Hephaestus's forge, where the ith demigod has a price threshold x_i. There are n Celestial bronze weapons in the forge, each with a price p_i. The 1st demigod is first in line, the 2nd is second in line, and so on, until the nth demigod is last in line. Demigods enter one by one, looking at the remaining weapons and buying the most expensive weapon they can afford (under or at their price threshold), since they want to use the greatest weapon available. If they cannot afford any of the remaining weapons, they will leave without buying anything and use their fists in war instead. Print the price of the weapon each customer buys, or -1 if they buy nothing.

Input

The first line contains a single positive integer n, where $1 \leq n \leq 10^5$. The second line contains n positive integers $x_1, x_2, ... x_n$ where all $1 \leq x_i \leq 10^9$. The third line contains n positive integers $p_1, p_2, ... p_n$ where all $1 \leq p_i \leq 10^9$.

Output

Print n integers, the ith of which is the price of the weapon the ith customer buys, or -1 if they buy nothing.

Example

standard input	standard output
4 10 3 39 6 7 100 29 50	7 -1 29 -1

Problem F. Papyrus

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

Athena, the Goddess of Wisdom, wants to record her knowledge on pieces of papyrus. Unfortunately, papyrus scrolls have a fixed height and can contain a maximum of K lines of writing. Fortunately, Athena has a large collection of scrolls, and can find a papyrus scroll of any width.

Athena wants to record a text on a scroll. The text that Athena wants to write consists of a series of words. Each word contains at most 100 characters, all of which are alphabetical letters (`a-z` or `A-Z`). For consistency, Athena follows a strict format for writing text on scrolls. For any scroll of width W, here are the rules:

1. All letters take up the same amount of horizontal space. A space has the same width as a letter. The width of a single letter is 1.
2. Each line contains a series of words. Adjacent words are separated by a single space. There are no spaces before the first word or after the last word on a line.
3. No word can be broken across multiple lines. No line can be empty.
4. The total width of all the letters and spaces on each line cannot exceed W.

Given the text and the maximum number of lines K, help Athena find the minimum width W required such that the text will fit in the papyrus scroll.

Input

The first line of the input will be an integer K, between 1 and 10^5, inclusive.

The second line will contain a integer N between 1 and 10^5, inclusive, indicating the number of words in the given text.

The third line will contain the corresponding N words, with a single space separating adjacent words. Each word will be at most 100 letters long.

Output

The first line of the output should contain two integers, separated by a single space. The first integer, W, should represent the minimum width of the papyrus scroll required. The second integer, L, must be the number of lines that the text will take up when written on a papyrus srcoll of width W.

The subsequent L lines should contain the words as how Athena would arrange them on a scroll of width W. (You can think of it as the text-wrapping function in text editors). Note that you should not print any extra spaces before or after each line.

Examples

standard input	standard output
3 7 little bee is flying over little flower	13 3 little bee is flying over little flower
2 3 knowledge is power	9 2 knowledge is power

Problem G. Gladiators' Language

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

Gladiators have a specialized language that is simple, having an alphabet consisting only of $\{a, b, c\}$, yet offers the opportunity to communicate emphasis through word transformations that preserve meaning. Given an initial string, determine whether it has the same meaning as a target string under the two phrase sets $A = \{a, bb, caababc\}$ and $B = \{bbbbbbbb, ab, bc, ca\}$.

Under a phrase set P, one transforms a string s to another string s' with the same meaning through a series of steps. Each step is either an insertion of a string $p \in P$ at a single location in s or a deletion of an appearance of $p \in P$ in s (this must be a deletion of contiguous letters). At any point in the process, the string may have length at most L.

As an example, with $L = 10$, "babc" has the same meaning as "c" under phrase set A, since we can delete the "a" and then the "bb", and it has the same meaning as "bbbbbbbb" under phrase set B, since we can delete the "ab" and then the "bc" and then add the "bbbbbbbb".

Input

The first line contains an integer L ($1 \leq L \leq 5 \times 10^5$) indicating the maximum string length that one may have at any time.

The second line contains the input string over the alphabet $\{a, b, c\}$. Its length is at most L.

The third line contains the target string over the same alphabet with length at most L.

Output

The first line should be a 1 or a 0 indicating whether the target string has the same meaning as the initial string under phrase set A.

The second line is the same as the first, except that it corresponds to using phrase set B instead.

Examples

standard input	standard output
3 abb a	1 1
10 aab cabcabb	0 1

Problem H. Zeus's Trap

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

Zeus is standing at the origin in the plane, continuously firing a perfectly straight, infinitely thin, and lethal lightning bolt in the direction of the lattice point (m, n) (which is not the origin). After initially aiming at (m, n), Zeus has turned the plane into a torus: The rectangle $(0, 0), (a, 0), (0, b), (a, b)$ has its opposite sides identified with matching orientation so that a point (x, y) is equivalent to another point (x', y') if $x - x'$ is a multiple of a and $y - y'$ is a multiple of b.

Our hero Odysseus is trapped in this plane and can only stand at lattice points. Zeus's lightning bolt will loop around the toroidal world infinitely, and it passes through Zeus without any change to Zeus or the lightning bolt. However, Odysseus will die if he's zapped by the lightning bolt. On how many distinct lattice points in this toroidal world (including the origin) would Odysseus die?

Input

The first line describes the size of the toroidal world: two space-separated integers a and b ($1 \le a, b \le 10^9$).

The second line has the direction in which Zeus aims: two space-separated integers m and n ($0 \le m, n \le 10^9$, $(m, n) \neq (0, 0)$). Notice that it is possible that $m \ge a$ and/or $n \ge b$.

Output

A single integer describing the number of deadly lattice points for Odysseus.

Examples

standard input	standard output
3 4 1 1	12
3 4 3 2	2
3 4 0 2	4

Problem I. Twisting Words (novice)

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

Hermes is very adept at twisting the words of others. The rest of the pantheon are annoyed about this, and are trying their best to prevent him from doing this. Help them figure out if Hermes will be able to twist their statements into something else, and if so, how he might do it.

As Hermes listens to people speak, he is constantly thinking of ways to twist each of their sentences. A sentence contains at most $1 \leq N \leq 15$ words, $s_1, \ldots, s_N$. For each word, Hermes can either keep it, omit it, or repeat it (have two copies of the word). Then, Hermes can rearrange all the letters in the words he chose in any order he likes. Note that Hermes cannot add new letters from words he has not chosen, nor can he omit letters from a word he has chosen.

Given such a sentence, determine if Hermes can twist it to arrive at a different statement T. Each word $s_1, \ldots, s_N$ and the entire target statement T contains only lowercase letters (a-z). If Hermes can, also find out whether he might keep, omit or repeat each original word.

Input

The first line contains an integer N, where $1 \leq N \leq 15$. The second line contains the N words in the order $s_1, \ldots, s_N$. Every two adjacent word are separated by a single space. You are guaranteed that each word does not exceed 100 characters in length, namely $1 \leq |s_i| \leq 100$. The last line contains the target string T, which is not more than 5000 characters long, so $1 \leq |T| \leq 5000$.

Output

On the first line, print `possible` or `impossible` depending on whether Hermes can form the target statement T with the given strings $s_1, \ldots, s_N$. If it is possible, print a second line containing N integers $k_1, \ldots, k_N$, representing what you think Hermes will do to arrive at T. $k_i = 0$ indicates Hermes will omit this word, and $k_i = 1$ and $k_i = 2$ represent keeping and repeating the word respectively. If there are multiple ways for Hermes to reach T by twisting the original sentence, print any one of them.

Examples

standard input	standard output
2 ab bek kebab	possible 1 1
2 ab keb kebaba	impossible

Problem J. Game

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

In order to test your wisdom, Athena challenges you to a two-player game. This game takes place on a 2D grid with N cells on each side, and begins in the starting cell (x, y), where $0 \leq x < N$ and $0 \leq y < N$.

From cell (x, y), there are six possible next cells each player can go to: $(x', y') \in \{(\lfloor x/2 \rfloor, y), (x, \lfloor y/2 \rfloor), (x-3, y-4), (x-4, y-3), (x-2, y-1), (x-1, y-2)\}$. However, no player is allowed to stay in the same cell, or leave the grid.

The first move is yours, and you and Athena take turns making moves. The player with no moves left loses. If Athena loses, she will reward you with $100000/k$ coins, where k is the number of moves it took before you won. If you lose, you must give $100000/k$ coins to charity. Your goal is to maximise the amount of money that you have. Athena's goal is to maximize the number of coins donated to charity. Even if she loses and has to reward you, she will always donate any remaining coins she has left to charity. If both of you play optimally, who wins, and in how many moves?

Input

The first line contains two integers, N, Q, representing the width and height of the grid, and the number of queries. You are guaranteed that $1 \leq N \leq 2000$, and $1 \leq Q \leq 2000$. The next Q lines contain two integers each. the ith line contains x_i, y_i, representing the starting cell of the ith game. You are guaranteed that $0 \leq x_i, y_i < N$.

Output

Print Q lines, each containing two integers. On the ith line, print w_i, k_i. w_i must be 1 if you win the ith game and 0 otherwise. k_i must be the minimum number of moves it takes for either you or Athena to win the game.

Example

standard input	standard output
3 2 2 2 1 2	0 2 1 1

Problem K. Sinkholes

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

Gaea is angry that a large part of her land has been fouled by pollutants, and has decided to cleanse and rebuild it by opening up several sinkholes on a 2D $R \times C$ grid. Each sinkhole begins as a single cell and grows to devour adjacent cells at a rate of one cell per second. Cells are adjacent if they share an edge. Of course, a cell that has already collapsed does not collapse again.

Unfortunately, you, an innocent soul, are currently on this patch of land! Gaea realizes this at the last moment and stablizes one of the grid cells. You need to get from your starting grid cell to the safe grid cell if you are to survive! Neither the cell which you are on nor the safe grid cell are sinkholes. What's the minimum speed s at which you need to run to be able to reach your target before it too collapses? To move at speed s means that you can move to an adjacent cell every $1/s$ seconds.

There are also some big trees in the land. You cannot run through these, and they also stabilize the ground so sinkholes will not collapse them. You may assume that sinkholes spread gradually: that is, as long as you arrive at a cell and leave it before a sinkhole expands to contain it, you are safe, otherwise you will fall into the hole.

Input

The first line contains two integers, R, C, where $1 \le R, C \le 1000$. The next R lines contain C characters each, representing the 2D grid. These characters are one of '`S`', '`T`', '`#`', '`!`', '`.`' denoting your starting location, your target safe location, a tree, a sinkhole and normal ground respectively. You are guaranteed that there is exactly one '`S`' and exactly one '`T`' in the grid.

Output

If you cannot reach safety before it collapses no matter how fast you run, print a single line with the word '`impossible`'. If you can walk as slow as you like and still reach your target, print a single line with the word '`stroll`'. Otherwise, print the minimum speed in decimal. Your answer must be accurate to 6 decimal places.

Examples

standard input	standard output
5 4 S#.. ..#. #.#! ..#. T#..	stroll
5 4 S#.. ..#. #..! ..#. T#..	1.5
5 4 ...S .!!! .... ###. T...	8.0
2 3 S#. #.T	impossible