General information on problems B, C, D, and E.

Memory limit

The limit is 256 MiB for each problem.

Source code limit

The size of each solution source code can't exceed 256 KiB.

Submissions limit

You can submit at most 50 solutions for each problem.

You can submit a solution to each task at most once per 30 seconds. This restriction does not apply in the last 15 minutes of the contest round.

Scoring

Each problem consists of several subtasks. The subtask score is awarded if all tests in the subtask are passed.

The number of points scored for the problem is the total number of points scored on each of its subtasks. The score for the subtask is the maximum number of points earned for this subtask among all the solutions submitted.

Feedback

To get feedback for your solution, go to "Runs" tab in PCMS2 Web Client and use "View Feedback" link. In each problem of the contest you will see the score for each subtask, or the verdict for the first failed test.

Scoreboard

The contestants' scoreboard is available during the contest. Use "Monitor" link in PCMS2 Web Client to access the scoreboard. The standings provided in PCMS2 Web Client are not final.

Problem A. Pawn Chess

Time limit: 1 second

Sergei was given the game «Pawn Chess». In this game, a chessboard of size $w \times h$ is provided. Black pawns and one white pawn are placed on it.

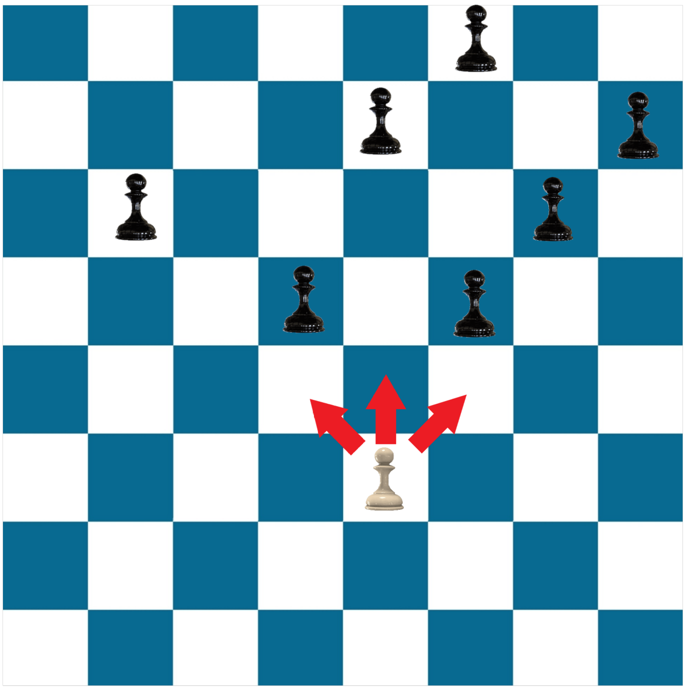
Black pawns cannot move, while the white pawn can move straight up (U), diagonally to the right and up (R), and diagonally to the left and up (L). In this game, a pawn can capture any black piece that is located one square directly above it or diagonally.

The player's task is to create a route for the white pawn to capture as many black pieces as possible.

The route is described by a sequence of letters:

- U — the pawn moves forward,
- R — the pawn moves diagonally to the right,
- L — the pawn moves diagonally to the left.

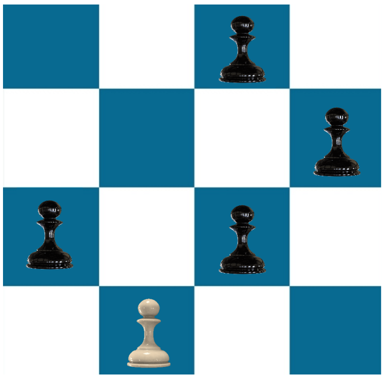
Рис. 1: For example, for the given board, the answer could be $URRR$.



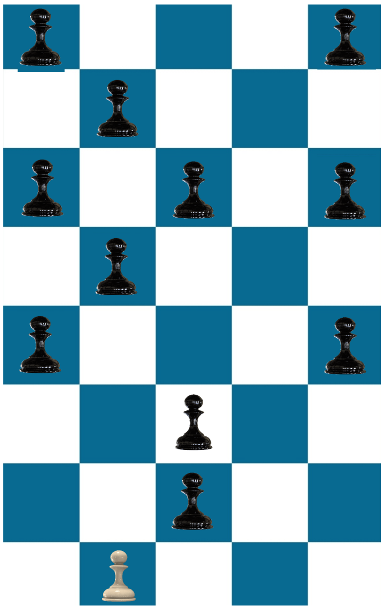
There are four test images for which a route needs to be found. If you don't know the answer for a test, write X. Attach the code or a **txt** file as the answer to this task. If you submit a **txt** file, the first line should contain the route for the first test, the second line for the second test, and so on. If you submit code, the first line of output should provide the route for the first test, the second line for the second test, and so forth.

For this task, up to three files are accepted for verification.

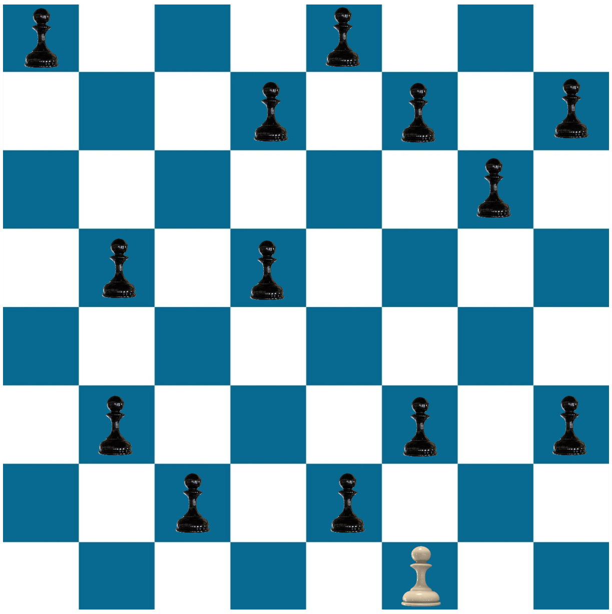
Each test is evaluated from 0 to 25 points according to the formula $25 \cdot \frac{points_{participant}}{points_{max}}$, where $points_{participant}$ is the number of pieces eaten by the participant's route, and $points_{max}$ is the number of pieces eaten by the jury's route. If the route contains symbols other than L, R, U, X or goes beyond the boundaries of the field, it will be scored 0 points.



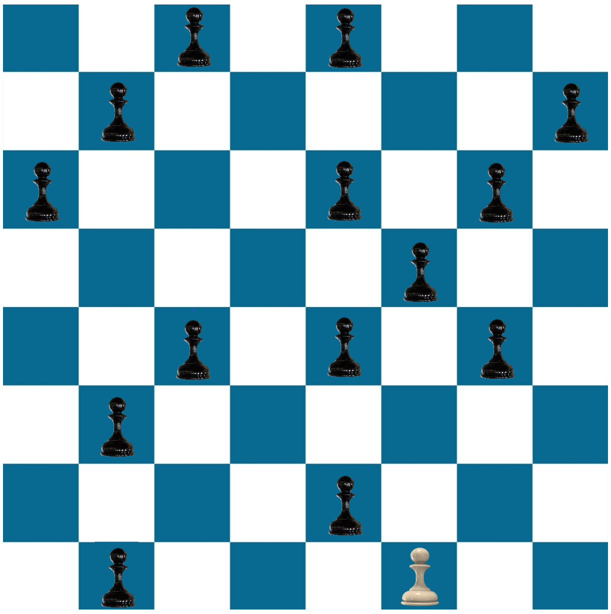
(a) Test 1



(b) Test 2



(c) Test 3



(d) Test 4

Test Number	Answer
1	(answer field)
2	(answer field)
3	(answer field)
4	(answer field)

Problem B. Circus

Time limit: 2 seconds

In honor of the circus's birthday, the clown Pilyulkin is giving out free tickets to children for the available seats left after ticket sales. The circus arena is divided into four sectors, labeled A, B, C, and D.

Pilyulkin distributes tickets according to the following principle: the first child receives a ticket in sector A, the second in sector B, the third in sector C, the fourth in sector D, the fifth in sector C again, the sixth in sector B, and so on (the sequence of sectors follows the pattern: $A{-}B{-}C{-}D{-}C{-}B{-}A{-}B{-}C{-}D{-}\ldots$).

As soon as Pilyulkin is unable to give a ticket for the next sector, he stops giving away free tickets and begins the performance.

How many children will be able to visit the circus for free if there is no other way to obtain a free ticket?

Input

In the first line of the input file, an integer A ($0 \le A \le 2 \cdot 10^{16}$), representing the number of tickets in sector A.

In the second line of the input file, an integer B ($0 \le B \le 2 \cdot 10^{16}$), representing the number of tickets in sector B.

In the third line of the input file, an integer C ($0 \le C \le 2 \cdot 10^{16}$), representing the number of tickets in sector C.

In the fourth line of the input file, an integer D ($0 \le D \le 2 \cdot 10^{16}$), representing the number of tickets in sector D.

Output

Output the number of tickets that Pilyulkin will be able to distribute before the start of the performance.

Scoring

Points for each subtask are awarded only if all tests for that subtask and its required subtasks are successfully passed.

Subtask	Points	Constraints	Required Subtasks	Check Information
0	0	Tests from the statement	—	Full
1	15	$A = B = C = D \le 10^6$	—	First error
2	25	$A, B, C, D \le 10^6$	1	First error
3	15	$A = B = C = D \le 2 \cdot 10^{16}$	1	First error
4	45	No additional constraints	1 — 3	First error

Examples

standard input	standard output
1 1 1 1	4
1 2 3 4	6
1 2 2 4	6
23 10 3 99	10

Problem C. Words

Time limit: 2 seconds

You are practicing touch typing, and today your goal is to type a sequence of m words. To achieve this, you can choose any m words from a set $s_1, \ldots, s_n$ ($1 \leq n \leq 2 \cdot 10^5$), typing them separated by spaces. Each word can be selected at most once.

Additionally, each word in the set may contain only the first k ($1 \leq k \leq 26$) letters of the Latin alphabet.

You know the time it takes to move your finger from one key to another on the keyboard. Formally, you have a matrix $t_{i,j}$ ($1 \leq i, j \leq k$), where $t_{i,j}$ represents the time it takes to move the finger from the i-th symbol to the j-th symbol. In this task, we will ignore the time for moving the finger from a symbol to a space and vice versa.

Thus, to type two words «ab» and «b», you need to move your finger from «a» to «b», spending $t_{1,2}$ time. Then, you instantly move your finger from «b» to the space and immediately move it from the space to «b». Therefore, the total time required is $t_{1,2}$.

Your goal is to type m words from the set in such a way as to minimize the total time spent. Find this minimum time.

Input

The first line contains a natural number n ($1 \leq n \leq 2 \cdot 10^5$) — the number of words in the set.

The second line contains n strings $s_1, \ldots, s_n$, each consisting of lowercase Latin letters.

The third line contains a natural number m ($1 \leq m \leq n$) — the number of words that need to be selected for typing.

The fourth line contains a natural number k ($1 \leq k \leq 26$), specifying the set of the first k Latin letters from which words can be composed.

The following lines contain a matrix $t_{i,j}$ ($1 \leq t_{i,j} \leq 10^9$) of size $k \times k$, consisting of non-negative integers.

It is guaranteed that the total length of all words does not exceed $2 \cdot 10^5$.

Output

Output a single number — the minimum time required to type m words.

Scoring

Points for each subtask are awarded only if all tests for that subtask and its required subtasks are successfully passed.

Subtask	Points	Constraints	Required Subtasks	Check Information
0	0	Tests from the statement	—	Full
1	20	$k = 1$	—	First error
2	20	Sum of lengths of all words is no more than 1000	—	First error
3	30	$n \leq 1000$	2	First error
4	30	No additional constraints	$1 - 3$	First error

Examples

standard input	standard output
2 ab ba 1 2 0 1 2 0	1
3 a aa aaa 2 1 5	5

Problem D. Pawn-chess

Time limit: 2 seconds

Sergey received the game "Pawn Chess"as a gift. In this game, there is a chessboard of size $w \times h$. The cell $(1; 1)$ is located in the bottom left corner, and black pawns are placed on it.

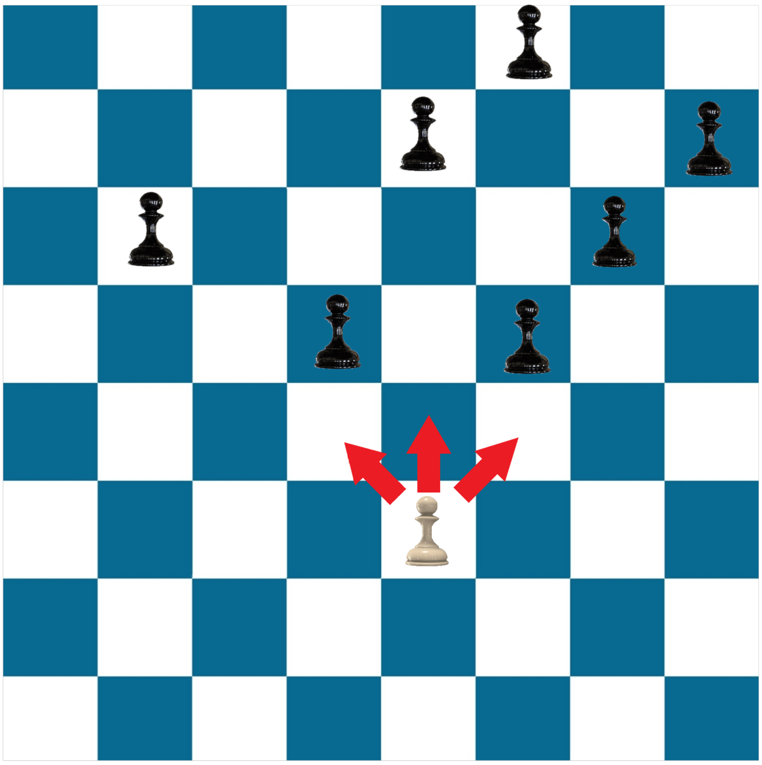
Black pawns cannot move, while the white pawn can move straight up (U), diagonally to the right and up (R), and diagonally to the left and up (L). In this game, a pawn can capture any black piece located directly in front of it, one square straight up or diagonally.

The player's task is to choose where to place the pawn and create a route for the white pawn so that it captures as many black pieces as possible.The white pawn cannot be placed on the field where there is a black pawn.

The route is described by a sequence of letters:

- U — the pawn moves forward,
- R — the pawn moves diagonally to the right,
- L — the pawn moves diagonally to the left.

Рис. 2: For example, for the given field, the answer would be $URRR$.



Input

The input file begins with two integers, h and w $(1 \leq w, h \leq 2 \cdot 10^5)$, representing the dimensions of the chessboard.

It is guaranteed that $w \cdot h \leq 2 \cdot 10^5$ and that there is an empty cell.

In each of the following h lines, there is a string of w characters, a_{ij} $(a_{ij} \in \{*, B\})$, where:

- $*$ represents an empty cell,

- B indicates a black pawn on the corresponding cell.

Output

Print two integers on the first line representing the coordinates to place the pawn (row number and column number).

On the second line, print the number of captured pawns.

If the pawn has captured at least one piece, then on the third line, output the route for the white pawn to capture the maximum number of black pieces.

If there are multiple valid answers, you can output any of them.

Scoring

Points for each subtask are awarded only if all tests for that subtask and its required subtasks are successfully passed.

Subtask	Points	Constraints	Required Subtasks	Check Information
0	0	Tests from the statement	—	Full
1	30	$h, w \leq 10$	—	First error
2	35	$h, w \leq 30$	1	First error
3	35	No additional constraints	1 — 2	First error

Example

standard input	standard output
8 8 *****B** ****B**B *B****B* ***B*B** ******** ******** ******** ********	4 3 3 RURR

Problem E. Playlist

Time limit: 2 seconds

Unicorn Sparks loves listening to his playlist on the VK social network. To add variety, Sparks chooses a new order for the songs each time, and this time is no exception. Sparks decided to select an order such that the duration of the songs is non-decreasing at the beginning and non-increasing afterward. Help Sparks find the number of ways to shuffle the songs to make the playlist interesting for him.

More formally: given an array $a_1, \ldots, a_n$ ($1 \le a_i \le 10^9$), where a_i represents the duration of the i-th song. Find the number of ways to rearrange the elements of array a such that there exists an index k satisfying $a_i \le a_{i+1}$ for all $1 \le i \le k-1$, and $a_i \ge a_{i+1}$ for all $k \le i \le n-1$.

As the answer can be very large, output it modulo $10^9 + 7$.

Input

The input file starts with an integer n ($1 \le n \le 2 \cdot 10^5$), representing the number of songs in Sparks' playlist.

The second line of the input file contains n numbers $a_1, \ldots, a_n$ ($1 \le a_i \le 10^9$), where a_i is the duration of the i-th song.

Output

Output a single number, the answer to the problem modulo $10^9 + 7$.

Scoring

Scores for each subtask are awarded only if all tests for that subtask and its required subtasks are successfully passed.

Subtask	Scores	Constraints	Required Subtasks	Check Information
0	0	Tests from the statement	—	Full
1	10	$n \le 10$	—	First error
2	20	All a_i are pairwise distinct	—	First error
3	20	$n \le 20$	1	First error
4	50	No additional constraints	1 — 3	First error

Examples

standard input	standard output
3 1 2 3	4
3 2 1 1	6