TASK	TETA	KRIZA	ACM	JANJE	PROSJEK	POLICE
input	standard input (*stdio*)					
output	standard output (*stdout*)					
time limit	1 second	1 second	1 second	1 second	1 second	1 second
memory limit	64 MB	64 MB	64 MB	64 MB	64 MB	64 MB
score	**50**	**80**	**100**	**120**	**140**	**160**
	total 650					

You have found yourself in the role of a nice lady working as a cashier in a canteen. One of the multiple reasons why the lady is considered nice by all students is because of her concern that you spend as less as possible when visiting the canteen.

How does she do that? Well, the strategy is really simple. Various meals can be bought in the canteen and their prices are known. Each day, a **menu** is offered. A menu includes 4 meals (usually it's soup, main course, side dish and dessert), but it's price is less than or equal to the sum of prices of its components. When the lady notices that you'd spend less money if she charged you with an entire menu instead of individual things from the menu you took, then she will do so, and you will leave full and with more money in your pocket.

You are standing in front of the cash register with your tray and want to know how much you have to pay. Write a programme to determine it!

Please note: The lady can charge you with multiple menus if that comes out cheaper.

INPUT

The first line of input contains the integer K ($1 \leqslant K \leqslant 20$), the number of meals that can be bought in the canteen. For simplicity's sake, we will denote the meals with integers from 1 to K.

The second line of input contains K integers, the i^{th} number representing the price of the meal denoted with i. The prices will be in range [1, 250].

The third line of input contains the integer X ($1 \leqslant X < 1000$), the menu price.

The following line of input contains 4 integers, the labels of the meals from the menu, different from each other.

The fourth line of input contains the integer T ($1 \leqslant T \leqslant 20$), the number of meals on your tray.

The following line contains the list of meals you've taken. Not all meals on the tray have to be distinct, it is possible to take multiple portions of a meal.

OUTPUT

The first and only line of output must contain an integer, the cost of your meal.

SAMPLE TESTS

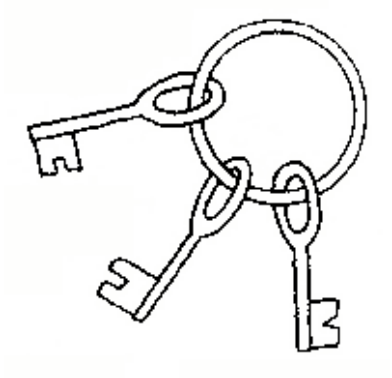
input	**input**
7 10 6 8 9 4 5 3 14 1 2 3 4 5 1 3 4 6 7	6 12 5 7 8 9 3 14 4 3 1 2 5 1 2 1 6 6
1em**output**	1em**output**
22	32

Clarification of the first example: You took meals 1, 3 and 4 that are part of a menu and the lady will charge you with an entire menu. Meals 6 and 7 aren't part of a menu, so you pay their full price.

Clarification of the second example: You took two portions of meal 1 (Wiener Schnitzel), but the lady will charge you one portion along with one portion of meal 2 as a menu, but will charge the other portion separately (because it doesn't pay off to charge it as a second menu). You took two juices (meal 6), but you pay for them separately because they are not a part of the menu.

The state of the economy is bad, the crysis is striking the country and people are losing their jobs. However, Sisyphus, the main hero of this task, has found himself a new job. Starting next monday, Sisyphus will be an assistant locksmith in a hotel. Naturally, first he needs to demonstrate his locksmithing abilities to the head locksmith.

This is why the head locksmith has given Sisyphus N keys on a big **round** pendant, blindfolded him and led him into a big room. That room contains N **locked** doors, denoted with numbers from 1 to N. Each of the keys on the pendant unlocks precisely one door.



Sisyphus' job is to unlock and lock each door again. He does this in a way that he moves along the wall, **not changing direction**, until he reaches a door. When he reaches a door, he tries unlocking it using the first (leftmost) key. In the case when the key doesn't unlock the door, Sisyphus moves it to the other (right) side of pendant and repeats this procedure until he finds the right key. His work is done when he goes through all the doors. The first door Sisyphus unlocked is denoted with 1, the next with 2, the one after with 3 and so on...

What Sisyphus doesn't know is that the head locksmith is, in fact, testing his endurance so he led him into a circular room. Therefore, Sisyphus will, after unlocking and locking the last door, go unlocking and locking the first door again. Because he's a hardworking and persistent fellow, Sisyphus has been doing this job for hours and hours without saying a single word. Only after the K^{th} successful unlocking and locking of a door he spoke: "If only I knew how many times so far I've put a wrong key in a door's lock!" Your task is to answer his question!

INPUT

The first line of input contains the integers N ($1 \leqslant N \leqslant 10^5$) and K ($1 \leqslant K \leqslant 10^9$) from the task.

The i^{th} of the following N lines contains the integer v_i ($1 \leqslant v_i \leqslant N$) which denotes that the i^{th} key on the pendant (from left to right) unlocks the door v_i.

OUTPUT

The first and only line of output must contain an integer representing the answer to Sisyphus' question.

SCORING

In test cases worth 40% of total points, it will hold $1 \leqslant N, K \leqslant 1\,000$.
In test cases worth 60% of total points, it will hold $1 \leqslant K \leqslant 50\,000$.

SAMPLE TESTS

input	**input**	**input**
3 5 1 2 3	4 6 4 2 1 3	10 7 1 3 2 4 5 7 6 8 9 10
output	**output**	**output**
4	13	25

Clarification of the second example:
The first locking/unlocking (door 1) – keys (left to right): 4 2 1 3
The second locking/unlocking (door 2) – keys (left to right): 1 3 4 2
The third locking/unlocking (door 3) – keys (left to right): 2 1 3 4
The fourth locking/unlocking (door 4) – keys (left to right): 3 4 2 1
The fifth locking/unlocking (door 1) – keys (left to right): 4 2 1 3
The sixth locking/unlocking (door 2) – keys (left to right): 1 3 4 2
The misplaced keys are underlined.

The team of the University of Zagreb – Stjepan, Ivan and Gustav – are taking part in the World Finals of the ACM International Collegiate Programming Contest in Morocco. Their technical guide Goran has come up with an invincible strategy to use for solving tasks in the finals.

In the very beginning, each team member quickly estimates the difficulty of each of the **N** tasks. The difficulties are described by numbers from 1 to 5, and their meaning is the following:

- 1 - hehehe
- 2 - bring it on!
- 3 - well OK.
- 4 - hmmmm...
- 5 - are you insane?

After this, they will distribute the tasks between them. For simplicity's sake, the array of tasks will be split into **three parts** so that each team member gets a **nonempty** array of **consecutive** tasks to contemplate about. The distribution is made so that the **sum of estimated difficulties** is minimal, whereas only the estimate of the team member whom that task is assigned to is calculated. Your task is to calculate that minimal possible sum.

INPUT

The first line of input contains the integer N ($3 \leqslant N \leqslant 150\,000$), the number of tasks.

Each of the following three lines contains N integers (from 1 to 5): the estimated task difficulties, given respectively. First of those lines corresponds to Stjepan's estimates, the second to Ivan's and the third to Gustav's.

OUTPUT

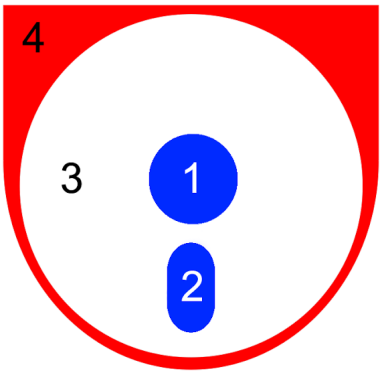
The first and only line of output must contain the minimal sum of difficulties.

SAMPLE TESTS

input	**input**
3 1 3 3 1 1 1 1 2 3	7 3 3 4 1 3 4 4 4 2 5 1 5 5 4 5 5 1 3 4 4 4
output	**output**
4	19

Clarification of the first example: Stjepan gets the first, Gustav the second, and Ivan the third task.

Young Bojan, today a successful student of electrical engineering, loved coloring ever since he was a little boy. Remembering careless days from his childhood, he decided to buy a coloring book and K colors and get to work. It's interesting that Bojan doesn't like colorful pictures, so he decided to color each picture using **at most three** different colors. Additionally, Bojan will never color two adjacent areas using the same color because, as he puts it, "what's the use of this line in between then?" Two areas are considered adjacent if their edges have at least one joint point. For example, areas denoted with 4 and 3 (see image below) are adjacent, whereas areas 1 and 2 aren't. Additionally, coloring of the image below is in accordance with all of Bojan's demands.



Before he begins coloring a picture, Bojan asks himself in how many ways he can color that picture so he meets with all his conditions. Given the fact that Bojan is studying electrical engineering, it is understandable that combinatorics isn't his strong point, so he asked you for help.

INPUT

The first and only line of input contains two integers N ($1 \leqslant N \leqslant 8$) and K ($1 \leqslant K \leqslant 1\,000$) which denote the ordinal number of the picture from the coloring book and the number of different colors Bojan can use, respectively.

You can find the coloring book with the numbered pictures on the next page.

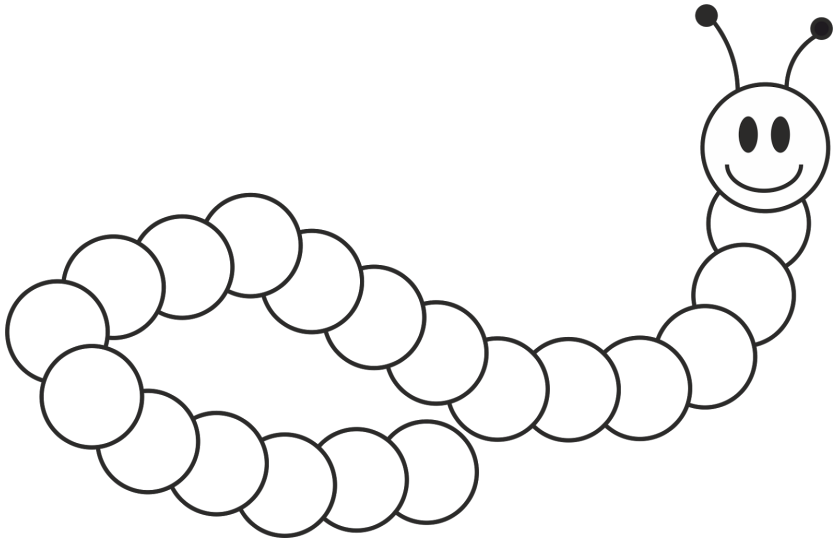
OUTPUT

The first and only line of output must contain the number of ways Bojan can color the N^{th} picture from the coloring book if he has K different colors at his disposal. Two colorings are different if they differ in color in at least one area.

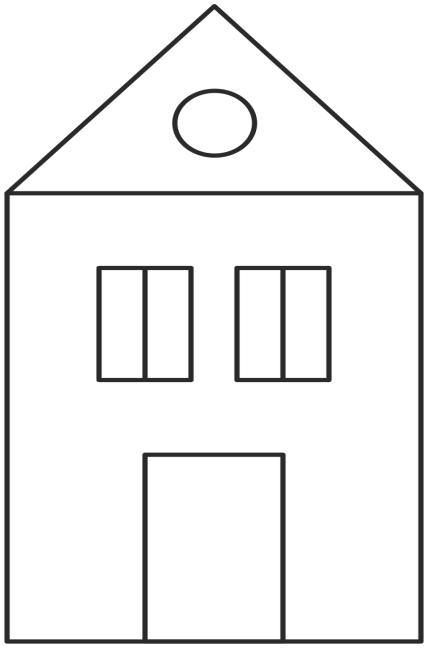
SAMPLE TESTS

input	**input**	**input**
2 2	5 3	7 3
output	**output**	**output**
0	12	96

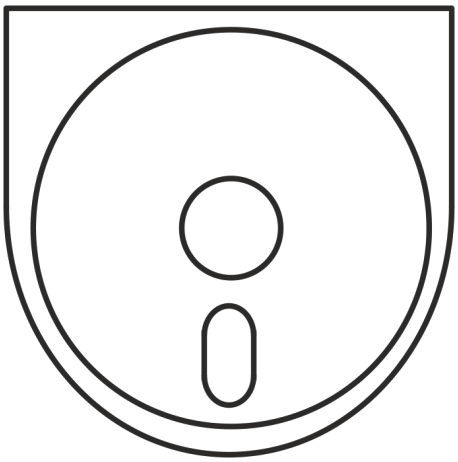
HAPPY COLORING BOOK



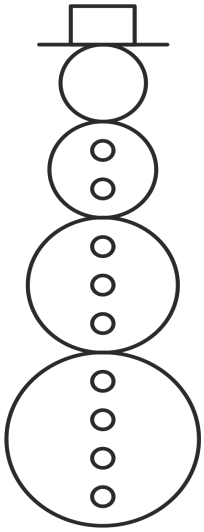
Picture 1. Caterpillar (the eyes and the antennas are not areas and should not be colored)



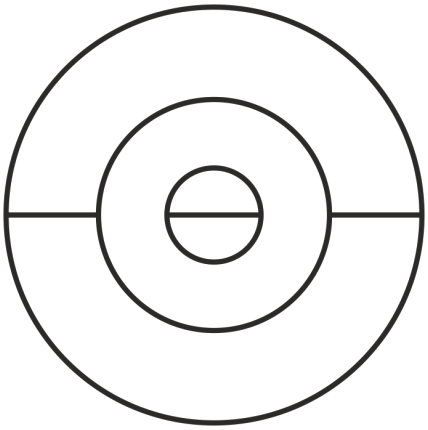
Picture 2. A sea cottage



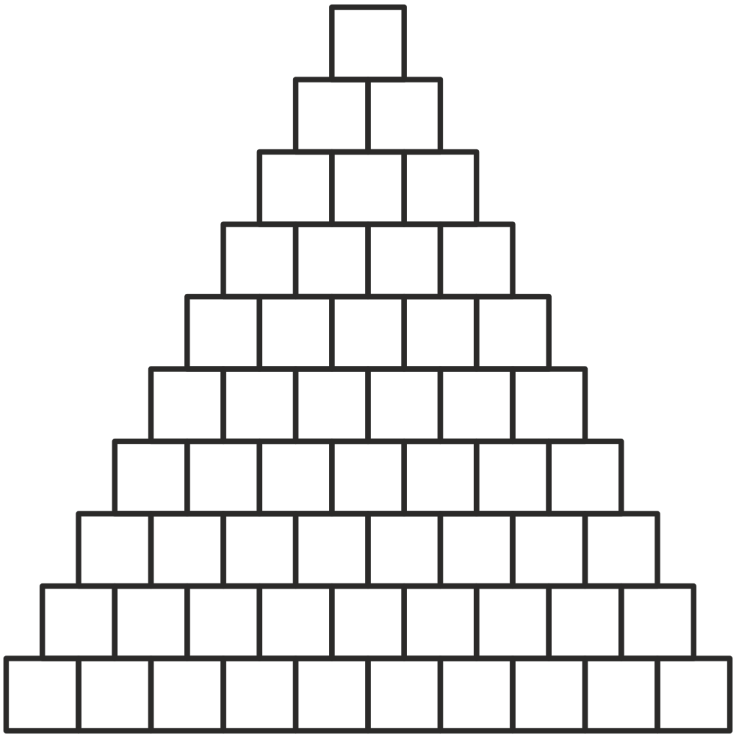
Picture 3. A well known logo



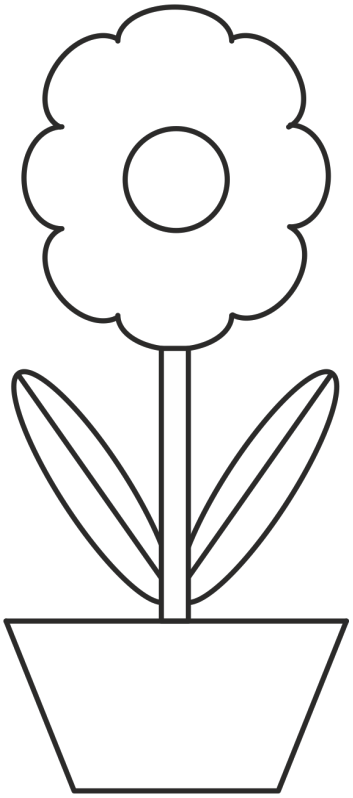
Picture 4. Frosty the Snowman



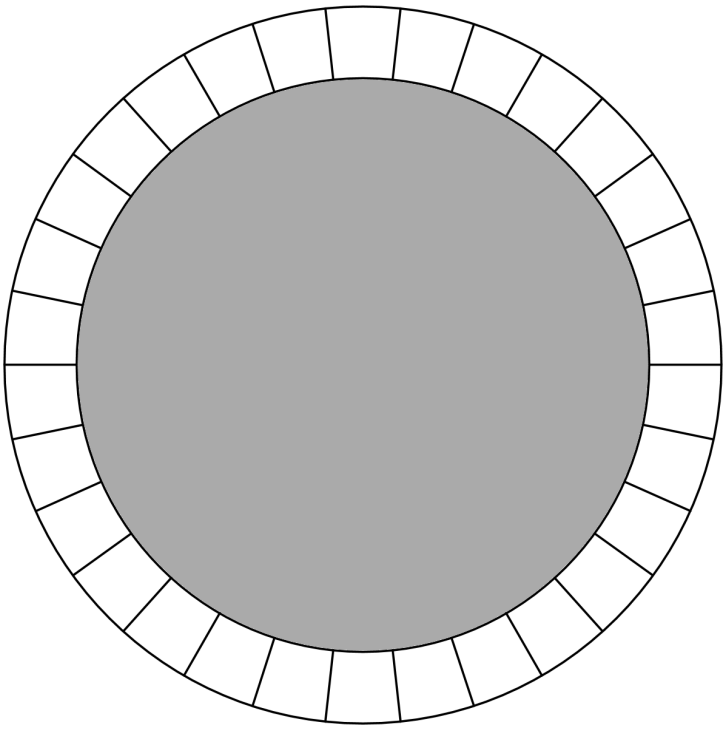
Picture 5. An abstract ball



Picture 6. Pyramid



Picture 7. Daisy



Picture 8. Trampoline (the gray part is not considered an area and should not be colored)

You are given an array of N integers. Find a consecutive subsequence of numbers of the length at least K that has the maximal possible average.

Please note: the average of a subsequence is the sum of all the numbers in the subsequence divided by its length.

INPUT

The first line of input contains two integers N ($1 \leqslant N \leqslant 3 \cdot 10^5$) and K ($1 \leqslant K \leqslant N$). The second line of input contains N integers a_i ($1 \leqslant a_i \leqslant 10^6$).

OUTPUT

The first and only line of output must contain the maximal possible average. An absolute deviation of ± 0.001 from the official solution is permitted.

SCORING

In test cases worth 30% of total points, it will hold that N is not larger than 5 000.

SAMPLE TESTS

input	**input**	**input**
4 1 1 2 3 4	4 2 2 4 3 4	6 3 7 1 2 1 3 6
output	**output**	**output**
4.000000	3.666666	3.333333

Librarian Jurica has N shelves in his library, and each shelf can contain M books. Jurica is a good librarian so he decided to make an inventory in the library and, if it's necessary, return the books that aren't in their place to their right place. He moves the books in the following way:

- moving the books one place to the left or to the right on a shelf if the place to the left or to the right is available,
- taking a book from a shelf and placing it to an available place on that or any other shelf.

Careful Jurica can't move books if he has a book in his hands. Additionally, he can't take more than one book at once.

Jurica has been having back pains ever since he had to move all the volumes of the printed edition of Wikipedia from the first to the second floor so now he wants to put all the books in place with **as little lifting** as possible because his back is hurting. What is the minimal number of lifting he needs?

INPUT

The first line of input contains the integers N and M ($1 \leqslant N \leqslant 1\,000$, $1 \leqslant M \leqslant 1\,000$).

Each of the following N lines contains M integers, the i^{th} line describing the current state of the i^{th} shelf.

Number 0 denotes an empty place on the shelf, and a number different than 0 denotes that there is a book in that place denoted with that number. All books are denoted with different numbers from 1 to K, where K is the total number of books on the shelves. After that, an additional N lines follow, each containing M integers, the i^{th} line describing the wanted state of the i^{th} shelf.

In the initial and final state of the shelves, the same books will appear.

OUTPUT

The first and only line of output must contain the required minimal number of lifting or -1 if it is impossible to arrange the books in the aforementioned way.

SCORING

In test cases worth 50% of points in total, each book will be on the same shelf in the initial and final state.

SAMPLE TESTS

input	**input**	**input**
2 4 1 0 2 0 3 5 4 0	3 3 1 2 3 4 5 6 7 8 0 4 2 3 6 5 1 0 7 8	2 2 1 2 3 4 2 3 4 1
output	**output**	**output**
2	4	-1

Clarification of the first example: Jurica will move book 1 one place to the right, then lift book 2 and move it over to the first place of the first shelf. He lifts book 5 and places it on the fourth place on the second shelf.