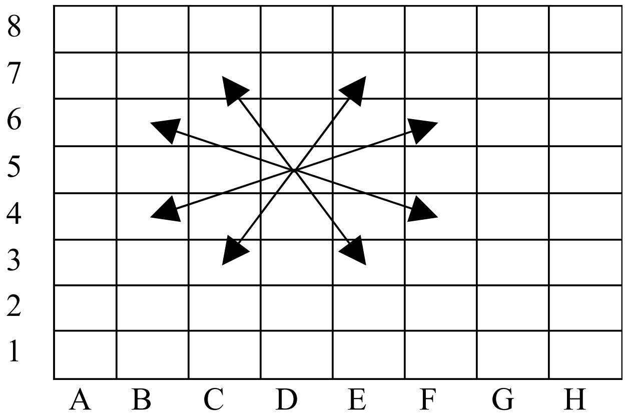
Contest Duration 2hour 30min
For All Problems
Time Limit: 2sc
Memory Limit: 64Mb

Problem A: Chess Knight



A Chess knight is placed on one of the chess cells. Every second the knight makes one move. Remember that the chess knight moves as shown in the picture.

If the knight is placed in D5 then after one second it can be in one of the following cells C7, E7, F6, F4, E3, C3, B4, B6.

You are given a chess cell where the knight is placed, you must count the number of cells where the knight can appear after N seconds.

Input Specification

On the first line of the input you are given a chess cell where the knight is placed. Each cell has a corresponding letter and number The Letter is a capital English letter from A to H and the number is an integer number from 1 to 8.
The Second line contains one integer number N ($0 \le N \le 10^9$), that is the number of seconds.

Output Specification

Output number of cells where can be knight after N seconds.

Sample Input	Sample Output
D5 1	8
A1 2	10

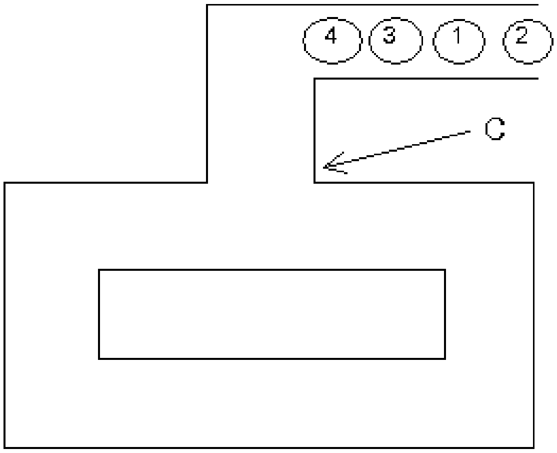
Author: Eduard Piliposyan

Problem B: Balls

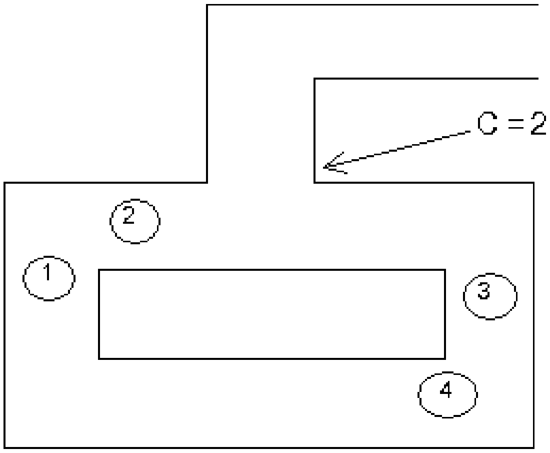
For sorting bowling game balls, one famous bowling club in Yerevan constructs his own Balls Sorting Machine. Machine works as follows: Balls enters to the machine one by one from the upper right corner, then fall down.In point C there is the Checker, which checks if the weight of ball is smaller or equal to the given number C, then the ball goes to the left, otherwise it goes to the right. Finally the sorted balls appear at the bottom of the machine.

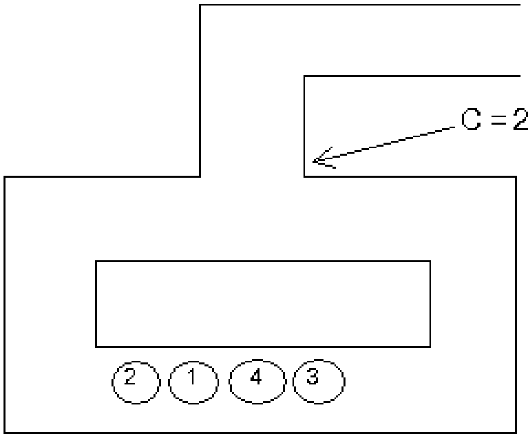
As shown on Picture 1, we have balls 4,3,1,2. After the machine works with C=2 we get the sorted sequence of balls: 2,1,4,3 (as shown in picture 3).

Picture 1



Picture 2





Picture 3

You are given balls sequence before machine work, and sorted sequence after machine work. You must find smallest non negative integer number C, that can be used as Checker number for getting from given started sequence, given sorted sequence.

Input Specification

First line contains one integer number N , $1 \le N \le 100000$, number of balls. Second line contains N integer numbers A_i separated by spaces $1 \le A_i \le 100000$, initial sequence of balls. Third line contains N integer numbers B_i separated by spaces $1 \le B_i \le 100000$, sorted sequence of balls after machine work.

Output Specification

Output smallest integer number C. It is guaranteed that there always exists such number.

Sample Input	Sample Output

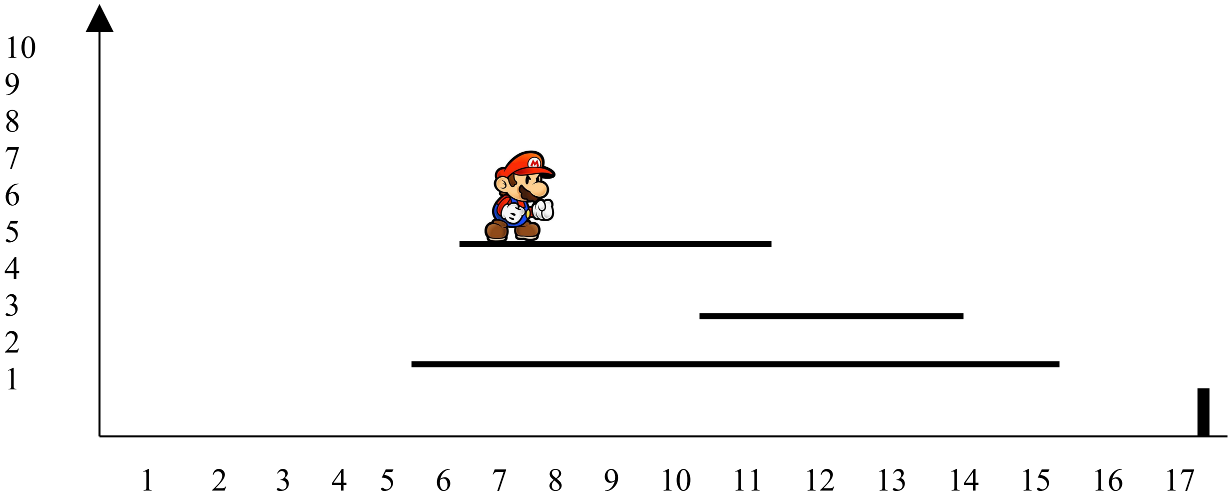
4
4 3 1 2
2 1 4 3
	2

Author: Eduard Piliposyan

Problem C: Mario

You are playing the famous game “Mario”. The goal of Mario is to reach the finish as soon as possible. There are N none moveable lifts. Each lift is parallel to the X axis (sea picture).

Mario is standing on one of the lifts. He runs with constant velocity V1, and falls down with constant velocity V2. At any point (except when Mario is falling down) he can decide whether to go to the left or to the right. Mario knows that the finish is located at point (F, 0).



For the pictures example V1 = 2 m/sc, and V2 = 5 m/sc, Mario can go left 1 metre, then fall down 3 metres, then go right until reaching point (15;2) , then fall down 2 meters, and then go to the right until reaching the finish. It will take him 1/2 + 3/5 + 9/2 + 2/5 + 2/2 = 7.0 seconds. But there is a better way. Mario can go right till point (11; 5), then fall down 2 metres, then go right till (14; 3), then fall 1 meter, then go right till (15;2) then fall down 2 metres, and then go right until reaching the finish. It will take him 6.0 seconds.

You must help Mario to get to the finish as soon as possible, i.e. you must find the minimal period of time needed to get to the finish.

Input Specification

The First line contains 2 integer numbers X_S, Y_S - Mario’s coordinates. It is guaranteed that Mario is standing on one of the lifts.

The next line contains four integers N, F, V_1, V_2 $1 \le N, F, V_1, V_2 \le 100000$.

The next N lines describe lifts. Each line contains 3 numbers Y, X_1, X_2 $0 \le Y \le 100000$, $0 \le X_1 < X_2 \le F$. Y is the height of the lift, X_1 is the left endpoint x coordinate of the lift, and X_2 is the right endpoint x coordinate.No two lifts intersects.

Output Specification

Output the minimal time that is needed to get to the finish, with 3 digits after decimal point.

Sample Input	Sample Output

7 5
3 17 2 5
5 6 11
3 10 14
2 5 15
	6.000

Author: Eduard Piliposyan

Problem D: HTML Tables

Every day, someone is registering in ArmCoder and ArmCo der team must update the information about contestants. X (the member of ArmCoder team) is the one responsible for this work.

One day he accidentally deleted all the information, except HTML Tables, which keep information about contestants. Now you must help him to convert tables to the text.

Example of HTML Table with the information of number 1 contestant, first name Narek, last name Saribekyan, RAU university, age 18 and with photo.

```
<tr>
<td> <p align=”center”>1</p></td>
<td> <p align=”center”>Narek</p></td>
<td> <p align=”center”>Saribekyan</p></td>
<td> <p align=”center”>RAU</p></td>
<td> <p align=”center”>18</p></td>
<td> <p align=”center”><img scr=”1.jpg”></p></td></tr>
```

So, it means that the text will be

#1 Narek Saribekyan, RAU, 18 years old, photo included (1.jpg).

If there isn’t a photo, the last line will be

```
<td> <p align=”center”>No Photo</p></td></tr>
```

And output will be

#1 Narek Saribekyan, RAU, 1 8 years old, photo not included.

Input Specification

You will be given HTML Tables . Input ends with “##”. Input doesn’t exceed 25 kb.

Output Specification

Output the text version of HTML Tables.

Sample Input

```
<tr>
<td> <p align=”center”>3</p></td>
<td> <p align=”center”>Narek</p></td>
<td> <p align=”center”>Saribekyan</p></td>
<td> <p align=”center”>RAU</p></td>
<td> <p align=”center”>18</p></td>
<td> <p align=”center”><img scr=”img_po.jpg”></p></td></tr>
<tr>
<td> <p align=”center”>8</p></td>
<td> <p align=”center”>Hayk</p></td>
<td> <p align=”center”>Saribekyan</p></td>
<td> <p align=”center”>Quantum</p></td>
<td> <p align=”center”>15</p></td>
<td> <p align=”center”>No Photo</p></td></tr>
##
```

Sample Output

```
#3 Narek Saribekyan, RAU, 18 years old, photo included (img_po.jpg).
#8 Hayk Saribekyan, Quantum, 15 years old, photo not included.
```

Author: Vartan Davtian

Problem E: Farm

There are N cows standing on a line, numbered from 1 to N (1≤N≤10000), and each cow has a price Wi. Two farmers are trying to buy cow 1, but there is a rule on the farm, that a farmer can buy a cow only if all other cows with numbers greater than it are already bought. And the other rule is that no farmer can buy more than K cows at a time (1≤K≤100). So, two farmers are competing against each other and both want to buy the first cow, and they will do everything to reach their aim. Also they are very smart, and are playing with optimal strategy to buy the first cow. The farmer, who has a winning strategy, tries also to minimize the sum paid, and the farmer who hasn't got a winning strategy, tries to maximize the sum paid by his opponent. So, you are to find who will buy the first cow and how much he will have to pay for that. Remember that at every step a farmer can buy from 1 to K cows inclusively – the cows numbered X-K+1 ... X, where X is the remaining number of cows.

Input Specification

First line of input contains two integer numbers N and K separated by space. In the second line there are N space separated integers Wi (Wi≤1000).

Output Specification

Output 1 on the first line if the farmer who plays first wins and 2 otherwise, when they both are playing with the strategy described above. On the second line output the amount of the money that the winner will pay.

Sample Input	Sample Output
3 2 10 20 15	2 30

Author: Aram Shatakhtsyan

Problem F: Monkey

A hungry monkey wants to eat banana. The monkey and banana are in a labyrinth consisting of rooms and corridors connecting them. Rooms can be in one of two states: locked or unlocked. If a room is locked, then the monkey cannot enter that room, but can leave it. Unlocked rooms can be freely entered and left.

Some rooms contain a switch. Pressing a switch changes states of a group of rooms, i.e. locked rooms from that groups become unlocked and vice versa. Same switch always changes states of the same group of rooms.

Upon entering a room having a switch, the monkey can press it if he chooses to do so.

Write a program that will help the monkey to find the room with banana as soon as possible, i.e. to find the smallest number of corridors the monkey has to pass to find the room with banana, possibly by pressing some of the switches.

Input data contain description of rooms and corridors in the labyrinth, initial states of all rooms, a list of switches and for each switch a list of rooms whose states can changed by it.

Input Specification

The first line contains two integers N, total number of rooms (1≤N≤100), and S, number of switches, i.e. number of rooms having switches, in labyrinth (1 ≤ S ≤8). Switches are located in the rooms with numbers from 1 to S.

The next N lines contain descriptions of rooms. The room number i is described with (i+1)th line which begins with either **0** (if the corresponding room is initially unlocked) or **1** (if initially locked). An integer K follows, total number of rooms connected via corridors with the room being described by that line. After it follows K numbers denoting those K rooms. A space character separates successive numbers in same line.

The following S lines describe switches, from first to Sth switch.

Each of those lines begins with an integer L, the number of rooms in a group whose states can be changed by switch being described. Following are L numbers – numbers of rooms in the group.

The last line contains two numbers A and B; A is number of a room where the monkey starts its search for banana and B is number of room containing banana.

Output Specification

The first and only line of output should contain the smallest number of corridors the monkey has to pass to find banana.

Note: Each test data will have a solution, i.e. there will always be a way to come to room B from room A.

Sample Input	Sample Output

4 1
0 1 3
1 2 3 4
0 2 1 2
0 1 2
1 2
3 4
	4

5 2
0 2 2 5
1 2 1 3
0 2 2 4
1 2 3 5
0 2 1 4
2 2 4
2 3 4
5 3
	3

6 2
0 2 6 5
1 2 4 6
0 1 4
1 3 2 5 3
0 3 1 4 6
0 3 1 5 2
3 2 5 3
1 4
6 3
	8