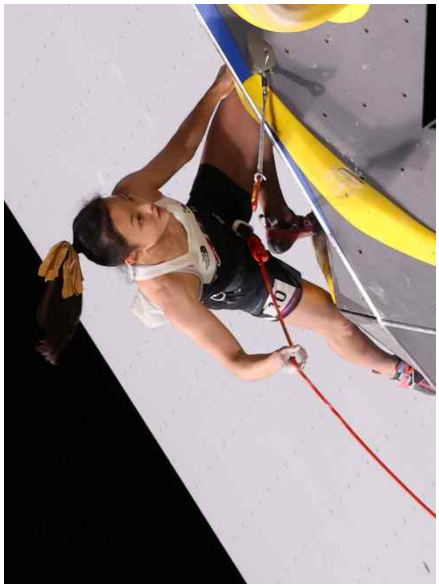
Practice Problem Set

Please check that you have 3 problems that are spanned across 7 pages in total (including Korean translation and this cover page).

A. Sport Climbing Combined	(1+1 pages)	Korean translation available
B. Ten	(2 pages)	
C. Moving Logs	(2 pages)	

Problem A
Sport Climbing Combined
Time Limit: 1 Second

Sport Climbing is an indoor rock climbing sport that started in 1986. Athletes originally competed only in **lead climbing** events, with **speed climbing** being added in 1989, and the **bouldering** discipline added a decade later in 1999. At the Olympic Games, athletes compete in the three disciplines that are combined in a single ranking to decide gold, silver and bronze medals. The ranking is determined by a multiplied score of climbers' placements in the three disciplines. For example, if a climber places first in lead, 5th in speed, second in bouldering, then the score would be 10. The climber with the lower score has precedence in the ranking.



Given the bib numbers of n climbers and their placements in the three disciplines, write a program to determine which climbers will be awarded gold, silver, and bronze medals. It may happen that two climbers have the same multiplied score. In this case, the climber with the lower added score of placements in the three disciplines wins. If two climbers have the same added score as well as the multiplied score, the climber with the lower bib number wins.

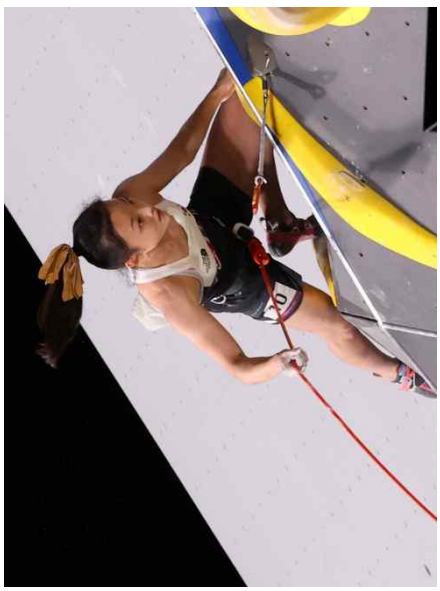
Input
Your program is to read from standard input. The first line contains a positive integer n ($3 \le n \le 100$) indicating the number of climbers. In the following n lines, each line contains four integers b_i, p_i, q_i, r_i, where b_i is the bib number of i-th climber, and p_i, q_i, r_i respectively are his/her placements in the three disciplines of lead, speed and bouldering. You may assume that the bib numbers are distinct and positive integers no more than 999; also, the placements of climbers in each discipline are integers between 1 and n. Two climbers may have the same placement in each discipline.

Output
Your program is to write to standard output. Print exactly one line that contains three integers representing the bib numbers of the climbers who will be awarded gold, silver, and bronze medals in order.

The following shows sample input and output for one test case.

Sample Input	Output for the Sample Input
4 301 4 3 2 815 2 2 1 717 1 1 4 505 3 4 2	815 717 301

Problem A
Sport Climbing Combined
제한 시간: 1 초



스포츠 클라이밍은 1986 년에 시작된 실내 암벽 등반 스포츠이다. 선수들은 원래 **리드 클라이밍** 종목에서만 겨루었는데, 1989 년에 **스피드 클라이밍**이 추가되었고, 10 년 후인 1999 년에 **볼더링** 종목이 추가되었다. 올림픽 게임에서는 금, 은, 동메달을 결정하기 위하여 선수들은 세 종목에서 겨루어 종합 순위를 매긴다. 종합 순위는 세 종목에서 거둔 순위를 곱한 점수로 결정된다. 예를 들어, 어떤 선수가 리드에서 1 위, 스피드에서 5 위, 볼더링에서 2 위를 했다면 점수는 10 점이 된다. 곱한 점수가 낮은 선수가 종합 순위에서 앞선다.

선수 n명의 등번호와 이들이 세 종목에서 거둔 순위가 주어질 때, 금, 은, 동메달을 받을 선수를 결정하는 프로그램을 작성하시오. 두 선수의 곱한 점수가 같을 수도 있다. 이 경우, 세 종목 순위의 합산 점수가 낮은 선수가 이긴다. 두 선수의 곱한 점수와 합산 점수가 모두 같으면 등번호가 낮은 선수가 이긴다.

Input
입력은 표준입력을 사용한다. 첫째 줄에 선수의 명수를 나타내는 양의 정수 n $(3 \le n \le 100)$이 주어진다. 이어 n개의 줄 각각에 네 정수 b_i, p_i, q_i, r_i가 주어지는데, b_i는 i번째 선수의 등번호이고, p_i, q_i, r_i는 각각 그 선수가 리드, 스피드, 볼더링 종목에서 거둔 순위를 나타낸다. 선수들의 등번호는 서로 다르고 999 이하인 양의 정수이다. 또한, 각 종목에서 선수의 순위는 1과 n 사이의 정수이다. 각 종목에서 두 선수의 순위가 같을 수도 있다.

Output
표준출력을 사용한다. 금, 은, 동메달을 받을 선수의 등번호를 나타내는 세 정수를 순서대로 한 줄에 출력한다.

다음은 한 테스트 경우에 대한 입출력 예이다.

Sample Input	Output for the Sample Input
4 301 4 3 2 815 2 2 1 717 1 1 4 505 3 4 2	815 717 301

Problem B
Ten
Time Limit: 0.5 Seconds

A real estate company IC is managing a rectangular section of land. The section is divided into $m\,n$segments in $m \times n$ matrix shape, where the number of rows and that of columns are m and n, respectively. Each segment has its own price as a positive integer. IC wants to sell a rectangular subsection of the land, but the price of the subsection should be ten. The price of a subsection is simply the sum of the prices of the segments in the subsection. Since there can be several such subsections, IC wants to identify the number of candidate subsections to sell. Write a program to help IC, counting the number of candidate subsections of the land.

For example, the prices of segments of the land having 5×7 segments are given as follows:

3	1	2	1	4	6	7
4	5	2	2	2	7	9
4	7	1	1	2	9	2
4	5	3	8	7	2	5
1	7	1	7	6	2	3

We can find four candidate subsections to sell marked by rectangles: the first one consists of four segments in the first and the second rows spanning over from the second to the third columns, the second, six segments in the second and the third rows spanning over from the third to the fifth columns, the third, two segments in the first row spanning over from the fifth to the sixth columns, and the fourth, three segments in the seventh column spanning over from the third to the fifth rows. Therefore, your program should report four for the above input.

Input
Your program is to read from standard input. The input starts with two positive integers m and n ($1 \leq m, n \leq 300$), denoting the dimensions of the land, which are given separated by a space. Each of the following m lines contains n positive integers p_{ij} representing the prices of the segments of the i-th row of the land ($1 \leq i \leq m$, $1 \leq j \leq n$, and $1 \leq p_{ij} \leq 10$). The prices are also separated by a space.

Output
Your program is to write to standard output. Print exactly one line containing an integer representing the number of rectangular subsections with the price of ten.

The following shows sample input and output for two test cases.

Sample Input 1	Output for the Sample Input 1
3 5 3 1 2 1 4 4 5 2 2 2 4 7 1 1 2	2

Sample Input 2 **Output for the Sample Input 2**

4 6 3 1 2 1 4 6 4 5 2 2 2 7 4 7 1 1 1 9 4 3 3 3 7 2	8

Problem C
Moving Logs
Time Limit: 1 Second

There are n logs placed in a rectangular warehouse. The logs do not intersect or overlap each other. The right wall of the warehouse is open, through which logs can be dragged out to the right. A log is only moved parallel in the positive x-axis direction. A log can only be pulled out if there are no other logs in the space through which it will move. In Figure C.1, the movement space of log 3 is grayed out. Log 3 cannot be pulled out until logs 1 and 5 have been removed.

Multiple logs can be pulled out simultaneously if there are no other logs in the space through which they will move. Suppose that it takes 1 unit of time to pull out a log. Your task is to pull out all the logs of the warehouse as quickly as possible.

In Figure C.1, in order to pull out all five logs, you have to pull out the logs one by one in the order of 1-5-3-2-4. Therefore, it takes 5 units of time to complete the task. Note that since the end point of log 1 is located in the movement space of log 5, it is not possible to pull out log 5 first.

Consider an example shown in Figure C.2. Logs 2 and 4 can be pulled out at the same time. After that, you can pull out logs 1 and 3 at the same time. Finally, you can pull out log 5. Therefore, it takes 3 units time.

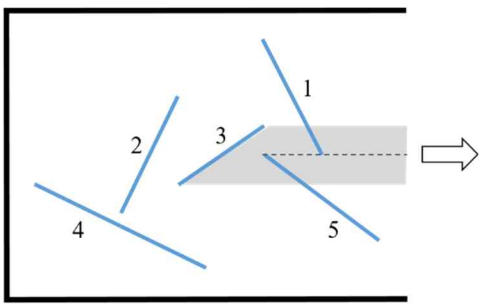


Figure C.1

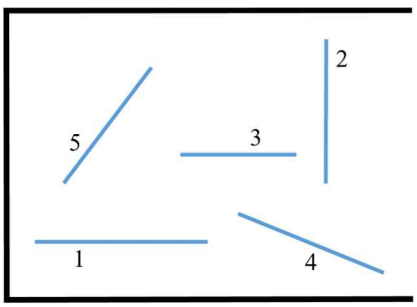


Figure C.2

Given the locations of n logs, write a program to find the minimum time required to pull out all the logs.

Input
Your program is to read from standard input. The input starts with a line containing an integer n ($1 \leq n \leq 20{,}000$), where n is the number of logs. The logs are numbered from 1 to n. In the following n lines, the i-th line contains four integers, x_1, y_1, x_2, and y_2, where (x_1, y_1) and (x_2, y_2) are the coordinates of both end points of the i-th log and all the integers are between 1 and 10^9. The length of a log is more than 0 and no two logs intersect each other at any point.

Output

Your program is to write to standard output. Print exactly one line. The line should contain an integer representing the minimum units of time to pull out all the logs.

The following shows sample input and output for two test cases.

Sample Input 1	Output for the Sample Input 1
5 9 9 11 5 4 3 6 7 6 4 9 6 1 4 7 1 13 2 9 5	5

Sample Input 2	Output for the Sample Input 2
5 1 2 7 2 11 4 11 9 10 5 6 5 8 3 13 1 2 4 5 8	3