At the Olympic Games, countries are ranked according to the number of medals their athletes won. However, there is more than one possible way of ranking countries by their medals. In Europe, for example, countries are first ranked by the number of gold medals their athletes won. Ties are broken by looking at silver medals, and then at bronze medals. In Canada, however, because Canadian athletes do not win very many gold medals, countries are ranked by the overall number of medals won, giving the same weight to gold, silver, and bronze medals.

In general, a ranking scheme can be thought of as a vector of positive weights. This vector is multiplied with the vector of medals won by each country, and the scalar product of the two vectors defines the score of the respective country, which is then used to produce the ranking. In this general scheme, the European ranking technique corresponds to the weight vector $(10^{20}, 10^{10}, 1)$, whereas the Canadian method corresponds to the vector $(1, 1, 1)$.

In this problem, you will only need to consider weight vectors of the form $(1/n^j, 1/n^k, 1/n^l)$, where n is the total number of medals won by all athletes in the Olympic Games, and j, k, and l are integers.

Input

The input contains multiple test cases. Each test case starts with an integer c, the number of countries to follow. Each of the following c lines contains the name of a country and three integers g, s, and b — the number of gold, silver, and bronze medals won by the country. The last test case has $c = 0$ and must not be processed. It is guaranteed that each test case contains at most 20 different countries and that the total number of medals smaller than 100. Country names do not contain whitespace characters.

Output

Given a list of countries and the number of gold, silver, and bronze medals won by each country, print the line

```
Canada wins!
```

if there is a weight vector of the above form such that Canada ranks first according to the ranking scheme defined by that vector. Print the line

```
Canada cannot win.
```

if no such vector exists.

Sample Input

```
2
Canada 3 2 1
USA 1 2 3
2
USA 2 2 2
Canada 1 1 1
0
```

Sample Output

```
Canada wins!
Canada cannot win.
```