To solve this exercise you will need the definitions and the procedures of problems P46254, P84786 and P39799.

Write a function that prints the relationship that have two given circles c1 and c2:

```cpp
int relationship (const Circle &c1, const Circle &c2);
```

Your function must return 1 if c1 is inside c2, 2 if c2 is inside c1, 3 if any circle in inside the other one but the circles intersect, and 0 otherwise (if the circles do not have any point in common).

Suppose that will never happen any of these extrem cases:

- The two circles intersect in a point.
- A circle is inside the other one, but shares a point with the border of the bigger circle.
- The two circles are equal.

Write a program that reads initial circles c1 and c2, followed by a series of orders, and prints which relationship have c1 and c2 in each step as it is shown in the examples.

**Input**

Input starts with two lines, one for c1, and the other one for c2, each one with three reals (the third the radius, strictly positive). Then a sequence of lines comes, each one of them starts with an integer i and an order s: i is 1 or 2, i indicates which circle must apply the order to; s is “move” or “scale”. If s is “move”, then two reals that indicate the increase of the coordinates come. If s is “scale”, then a real strictly positive that indicates scale factor comes.

**Output**

Your program must print the relationship between the two circles at the beginning and in each step, as it is shown in the instance.

<table>
<thead>
<tr>
<th>Sample input</th>
<th>Sample output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 5</td>
<td>the second circle is inside the first one</td>
</tr>
<tr>
<td>1 1 2</td>
<td>the first circle is inside the second one</td>
</tr>
<tr>
<td>2 scale 10</td>
<td>circles intersect</td>
</tr>
<tr>
<td>1 move 20 0.5</td>
<td>circles do not intersect</td>
</tr>
<tr>
<td>2 move -5 -10</td>
<td></td>
</tr>
</tbody>
</table>

**Problem information**

Author: Salvador Roura
Translator: Carlos Molina
Generation: 2023-07-14 18:04:27

https://jutge.org