Point inside a rectangle   X82451_en

Given list of pairs ⟨point,rectangle⟩, for each pair we want to know if the point is inside, at the borders, or outside the rectangle. **Complete (and respect) the following code to achieve this goal. Not respecting the code will invalidate your submission, even if it is accepted**

```cpp
#include <iostream>
#include <string>

using namespace std;

// Represents a point by its coordinates x,y.
struct Point {
  int x,y;
};

// Reads a point from the standard input and returns it.
Point read_point()
{
  Point p;
  cin>>p.x>>p.y;
  return p;
}

// Represents a rectangle by the positions its horizontal limits xmin<xmax
// and the positions of its vertical limits ymin<ymax.
struct Rectangle {
  int xmin,ymin,xmax,ymax;
};

// Reads a rectangle from the input and returns it. Assumes that the input form
Rectangle read_rectangle()
{
  Rectangle r;
  cin>>r.xmin>>r.ymin>>r.xmax>>r.ymax;
  return r;
}

// Returns "inside", "border" or "outside" depending on whether
// p is inside, at the border, or outside of r.
string containtment(Point p,Rectangle r)
{
  ...
}

int main()
{
```

```
    . . .
}
```

Exam score: 2.5 **Automatic part:** 100%

Input

The first line of the input has an integer $n \geq 1$. Each one of the next n lines has six integers $x, y, x_{min}, y_{min}, x_{max}, y_{max}$ holding $x_{min} < x_{max}$ and $y_{min} < y_{max}$.

Output

For each input line with $x, y, x_{min}, y_{min}, x_{max}, y_{max}$, write "inside", "border" or "outside" depending on whether the point represented by x, y is inside, at the border, or outside the rectangle represented by $x_{min}, y_{min}, x_{max}, y_{max}$, followed by an end of line.

Sample input

```
10
1 -2 -2 0 2 1
-3 1 -3 -4 4 2
-3 3 -5 1 -2 4
1 -3 -4 3 2 4
-5 -3 -3 -2 2 0
-3 4 -3 2 4 3
1 -2 -4 -3 4 -2
4 -4 -1 -1 3 2
-5 0 -4 -5 -2 1
-2 1 -3 -5 1 4
```

Sample output

```
outside
border
inside
outside
outside
outside
border
outside
outside
inside
```

Problem information

Author : Professorat de PRO1
Generation : 2016-07-25 19:33:00