

## Central Europe Regional Contest 2018

---

### Shooter Island

`island.c`, `island.cpp`, `Island.java`, `island.py`

You were recently promoted to captain and your troopers are now operating on a special mission in a storm. The battlefield is kind of extraordinary, it is located above the Arctic Circle on a huge ice floe. You coordinate the actions from the headquarters. There are many high-end computers keeping you up to date on the developments at the battlefield, which is modeled by the AI interface as a grid of unit squares. Each unit square is identified by its row and column index in the grid. Bigger rectangles, consisting of unit squares, are described by a pair of unit squares in the opposite corners of the rectangle. At the beginning, all squares are covered with ice.

There are two important types of information you receive from the computers:

0. Information about hits: Your enemy hit a rectangle described by unit squares  $[x_1, y_1]$  and  $[x_2, y_2]$ . This rectangle is then immediately flooded by cold arctic water.
1. A query by your troopers: They ask whether it is possible to go from square  $[x_1, y_1]$  to  $[x_2, y_2]$  by a boat. The boat may be represented by a circle of radius 0.31416. Note that the boat has to stay fully on water surface all the time and it is not allowed to leave the battlefield area.

Your troopers need your help! Can you guide them reliably?

#### Input Specification

The first input line contains an integer  $L$  ( $1 \leq L \leq 2 \cdot 10^5$ ), the number of lines to follow. Each of the next  $L$  lines contains five integers  $t, x_1, y_1, x_2, y_2$  ( $t \in \{0, 1\}$ ,  $1 \leq x_1, x_2 \leq 50$ ,  $1 \leq y_1, y_2 \leq 10^5$ ). Number  $t$  is the type of information and pairs  $[x_1, y_1]$  and  $[x_2, y_2]$  specify the respective unit squares.

#### Output Specification

For each query, output 1 if it is possible to sail from unit square  $[x_1, y_1]$  to unit square  $[x_2, y_2]$ , and 0 otherwise.

**Sample Input 1**

6  
0 4 4 6 6  
0 6 6 7 8  
0 1 3 3 3  
1 1 7 6 1  
1 5 4 6 8  
1 4 5 1 3

**Output for Sample Input 1**

0  
1  
0

**Sample Input 2**

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

**Output for Sample Input 2**

1

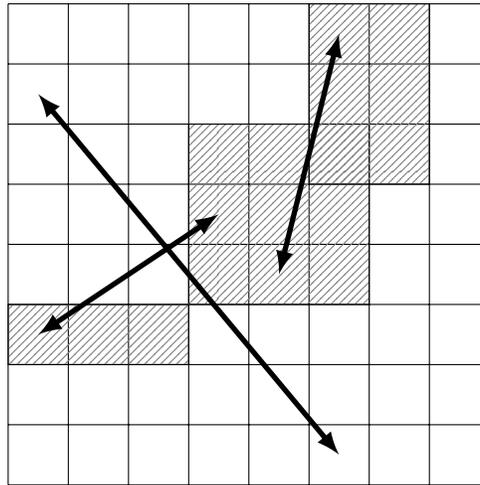


Figure 1: Illustration of Sample Input 1.