24th Annual Rowan University Programming Contest

hosted by the
Computer Science Department

Friday, 16 April 2010

Contest Problem

1 Background

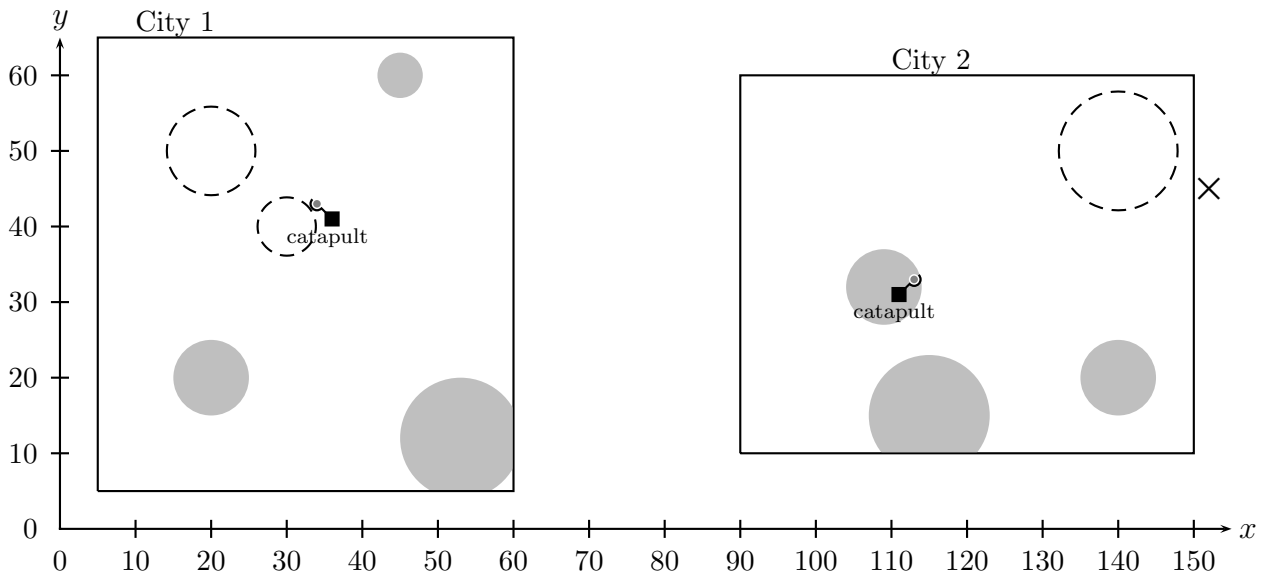
Two walled cities are at war. Each is rectangular, with an indestructible wall around it. Each also has a single catapult, which they use to launch barrels of gunpowder at each other. The barrels are measured based on their blast radius.

As part of war preparations, each side has had spies trying to find the enemy's catapult. Each side has a list of spots they will target, where suspicious buildings or tents were located, in hopes that one of those targets will be the other side's hidden catapult. Even if they do not hit the catapult, there will still be some damage to the other city's territory.

The targeted spots were carefully chosen to ensure that the damage zones will not overlap, and while damage zones may be so near to corners that they contact both walls, they will never actually contact a corner, because that would be too wasteful. However, some targets are close to the walls, and in such a case do not destroy a completely circular area. Also, sometimes the spies have sloppy handwriting, and so some barrels land outside the city entirely.

Because it takes several minutes to prepare the catapult and fire, any battle proceeds with strict alternation of volleys. So one city will fire, and then the other, and then the first, until such time as a catapult is destroyed or both cities run out of ammunition.

The diagram below illustrates an example after a battle has ended. The city walls are in black, the catapults are labeled, and areas damaged by exploding barrels of gunpowder are shaded gray. Any barrels which landed outside the walls have their locations marked with an X, but did no damage to the city. (The dashed circles represent areas of damage which would have been destroyed per each side's plans, if the battle hadn't ended.)

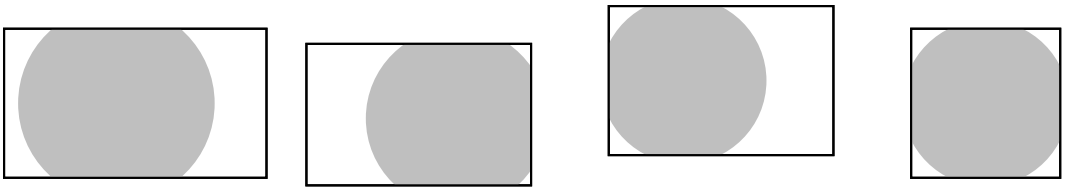


Note that one of City 1's attacks completely overshot City 2, landing outside the city and doing no damage. Unfortunately for City 2, their catapult was lost to City 1's fourth barrel. Notice that barrels which explode close to the wall do not damage a fully circular area, because the walls are indestructible. (This diagram corresponds to the values in the Sample Input on page 4.)

Your challenge is to write a program that computes damage assessments for battles, given the information about how the two cities are laid out and the battle plans drawn up by the opposing commanders.

2 Complications

The generals consider it wasteful to bomb the corners of a city, so no blast radius will ever include a corner. However, they may sometimes direct their attacks so that a blast radius contacts more than one wall.



Your program will have to correctly handle any such cases.

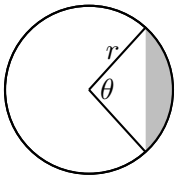
3 Math Which May Be Helpful

The distance between two points (x_1, y_1) and (x_2, y_2) is: $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$.

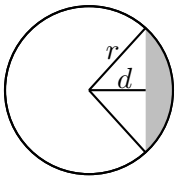
The area of a circle with radius r is: $A = \pi r^2$.
(For the purposes of this contest, you should consider π to be 3.14159265.)

The area of a circle segment (the shaded portion of the diagram at right), where the circle has radius r, and the angle occupied by the segment is θ, is: $A = \frac{1}{2}r^2(\theta - \sin(\theta))$.

θ is the angle's value in radians.



The vertex angle of an isosceles triangle (labeled θ of the triangle above) can be found, if you know the radius r and the distance d from the center of the circle to the straight edge of the segment (as diagrammed at right), using this formula: $\theta = 2 \arccos\left(\frac{d}{r}\right)$.



4 Trigonometric Functions

C/C++, Java, and Visual Basic all include functions for sine, and all spell it *sin*.
arccos is available in C/C++ as *acos()*, which you can get by including the *math.h* header.
arccos is available in Java as *Math.acos()*.
arccos is not available in all versions of Visual Basic, so it has been included here[1] (older versions of VB use *Atn* and *Sqr* instead of *Math.Atan* and *Math.Sqrt*):

```
Function ArcCos(X As Double) As Double
    ArcCos = Math.Atan(-X / Math.Sqrt(-X * X + 1)) + 2 * Math.Atan(1)
End Function
```

Note that these functions all work with angles in radians, as needed by the formula above.

[1] code taken from: <http://msdn.microsoft.com/en-us/library/w3t84e33(VS.80).aspx>

5 Input

5.1 Input Specification

For text input, your program should accept input in the following format:

1. An integer, $\mathcal{D}$, where $1 \leq \mathcal{D} \leq 50$, which is the number of datasets in this file.
2. $\mathcal{D}$ data sets, each of which is in this format:
   (a) One line with 4 integers, the (X,Y) coordinates of opposite corners of City 1.
   (b) One line with 2 integers, the (X,Y) coordinates of City 1's catapult.
   (c) One line with 4 integers, the (X,Y) coordinates of opposite corners of City 2.
   (d) One line with 2 integers, the (X,Y) coordinates of City 2's catapult.
   (e) One line with a single integer, $\mathcal{T}$, where $1 \leq \mathcal{T} \leq 50$, which is number of targets in each city's battle plan.
   (f) 1 line with $\mathcal{T}$ integers, representing the barrels in City 1's armory. These barrels will be fired in the order they are listed.
   (g) 1 line with $\mathcal{T}$ integers, representing the barrels in City 2's armory. These barrels will be fired in the order they are listed.
   (h) $\mathcal{T}$ lines with four integers, in two sets of two, representing the planned targets of each city. The first pair represents the X and Y coordinate of City 1's target for each volley. The second pair represents the X and Y coordinate of City 2's target in the same exhange.

5.2 Sample Input

Data in file	Item # above:	Meaning in plain English
1	1	*this file has 1 data set*
5 5 60 65	2a	*City 1 has corners at (5,5) and (60,65)*
36 41	2b	*City 1's catapult is at (36,41)*
90 10 150 60	2c	*City 2 has corners at (90,10) and (150,60)*
111 31	2d	*City 2's catapult is at (111,31)*
5	2e	*each city has 5 barrels*
5 3 8 5 8	2f	*the blast radii of City 1's barrels, in the order they will be fired*
5 8 3 6 4	2g	*the blast radii of City 2's barrels, in the order they will be fired*
140 20 20 20		
152 45 53 12		*the list of targets drawn up by each city's generals*
115 15 45 60	2h	*City 1's first target is (140,20); City 2's first target is (20,20)*
109 32 20 50		*City 1's next target is (152,45); City 2's next target is (53,12)*
140 50 30 40		

(This input, corresponding to the diagram on page 1, will be on the website as **sample.txt**.)

You may choose to have your program read the input from the keyboard, or ask the user for a filename and then read the file. Users of GUI-based programming environments may prefer to use text boxes into which the values can be entered, and buttons to begin their calculation. Any reasonable variation in the spirit of the problem is acceptable.

Do not hard-code the values into your program.

6 Output

6.1 Output Specification

For each data set configuration, your program must generate output as follows:

1. The text '`Analyzing D data sets`', where **D** is the number of data sets in the input.
2. The text '`Data set S:`', where **S** is the number of the data set being reported on.
3. One of the following, depending on which is true:
   (a) The text '`Neither catapult destroyed after T shots`', where **T** is the number of shots that were fired.
   (b) The text '`City C catapult destroyed after T shots`', where **T** is the number of shots that were fired by the enemy at the point where the catapult was destroyed
4. The text '`City 1 damage:  A1 square meters.`', where **A1** is the area of City 1 which was damaged by exploding barrels.
5. The text '`City 2 damage:  A2 square meters.`', where **A2** is the area of City 2 which was damaged by exploding barrels.

6.2 Sample Output

```
Analyzing 1 Data sets
 Data set 1:
  City 2 catapult destroyed after 4 shots.
  City 1 damage: 297.4 square meters.
  City 2 damage: 332.0 square meters.
```

(This output corresponds to the Sample Input from §5.2.)

Your output does **not** have to duplicate the sample output as regards spacing or use of upper/lower case. Your output should be neat, but need not exactly match the sample.

SPECIFIC NOTE ABOUT THE SAMPLE DATA:

City 2 will have fired only 3 barrels at the point where their catapult is destroyed; the output message refers to how many shots were needed by City 1 to destroy City 2's catapult.

GENERAL NOTES:

All measurements are given in meters, and all values will in the interval $[0, 10000]$.

You need not do error-checking on the input. Each line will have exactly the number of items described with no stray characters. There will be no blank lines.

While all input data are integers, you should print print the damage assessments rounded to the nearest tenth.

The catapults are assumed to have a radius of zero.

City 1 always fires first. (That's why they're City 1.)

7 Test Data

Run your program on this input and print the results. **You must submit printed output to earn full points.** Your program will also be run on data known only to the judges.

7.1 Test Input #1

```
3
90 10 150 60
111 31
5 5 60 65
36 41
5
5 8 3 6 4
5 3 8 5 8
20 20 140 20
53 12 152 45
45 60 115 15
20 50 109 32
30 40 140 50
5 5 60 65
36 41
90 10 150 60
111 31
1
5
5
140 20 20 20
0 0 50 50
25 25
50 50 100 100
75 75
15
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
51 51 1 1
53 51 3 1
55 51 5 1
57 51 7 1
59 51 9 1
61 51 11 1
63 51 13 1
65 51 15 1
67 51 17 1
69 51 19 1
71 51 21 1
73 51 23 1
75 51 25 1
77 51 27 1
79 51 29 1
```

7.2 Test Input #2

```
3
10 10  20 20
11 11
10 30  20 40
15 35
9
1 2 1 2 1 2 1 2 1
7 1 2 1 2 1 2 1 2
10 45  15 15
15 45  10 21
20 45  15 21
10 50  20 21
15 50  10 26
20 50  15 26
10 55  20 26
15 55  5  5
20 55  25 25
1000 1000 2000 2000
1500 1500
3000 3000 4000 4000
3500 3500
5
20 20 20 20 20
10 20 30 40 50
3100 3100 1100 1100
3200 3200 1200 1200
3300 3300 1300 1300
3400 3400 1400 1400
3500 3500 1500 1500
1 1 3 3
2 2
20 60  50 10
35 45
5
5 4 3 2 1
1 2 3 4 5
19 45 4 4
51 45 1 10
35 9  10 1
35 61 20 1
35 47 30 1
```

All sample and test data sets are available at <http://elvis.rowan.edu/rupc/2010>