26/07/2010

Topic : Simple Geometry

10387 – Billiard

Problem A: Billiard

Time Limit : 3.000 seconds

In a billiard table with horizontal side **a** inches and vertical side **b** inches, a ball is launched from the middle of the table. After **s** > 0 seconds the ball returns to the point from which it was launched, after having made **m** bounces off the vertical sides and **n** bounces off the horizontal sides of the table. Find the launching angle **A** (measured from the horizontal), which will be between 0 and 90 degrees inclusive, and the initial velocity of the ball.

Assume that the collisions with a side are elastic (no energy loss), and thus the velocity component of the ball parallel to each side remains unchanged. Also, assume the ball has a radius of zero. Remember that, unlike pool tables, billiard tables have no pockets.

Input

Input consists of a sequence of lines, each containing five nonnegative integers separated by whitespace. The five numbers are: **a**, **b**, **s**, **m**, and **n**, respectively. All numbers are positive integers not greater than 10000.

Input is terminated by a line containing five zeroes.

Output

For each input line except the last, output a line containing two real numbers (accurate to two decimal places) separated by a single space. The first number is the measure of the angle **A** in degrees and the second is the velocity of the ball measured in inches per second, according to the description above.

Sample Input

```
100 100 1 1 1
200 100 5 3 4
201 132 48 1900 156
0 0 0 0 0
```

Sample Output

```
45.00 141.42
33.69 144.22
3.09 7967.81
```

10112 – Myacm Triangles

Problem B: Myacm Triangles

Time Limit: 3.000 seconds

Source file:	`triangle.{c, cpp, java, pas}`
Input file:	`triangle.in`
Output file:	`triangle.out`

There has been considerable archeological work on the ancient Myacm culture. Many artifacts have been found in what have been called power fields: a fairly small area, less than 100 meters square where there are from four to fifteen tall monuments with crystals on top. Such an area is mapped out above. Most of the artifacts discovered have come from inside a triangular area between just three of the monuments, now called the power triangle. After considerable analysis archeologists agree how this triangle is selected from all the triangles with three monuments as vertices: it is the triangle with the largest possible area that does not contain any other monuments inside the triangle or on an edge of the triangle. Each field contains only one such triangle.

Archeological teams are continuing to find more power fields. They would like to automate the task of locating the power triangles in power fields. Write a program that takes the positions of the monuments in any number of power fields as input and determines the power triangle for each power field.

A useful formula: the area of a triangle with vertices (x_1, y_1), (x_2, y_2), and (x_3, y_3) is the absolute value of

$$0.5 \times [(y_3 - y_1)(x_2 - x_1) - (y_2 - y_1)(x_3 - x_1)].$$

For each power field there are several lines of data. The first line is the number of monuments: at least 4, and at most 15. For each monument there is a data line that starts with a one character label for the monument and is followed by the coordinates of the monument, which are nonnegative integers less than 100. The first label is A, and the next is B, and so on.

There is at least one such power field described. The end of input is indicated by a 0 for the number of monuments. The first sample data below corresponds to the diagram in the problem.

For each power field there is one line of output. It contains the three labels of the vertices of the power triangle, listed in increasing alphabetical order, with no spaces.

Example input:

```
6
A 1 0
B 4 0
C 0 3
D 1 3
E 4 4
F 0 6
4
A 0 0
B 1 0
C 99 0
D 99 99
0
```

Example output:

```
BEF
BCD
```

10250 – The Other Two Trees

Problem E

The Other Two Trees

Input: standard input **Output:** standard output

Time Limit: 2 seconds

You have a quadrilateral shaped land whose opposite fences are of equal length. You have four neighbors whose lands are exactly adjacent to your four fences, that means you have a common fence with all of them. For example if you have a fence of length **d** in one side, this fence of length **d** is also the fence of the adjacent neighbor on that side. The adjacent neighbors have no fence in common among themselves and their lands also don't intersect. The main difference between their land and your land is that their lands are all square shaped. All your neighbors have a tree at the center of their lands. Given the Cartesian coordinates of trees of two opposite neighbors, you will have to find the Cartesian coordinates of the other two trees.

Input

The input file contains several lines of input. Each line contains four floating point or integer numbers **x1, y1, x2, y2,** where **(x1, y1), (x2, y2)** are the coordinates of the trees of two opposite neighbors. Input is terminated by end of file.

Output

For each line of input produce one line of output which contains the line "**Impossible.**" without the quotes, if you cannot determine the coordinates of the other two trees. Otherwise, print four floating point numbers separated by a single space with ten digits after the decimal point **ax1, ay1, ax2, ay2,** where **(ax1, ay1)** and **(ax2, ay2)** are the coordinates of the other two trees. The output will be checked with special judge program, so don't worry about the ordering of the points or small precision errors. The sample output will make it clear.

Sample Input

```
10 0 -10 0
10 0 -10 0
10 0 -10 0
```

Sample Output

```
0.0000000000 10.0000000000 0.0000000000 -10.0000000000
0.0000000000 10.0000000000 -0.0000000000 -10.0000000000
0.0000000000 -10.0000000000 0.0000000000 10.0000000000
```

---

579 – Clock Hands

ClockHands

The medieval interest in mechanical contrivances is well illustrated by the development of the mechanical clock, the oldest of which is driven by weights and controlled by a verge, an oscillating arm engaging with a gear wheel. It dates back to 1386.

Clocks driven by springs had appeared by the mid-15th century, making it possible to con- struct more compact mechanisms and preparing the way for the portable clock.

English spring-driven pendulum clocks were first commonly kept on a small wall bracket and later on a shelf. Many bracket clocks contained a drawer to hold the winding key. The earliest bracket clocks, made for a period after 1660, were of architectural design, with pillars at the sides and a pediment on top.

In 17th- and 18th-century France, the table clock became an object of monumental design, the best examples of which are minor works of sculpture.

The longcase clocks (also called grandfather clocks) are tall pendulum clock enclosed in a wooden case that stands upon the floor and is typically from 6 to 7.5 feet (1.8 to 2.3 m) in height. Later, the name ``grandfather clock" became popular after the popular song "My Grandfather's Clock," written in 1876 by Henry Clay Work.

One of the first atomic clocks was an ammonia-controlled clock. It was built in 1949 at the National Bureau of Standards, Washington, D.C.; in this clock the frequency did not vary by more than one part in 10^8

Nuclear clocks are built using two clocks. The aggregate of atoms that emit the gamma radiation of precise frequency may be called the emitter clock; the group of atoms that absorb this radiation is the absorber clock. One pair of these nuclear clocks can detect energy changes of one part in 10^{14} , being about 1,000 times more sensitive than the best atomic clock.

The cesium clock is the most accurate type of clock yet developed. This device makes use of transitions between the spin states of the cesium nucleus and produces a frequency which is so regular that it has been adopted for establishing the time standard.

The history of clocks is fascinating, but unrelated to this problem. In this problem, you are asked to find the angle between the minute hand and the hour hand on a regular analog clock. Assume that the second hand, if there were one, would be pointing straight up at the 12. Give all angles as the smallest positive angles. For example 9:00 is 90 degrees; not -90 or 270 degrees.

Input

The input is a list of times in the form $H:M$, each on their own line, with 1<= H <= 12 and 0<= M <= 59. The input is terminated with the time `0:00`. Note that H may be represented with 1 or 2 digits (for 1-9 or 10-12, respectively); M is always represented with 2 digits (The input times are what you typically see on a digital clock).

Output

The output displays the smallest positive angle in degrees between the hands for each time. The answer should between `0` degrees and `180` degrees for all input times. Display each angle on a line by itself in the same order as the input. The output should be rounded to the nearest 1/1000, i.e., three places after the decimal point should be printed.

Sample Input

```
12:00
9:00
8:10
0:00
```

Sample Output

```
0.000
90.000
175.000
```