Area of given shape

locked

Problem | Submissions | Leaderboard | Discussions

Using structure definitions given in code stub, create a function with following prototype to calculate area of given shape. If two points of a rectangle are on the same line, following function should return 0. Assume value of pi to be 3.14 (please do not try to be more or less precise) and area should be correct up to two decimal places.

```
float calculate_area(sh);
```

Assume that variable of type struct circle would always have value “circle” stored as its name. And variable of type struct rectangle would always have value “rectangle” stored as its name.

Based on given input initialize appropriate member of union shape variable and pass that variable to calculate_area function and print output of received from calculate_area function.

Input Format

Input will contain n + 1 lines.

First line will have value n (number of inputs).

Next n lines will contain values for shape rectangle or circle.

Constraints

0 < n <= 1000

0 < x, y, radius <= 1000

name would always contain “circle” or “radius”

Output Format

Output should have n + 1 lines.

First line will have value n (number of inputs).

Next n lines will contain values for shape rectangle or circle.

Please print exactly two decimal places for area in output file(even when it is zero)

Sample Input 0

```
3
rectangle {{1,1},{10,10}}
circle {{0,0},10}
rectangle {{0,0},{0,5}}
```

Sample Output 0

```
3
81.00
314.00
0.00
```

Sample Input 1

```
4
rectangle {{-1,-4},{-10,-40}}
circle {{-10,-10},10}
circle {{-10,-10},0}
rectangle {{5,1},{4,1}}
```

Sample Output 1

```
4
324.00
314.00
0.00
0.00
```

Submissions: 116
Max Score: 10
Difficulty: Medium

Rate This Challenge:

Download problem statement

Download all test cases

Suggest Edits

Collapse

Admin Options

Edit Challenge

View Submissions

C

```c
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

typedef struct point
{
    int x, y;
}pt;

typedef struct rectangle
{
    char name[20];
    pt pt1;
    pt pt2;
}rect;
```

```
17
18 typedef struct circle
19 {
20     char name[20];
21     pt centre;
22     int radius;
23 }cl;
24
25 typedef union shape
26 {
27     rect r;
28     cl c;
29 }sh;
30
31 int main() {
32
33     /* Enter your code here. Read input from STDIN. Print output to STDOUT */
34     return 0;
35 }
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code