CS69001 Computing Laboratory – I

Assignment No: A1

Date: 30–July–2018

A circle C in the two-dimensional plane is specified by its center $Z = (c,d)$ and radius r, where c,d,r are real numbers, and $r \geqslant 0$. Let C_1, C_2, C_3 be three circles with centers $Z_i = (c_i, d_i)$ and radiuses r_i for $i = 1,2,3$. The goal of this assignment is to compute the area common to the three circles. Solve the following parts to achieve this goal. For simplicity assume that no two centers have the same x- or y-coordinate. If $c_1 < c_2$, we say that C_1 is to the left of C_2, and C_2 is to the right of C_1.

Part 1: Let $P = (x,y)$ be a point on the circle. Its angle θ can be determined by the built-in math library call **`atan2(y-d,x-c)`** which returns a floating-point value (**`double`**) in radian in the range $[-\pi, +\pi]$. Fig 1 illustrates the conventions.

Define a data type **`point`** to represent a point (a pair of floating-point coordinates x and y), and a data type **`circle`** to represent a circle (a point (c,d) for the center, and a floating-point radius r). Write a function **`getangle(C,P)`** to get the angle of a point P on the circle C.

Part 2: Let $C_1 = ((c_1,d_1),r_1)$ and $C_2 = ((c_2,d_2),r_2)$ be two circles. Assume that $r_1 \geqslant r_2$ (that is, C_1 is the bigger circle)—if not, exchange the roles of the two circles. Write a function **`relpos(C1,C2)`** to obtain the relative position of the circles. The return value should be one of **`DISJOINT`**, **`INSIDE`**, and **`OVERLAP`**. These three cases are illustrated in Fig 2. Let $d = \sqrt{(c_1-c_2)^2 + (d_1-d_2)^2}$ be the distance between the two centers. Then, C_1, C_2 are disjoint if $d \geqslant r_1 + r_2$, C_2 is completely inside C_1 if $d \leqslant r_1 - r_2$, and C_1, C_2 are overlapping if $r_1 - r_2 < d < r_1 + r_2$.

The conditions $d = r_1 + r_2$ and $d = r_1 - r_2$ stand for the cases when the two circles touch (outside and inside, respectively). You do not have to handle these two cases separately.

These calculations are done by floating-point arithmetic, and may introduce floating-point approximations leading to erroneous answers. If it is known that the center coordinates and the radiuses are integers, you can compare d^2 with $(r_1+r_2)^2$ and $(r_1-r_2)^2$ to ensure exact computations. In order to compute a square s^2, use **`s*s`** (instead of **`pow(s,2)`** which uses floating-point arithmetic).

Part 3: Let $C = ((c,d),r)$ be a circle, and let $P_1 = (x_1,y_1)$ and $P_2 = (x_2,y_2)$ be points on C having angles $\theta_1 = \tan^{-1}((y_1-d)/(x_1-c))$ and $\theta_2 = \tan^{-1}((y_2-d)/(x_2-c))$ (see Part 1). Consider the arc of the circle from P_1 to P_2 in the *counterclockwise* direction. A segment of the circle is defined by this arc and the straight line segment P_1P_2. Write a function **`segmentarea(C,P1,P2)`** to compute and return the area of the (counterclockwise) segment of C from P_1 to P_2. See Fig 3 for examples.

Let $\theta = \theta_2 - \theta_1$. If $\theta < 0$, add 2π to it. Any sector of C extending an angle of θ has area $A_1 = \frac{1}{2}\theta r^2$. Let the area of the triangle ZP_1P_2 be A_2 (where $Z = (c,d)$ is the center of C). If $\theta \leqslant \pi$, then the segment has area $A_1 - A_2$, whereas if $\theta \geqslant \pi$, then the segment has area $A_1 + A_2$.

Part 4: Let C_1, C_2 be overlapping circles. Write a function **`findintpts(C1,C2,...)`** to compute the two points P_1, P_2 where C_1 and C_2 intersect. These points can be computed as follows. Let the circles have the following equations.

$$\begin{aligned} C_1: \ (x-c_1)^2 + (y-d_1)^2 &= r_1^2, \\ C_2: \ (x-c_2)^2 + (y-d_2)^2 &= r_2^2. \end{aligned}$$

At the points of intersection, we have

$$(x-c_1)^2 + (y-d_1)^2 - r_1^2 = (x-c_2)^2 + (y-d_2)^2 - r_2^2.$$

Simplifying gives

$$2x(c_1-c_2) + 2y(d_1-d_2) = (c_1^2 - c_2^2) + (d_1^2 - d_2^2) + (r_2^2 - r_1^2).$$

We assume that $c_1 \neq c_2$ and $d_1 \neq d_2$. Therefore

$$y = \left(\frac{c_2 - c_1}{d_1 - d_2}\right) x + \left(\frac{(c_1^2 - c_2^2) + (d_1^2 - d_2^2) + (r_2^2 - r_1^2)}{2(d_1 - d_2)}\right).$$

Plugging this expression for y in the equation of C_1 or C_2 gives us a quadratic equation in x. Let x_1, x_2 be the two roots of this equation. Correspondingly, we get two unique values y_1, y_2 for y from the linear equation. We have $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$.

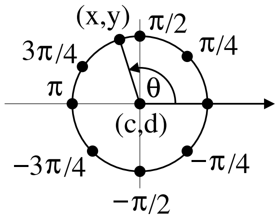
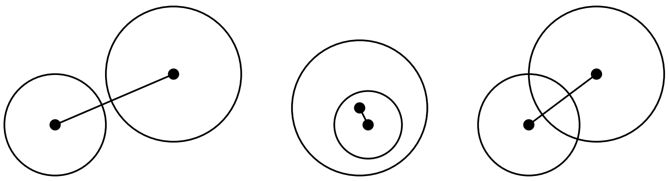
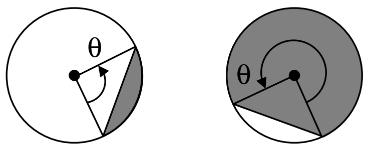


Fig 1: θ = atan2(y−d,x−c)



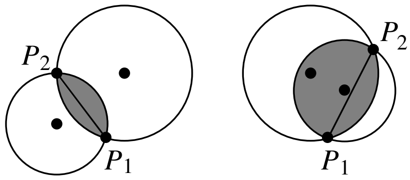
(a) Disjoint (b) Inside (c) Overlapping

Fig 2: Relative position of two circles



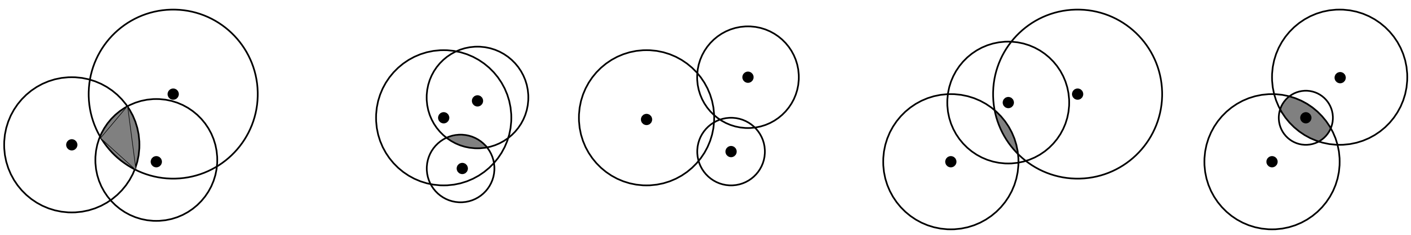
(a) θ < π (b) θ > π

Fig 3: Area of segment



(a) Small arcs (b) Large arc

Fig 4: Intersection of two circles



(i) (ii) (i) (ii)

(a) Partial overlap (b) Disjoint (c) Inside

Fig 5: Intersection of three mutually overlapping circles
Cases correspond to relative position of intersection arcs

Part 5: Write a function `intarea2(C1,C2)` to compute and return the area A common to the two circles $C_1 = (Z_1, r_1)$ and $C_2 = (Z_2, r_2)$ with the centers $Z_1 = (c_1, d_1)$ and $Z_2 = (c_2, d_2)$. Consider the three cases. If C_1, C_2 are disjoint, then $A = 0$. If one of C_1, C_2 is completely inside the other, then A is the area of the smaller circle.

The case when C_1 and C_2 overlap partially is the complicated one, and is demonstrated in Fig 4. Compute the two intersection points P_1, P_2 of the two circles in this case. Suppose that C_1 is the left circle and C_2 the right circle (that is, $c_1 < c_2$), and that P_1 has smaller y-coordinate than P_2. Then, A is the sum of the area of the segment of C_1 from P_1 to P_2 and the area of the segment of C_2 from P_2 to P_1.

Part 6: This is the final goal of this assignment: write a function **`intarea3(C1,C2,C3)`** to compute and return the area A common to three circles $C_i = (Z_i, r_i)$ with $Z_i = (c_i, d_i)$ for $i = 1,2,3$. Once again, you need to consider several cases. First, if any two of the given circles are disjoint, then $A = 0$. Second, if one of the circles is completely inside a second, then A is the area common to the contained and the remaining circles.

Finally, consider the complicated case that any two of the three circles overlap partially with one another. This case has several sub-cases as illustrated in Fig 5. Suppose that C_1, C_2, C_3 are sorted with respect to their x-coordinates. In particular, C_1 is the circle with the smallest x-coordinate.

Let the intersection points of C_1, C_2 be Q_1, Q_2 with $y(Q_1) < y(Q_2)$. Denote by α_2 the counterclockwise arc of C_1 from Q_1 to Q_2. Likewise, let C_1, C_3 intersect at points Q_3, Q_4 with $y(Q_3) < y(Q_4)$. Also, let α_3 denote the arc of C_1 from Q_3 to Q_4. The different cases of Fig 5 correspond to the relative positions of the arcs α_2 and α_3. We also need the points of intersection Q_5, Q_6 of the circles C_2, C_3.

The following three cases can be recognized by the angles θ_i of Q_i for $i = 1,2,3,4$ in C_1. Ensure three conditions: (i) $\theta_1 < \theta_2$, (ii) $\theta_3 < \theta_4$, and (iii) the four angles $\theta_1, \theta_2, \theta_3, \theta_4$ lie in a band of width 2π. In case of violations of these conditions, add 2π to appropriate angles. For example, suppose that **`atan2`** returns $\theta_1 = 5\pi/6$, $\theta_2 = \pi/3$, $\theta_3 = \pi/6$, and $\theta_4 = 2\pi/3$. This is Case 1 (partial overlap). But since $\theta_1 > \theta_2$, we change $\theta_2 = 2\pi + \pi/3 = 7\pi/3$. But $\theta_3 < \theta_4$, so we do not change θ_4. But then, the arcs $\alpha_2 = [5\pi/6, 7\pi/3]$ and $\alpha_3 = [\pi/6, 2\pi/3]$ appear disjoint (Case 2). This problem happened because $\theta_2 - \theta_3 = 7\pi/3 - \pi/6 = 13\pi/6 > 2\pi$. So change $\theta_3 = \pi/6 + 2\pi = 13\pi/6$ and $\theta_4 = 2\pi/3 + 2\pi = 8\pi/3$.

Case 1: α_2 and α_3 overlap partially (Part (a) of Fig 5). This means that the intersection points with C_1 appear in one of the two sequences Q_3, Q_1, Q_4, Q_2 or Q_1, Q_3, Q_2, Q_4 in the increasing order of their angles relative to the center of C_1. Take P_1, P_2 to be the middle two points in this sequence, that is, $(P_1, P_2) = (Q_1, Q_4)$ or $(P_1, P_2) = (Q_3, Q_2)$. Look at the points Q_5, Q_6 defined above. One of these points is inside C_1, the other is outside C_1. Call P_3 the point which is inside C_1. The intersection area A of C_1, C_2, C_3 is that of the region $P_1P_2P_3$ bounded by the three arcs P_1P_2, P_2P_3, and P_3P_1 of the three circles, and can be computed by adding the three segments areas and the area of the triangle $P_1P_2P_3$.

Case 2: α_2 and α_3 are disjoint (Part (b) of Fig 5). In this case, the four intersection points with C_1 form one of the sequences Q_1, Q_2, Q_3, Q_4 or Q_3, Q_4, Q_1, Q_2 (sorted by their angles with the center of C_1). If both Q_5 and Q_6 are inside C_1, the area A is the area of intersection of C_2 and C_3 (see Part (b)(i)). If both Q_5 and Q_6 are outside C_1 (see Part (b)(ii)), then $A = 0$.

Case 3: One of α_2, α_3 is completely contained in the other (Part (c) of Fig 5). Here, the sequence of intersection points is either Q_3, Q_1, Q_2, Q_4 or Q_1, Q_3, Q_4, Q_2. As in Case 1, take P_1, P_2 to be the middle two points of intersection in this sequence. We again consider the points Q_5, Q_6. If both are outside C_1 (see Fig 5(c)(i)), the desired area A is the intersection area of C_1 and the circle (C_2 or C_3) having the smaller arc with C_1. Finally, if both Q_5 and Q_6 are inside C_1, we have the most complicated situation as illustrated in Fig 5(c)(ii). If $y(Q_5) > y(Q_6)$, take $(P_3, P_4) = (Q_5, Q_6)$, otherwise take $(P_3, P_4) = (Q_6, Q_5)$. Now, A is the area of the region $P_1P_2P_3P_4$ enclosed by four arcs (two of which belong to circle with the contained arc). So we have to sum the four segment areas and the area of the quadrilateral $P_1P_2P_3P_4$ (the sum of the areas of the two triangles $P_1P_2P_3$ and $P_1P_3P_4$).

The *main*() function

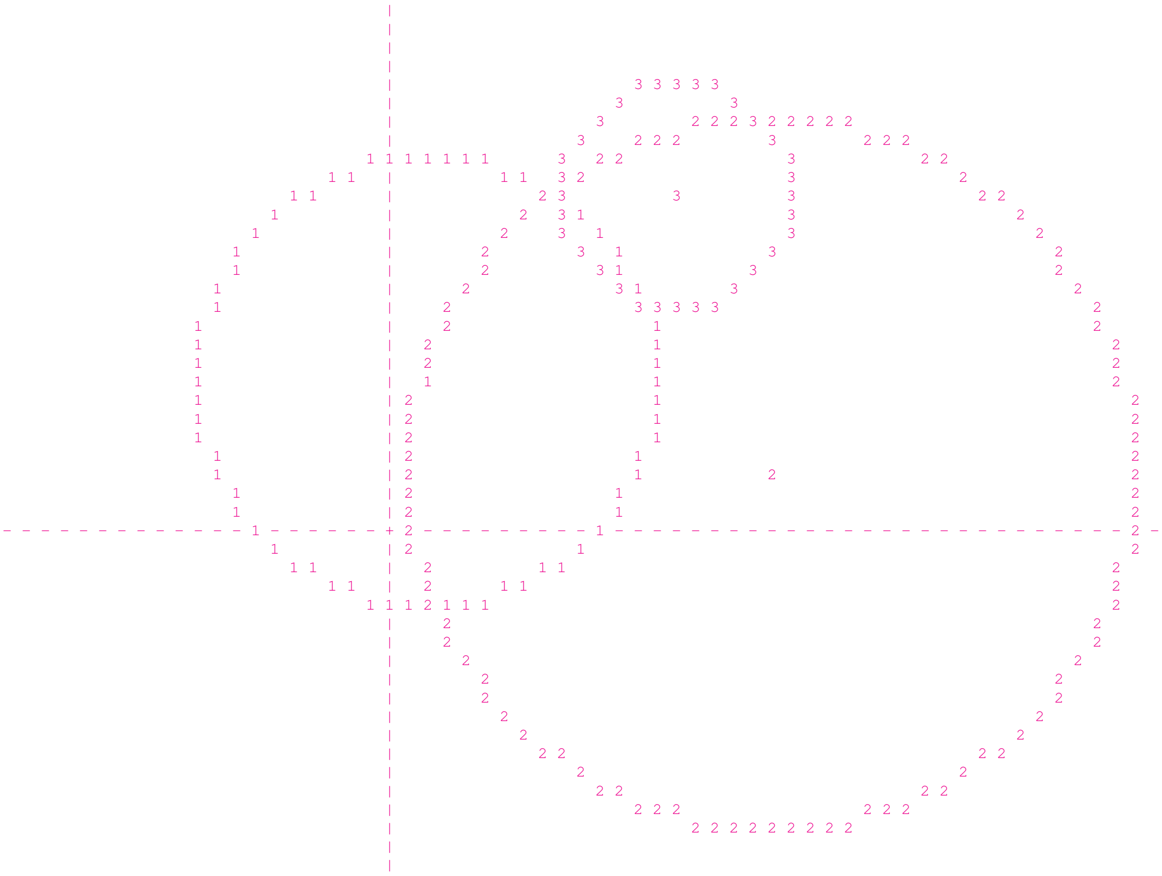
- Read c_i, d_i, r_i from the user. Assume that the user enters integer values for all of these nine quantities. Assume also that the user ensures that no two x- or y-coordinates are the same.
- Find the relative positions of the three pairs (C_1, C_2), (C_1, C_3), and (C_2, C_3) by calling **`relpos`** thrice.
- For each of these three pairs, if overlapping, compute the two intersection points using **`findintpts`**.

- Compute and print the common intersection areas for the three pairs by calling **`intarea2`** thrice.
- Call **`intarea3`** and print the area common to C_1, C_2, C_3.

Sample output

The following output illustrates Case 5(c)(i). Visit the lab webpage for sample outputs in all possible cases.

```
+++ Input data for three circles
    C1:  2  8 12
    C2: 20  3 19
    C3: 15 18  6

+++ Relative positions of
    C1,C2: OVERLAP
    C1,C3: OVERLAP
    C2,C3: OVERLAP

+++ Intersection points
    Between C1 and C2: (  8.473393, 18.104216 ) and (  2.334630, -3.995333 )
    Between C1 and C3: ( 13.213817, 12.272038 ) and (  9.005514, 17.742832 )
    Between C2 and C3: ( 19.478528, 21.992843 ) and (  9.021472, 18.507157 )

+++ Circle areas
    area(C1) =  452.389342
    area(C2) = 1134.114948
    area(C3) =  113.097336

+++ Intersection areas of two circles
    area(C1,C2) =  202.844761
    area(C1,C3) =    7.465878
    area(C2,C3) =   90.278871

+++ Intersection area of three circles
    area(C1,C2,C3) =    7.465878
```



Submit a single C/C++ source file. Do not use global/static variables.