Circular Sector Calculator

Create a program that calculates the arc length, circumference and area of a circular sector.

Note: **Bold** words are output while non-bold words are input in the following console samples.

Console Sample 1 (valid input)

```
Circular Sector Calculator

Enter the radius: .5
Enter the central angle in degrees: 120.5

Arc Length: 1.05156
Circumference: 2.05156
Area: 0.26289
```

Console Sample 2 (invalid radius)

```
Circular Sector Calculator

Enter the radius: 0
Enter the central angle in degrees: 1

Radius must be positive!
Central angle in degrees must fall in the interval (0, 360)!
```

Console Sample 3 (invalid central angle in degrees)

```
Circular Sector Calculator

Enter the radius: 1
Enter the central angle in degrees: 0

Radius must be positive!
Central angle in degrees must fall in the interval (0, 360)!
```

Console Sample 4 (invalid central angle in degrees)

```
Circular Sector Calculator

Enter the radius: 1
Enter the central angle in degrees: 360

Radius must be positive!
Central angle in degrees must fall in the interval (0, 360)!
```

Console Sample 5 (invalid radius and central angle in degrees)

```
Circular Sector Calculator

Enter the radius: -.1
Enter the central angle in degrees: -.1
```

```
Radius must be positive!
Central angle in degrees must fall in the interval (0, 360)!
```

Specifications

- You have to define a class *CircularSector* with three private data members *radius*, *central_angle_in_degrees* and *pi*, out of which *pi* is initialized with 3.14159 as part of the class definition, *radius* must be positive and *central_angle_in_degrees* must fall within the open interval (0, 360). Both *radius* and *central_angle_in_degrees* accept decimal entries.
- Declare and define a non-default constructor for the class *CircularSector* which requires parameters corresponding to all the data members except for *pi*.
- Declare and define the necessary getter and setter for the private data members *radius* and *central_angle_in_degrees* respectively in the class *CircularSector*.
- Declare and define a public member function *get_arc_length()* in the class *CircularSector* to calculate the arc length and return the result. The formula for this calculation is:

  ```
  arc_length = pi * radius * central_angle_in_degrees / 180.0
  ```

- Declare and define a public member function *get_circumference()* in the class *CircularSector* to calculate the circumference and return the result. The formulae for this calculation are:

  ```
  circumference = 2 * radius + arc_length
  ```

- Declare and define a public member function *get_area()* in the class *CircularSector* to calculate the area and return the result. The formula for this calculation is:

  ```
  area = radius * arc_length / 2.0
  ```

- Declare and define a public member function *display_results()* in the class *CircularSector* to display all the calculation results as shown in Console Sample 1 above. In this function, you are supposed to call the other member functions *get_arc_length()*, *get_circumference()* and *get_area()* directly to get the arc length, circumference and area respectively.
- In the *main()*, after displaying the title and getting all the input values from the user, you must first check the validity of those input values as requested. If all the input values are valid, the object will be created and the member function *display_results()* will be called on the object. Otherwise, the application will quit after displaying the error messages as shown in Console Sample 2, 3, 4 or 5.
- No requirement on the precision of the results.