

```
31     }  
32  
33     bubble(a,n);  
34  
35     printf("\n\n Finally sorted array is: ");  
36     for( i=0;i<=n-1;i++)  
37         printf("%d ",a[i]);  
38 } //end program.
```

## Tutorial 2

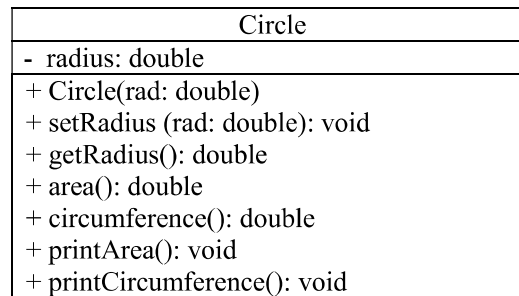
### Classes & Objects

1. Write a class `Circle` that has the following instance variables and methods:

```
public class Circle
{
    private double radius;          // radius of circle
    private static final double PI = 3.14159;

    // constructor
    public Circle(double rad) {...}
    // mutator method - set radius
    public void setRadius(double rad){...}
    // accessor method - get radius
    public double getRadius(){...}
    // calculate area
    public double area(){...}
    // calculate circumference
    public double circumference() {...}
    // print area
    public void printArea(){...}
    // print circumference
    public void printCircumference(){...}
}
```

The UML class diagram for the `Circle` class is given below:



Write an application class `CircleApp` to test the `Circle` class. The class `CircleApp` should display a menu. The user can then select an option of the following: (1) create a new circle; (2) print area; (3) print circumference; and (4) quit. Implement the operations for each option.

A sample program run is given below:

```
----jGRASP exec: java CircleApp

==== Circle Computation ====
|1. Create a new circle      |
|2. Print Area               |
|3. Print circumference      |
|4. Quit                    |
=====
Choose option (1-3):
1
Enter the radius to compute the area and circumference
```