

Circle Class

ASSIGNMENT WITHIN CLASS – CECS-2222 | SECCIÓN 22

COLON_67689

Reunión 1:

Correr el programa número 8 del capítulo 13 del texto. Adjunto el código en base como lo seguí en la clase, luego de ver la grabación.

Circle Class – Meeting Code

```
// Makeing the 8th programing challenge in class from the chapter 13 of the textbook.
// Here I'm following the instructions of the code written by the profesor in class.
// I tried to do it on my on but is a bit deferent.
#include <iostream>
using namespace std;

class Circle
{
private:
    double radius;
public:
    Circle();
    Circle(double aRad);
    void setRadius(double aRad);
    double getRadius() const;
    double getArea() const;
    double getDiameter() const;
    double getCircumference() const;
    void display() const;
};

Circle::Circle()
{
    setRadius(1.0);
}
Circle::Circle(double aRad)
{
    setRadius(aRad);
}

// Mutator function
void Circle::setRadius(double aRad)
{
    radius = aRad;
}

double Circle::getRadius() const
{
    return radius;
}
```

```

// Member funtion
double Circle::getArea() const
{
    const double PI = 3.14159;
    return PI * getRadius() * getRadius(); // I did not do it like this in my
first try.
}

double Circle::getDiameter() const
{
    return getRadius() * 2;
}

double Circle::getCircumference() const
{
    const double PI = 3.14159;
    return 2 * PI * getRadius();
}

void Circle::display() const
{
    cout << "Here is the data report for the Circular object:\n";
    cout << "Area " << getArea() << endl;
    cout << "Diameter " << getDiameter() << endl;
    cout << "Circunference " << getCircumference() << endl;
}

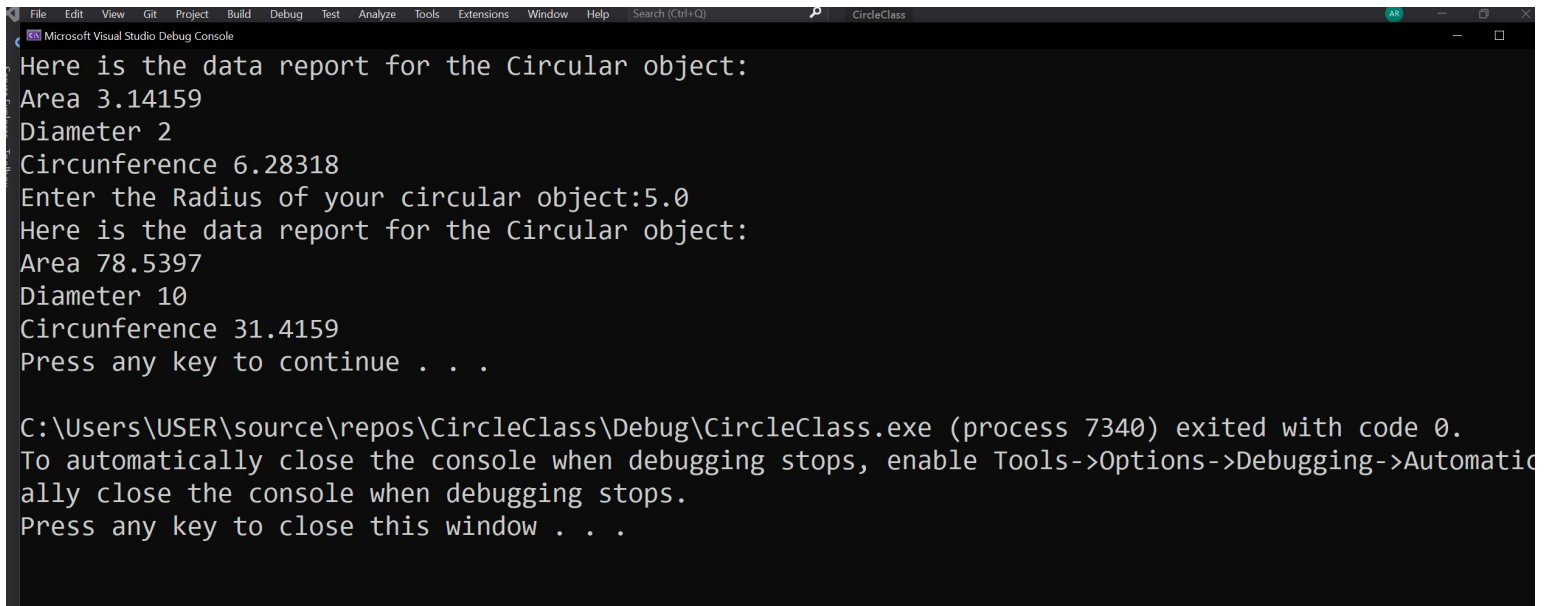
int main()
{
    Circle circular;      // Defining the obgect with the Circle class
    double aRad;
    circular.display();

    cout << "Enter the Radius of your circular object:";
    cin >> aRad;
    circular.setRadius(aRad);

    circular.display();
    system("pause");
    return 0;
}

```

Imagen de la salida:



```
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) CircleClass
Microsoft Visual Studio Debug Console
Here is the data report for the Circular object:
Area 3.14159
Diameter 2
Circunference 6.28318
Enter the Radius of your circular object:5.0
Here is the data report for the Circular object:
Area 78.5397
Diameter 10
Circunference 31.4159
Press any key to continue . . .

C:\Users\USER\source\repos\CircleClass\Debug\CircleClass.exe (process 7340) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatic
ally close the console when debugging stops.
Press any key to close this window . . .
```

Código adicional

Este código lo trabajé antes de ver la clase grabada. Mi intención era repasar un poco de lo último aprendido en el curso anterior y así tratar de resolver el ejercicio lo mas que pudiera:

Documento – circleClass.h:

```
// Declaring the Class for the circle
#include <iostream>

using namespace std;

class Circle {
private:
    double radius;
    double pi = 3.14159;

public:
    //Circle::Circle() // This is the default constructor but to my understanding of it,
    this has no parameters // therefore it cannot set the radius to 0.0; I think.
    //{
    //Circle::Circle(double rad); // This is the constructor that accepts the radius of
    the circle as an argument.
    void setRadius(double rad);
    double getRadius() const;
    double getArea() const;
    double getDiameter() const;
    double getCircumference() const;
};

// Circle::Circle(double rad) Here the constructor accepts the argument for the radius.
//{
//    radius = 0.0;
//}

void Circle::setRadius(double rad)
{
    //if (rad >= 0)
    radius = rad;
    //else
    //{
    //    cout << "Invalid Radius\n";
    //    exit(EXIT_FAILURE);
    //}
}

double Circle::getRadius() const
{
    return radius;
}
```

```

}

double Circle::getArea() const
{
    return pi * radius * radius;
}

double Circle::getDiameter() const
{
    return radius * 2;
}

double Circle::getCircumference() const
{
    return 2 * pi * radius;
}

// Deje lo inicialmente que tenía como los constructores en forma de comentario porque al
// quise intentar hacerlo todo antes de ver la clase porque no recordaba bien, y quería
// hacerlo por
// mi mismo. No fue hasta que vi la clase que aunque había entendido sobre los
// constructores es que supe
// como implementarlos. Me corrió bien al final pero creo que debí ver la clase primero
// aún después de
// repasar el material y volver a evaluar los programas anteriores.
// Lo voy a dejar así, pero escribiré tal como se hizo en la clase para práctica.

```

Documento CircleClass.cpp

```

// Exercise 8 from chapter 13 - Class Circle program.
#include <iostream>
#include "circleClass.h"
using namespace std;

int main()
{
    Circle circular;          // The object or instance of the circle class.
    double cirRad;            // The local variable for the radius.

    // Entering the radius of a circular object.
    cout << "Let's learn the Area, Diameter and Circunference of a circular object given
the radius\n";
    cout << "Enter the radius: ";
    cin >> cirRad;

    // Store the Radius of cirRad on the circular object.
    circular.setRadius(cirRad);

    // Displaying the data of the circular object.
    cout << "Here is the data report for the Circular object:\n";
    cout << "Area " << circular.getArea() << endl;
    cout << "Diameter " << circular.getDiameter() << endl;
    cout << "Circunference " << circular.getCircumference() << endl;
}

```

```

    return 0;
}

```

Imagen de la Corrida

The image shows a C++ IDE with two source files: `CircleClass.cpp` and `CircleClass.h`. The `CircleClass.cpp` file contains the `main` function and the implementation of the `Circle` class methods. The `CircleClass.h` file contains the class declaration and the definition of the `Circle` class.

CircleClass.cpp

```

1 // Exercise 8 from chapter 13 - Class Circle program.
2 #include <iostream>
3 #include "CircleClass.h"
4 using namespace std;
5
6 int main()
7 {
8     Circle circular;    // The object or instance of the circle class.
9     double cirRad;      // The local variable for the radius.
10
11     // Entering the radius of a circular object.
12     cout << "Let's learn the Area, Diameter and Circumference of a circular object given the radius\n";
13     cout << "Enter the radius: ";
14     cin >> cirRad;
15
16     // Store the Radius of cirRad on the circular object.
17     circular.setRadius(cirRad);
18
19     // Displaying the data of the circular object.
20     cout << "Here is the data report for the Circular object:\n";
21     cout << "Area " << circular.getArea() << endl;
22     cout << "Diameter " << circular.getDiameter() << endl;
23     cout << "Circumference " << circular.getCircumference() << endl;
24
25     return 0;
26
27 }

```

CircleClass.h

```

1 // Declaring the Class for the circle
2 #include <iostream>
3
4 using namespace std;
5
6 class Circle {
7 private:
8     double radius;
9     double pi = 3.14159;
10
11 public:
12     //Circle::Circle() // This is the default constructor but to my understanding of it, this has no
13     // | | | | // therefore it cannot set the radius to 0.0; I think.
14     //{}
15     //Circle::Circle(double rad); // This is the constructor that accepts the radius of the circle
16     void setRadius(double rad);
17     double getRadius() const;
18     double getArea() const;
19     double getDiameter() const;
20     double getCircumference() const;
21 };
22
23 // Circle::Circle(double rad) Here the constructor accepts the argument for the radius.
24 //{}
25 // radius = 0.0;
26 //{}
27
28 void Circle::setRadius(double rad)
29 {
30     //if (rad >= 0)
31     | radius = rad;
32     //else
33     //{}
34     // cout << "Invalid Radius\n";
35     // exit(EXIT_FAILURE);
36     //{}
37 }
38
39
40 double Circle::getRadius() const
41 {
42     return radius;
43 }

```

Terminal Output

```

cd "/Users/albertocolon/Documents/Politecnica Local/Trimestre 2021-2022/3-SP - 22/CECS/CECS-2222_22/CECS_2222/Modulo-1/" && g++ CircleClass.cpp -o Circle
albertocolon@Albertos-Mac-mini: CECS_2222 % cd "/Users/albertocolon/Documents/Politecnica Local/Trimestre 2021-2022/3-SP - 22/CECS/CECS-2222_22/CECS_2222/Modulo-1/" && g++ CircleClass.cpp -o Circle &&
./CircleClass.h:19:19: warning: default member initializer for non-static data member is a C++11 extension [-Wc++11-extensions]
double pi = 3.14159;
                  ^
1 warning generated.
Let's learn the Area, Diameter and Circumference of a circular object given the radius
Enter the radius: 5.5
Here is the data report for the Circular object:
Area 95.8331
Diameter 11
Circumference 34.5575
albertocolon@Albertos-Mac-mini: Modulo-1 %

```