ASSIGNMENT: 1 Advanced Programing Jaswanth Sai K

Subject Code: 19CSE201 Roll No:CH.EN.U4CSE20130

Question:

Create a Circle class with a centre (x,y) and radius r. Use parameterized constructor to initialize the radius and centre of a circle object. Write a copy constructor that copies the values of one circle object to another one. Use a member function to display the values x,y,r of a circle object. Allow a friend function to update the centre and radius of the circle. There should a static variable ‘number\_of\_circles’ and a static member function ‘update\_number\_of\_circles’ which will update the ‘number\_of\_circles’ whenever an instance of a circle is created.

Solution:

Aim:

To Create a C++ program For above problem and check whether it is working or not.

Algorithm:

Step 1: Create a class with name Circle

Step 2: Create member variables which are private by default.

Step 3: Create a default constructor to count the number of objects created.

Step 4: Create a parameterized constructor which will have the same name as class which takes arguments and updates the value of the member variables

Step 5: Create a copy constructor which takes object of the class as an argument.

Step 6: Create a void display function which is a member function that prints the values of Circle object.

Step 7: Declare a friend function within the class which will be able to access the private member variables.

Step 8: Update the number\_of\_circles value in the constructor which will calculate the number of objects created.

Step 9: Update the value of member variables using the friend function.

Step 10: Create objects in the main function.

Step 11: Call the display function to print the values

Step 12: Call the friend function to update the value.

Step 13: Invoke the static members using the class name and the scope resolution operator.

Code:

#include<iostream>

using namespace std;

class circle{

private:

//private data members.

float x\_corr,y\_corr;

float radius;

public:

static int number\_of\_circles ; // static data member.

circle(){

update\_number\_of\_circle();

}

circle(int a,int b,float rad) // Declaring parameterized Constructor.

{

x\_corr = a;

y\_corr = b;

radius = rad;

}

circle(circle &copy\_circle) // Declaring Copy Constructor.

{

x\_corr = copy\_circle.x\_corr;

y\_corr = copy\_circle.y\_corr;

radius = copy\_circle.radius;

}

void display\_circle(string obj\_name) // Declaring member function for displaying data.

{

cout<<"Data of Object - "<<obj\_name<<" :\t";

cout<<"Center :(x,y) = ("<<x\_corr<<","<<y\_corr<<")\t";

cout<<"Radius : "<<radius<<endl;

}

friend void update\_circle(circle &,float , float , double); // defining friend function.

static void update\_number\_of\_circle() // static member function for update\_number\_of\_circle

{

number\_of\_circles += 1;

}

};

// initializing static varible

int circle::number\_of\_circles = 0;

// Declaring friend function.

void update\_circle(circle &object\_name,float x\_corrdinate,float y\_corrdinate,double radi)

{

object\_name.x\_corr = x\_corrdinate;

object\_name.y\_corr = y\_corrdinate;

object\_name.radius = radi;

}

int main()

{

circle c1,c2,c3; // Declaring objects c1,c2,c3.

c1 = circle(4,5,5.0); // Assigning values to private Data members

c2 = circle(5,9,8.0); // using parameterized constructor for c1,c2 circle objects.

c3 = circle(c1); // Copying C1 to C2 using Copy Constructor.

cout<<"\nData after initializing & CopyConstructor :-\n\n";

c1.display\_circle("c1");

c2.display\_circle("c2"); // displaying data members of class using member function.

c3.display\_circle("c3");

update\_circle(c2 ,9,-3,6.5 ); // updating object c2,c3 with new values using friend function

update\_circle(c3 ,8,7,5.89);

cout<<"\nData after Updating using friend function :-\n\n";

c1.display\_circle("c1");

c2.display\_circle("c2"); // displaying data members of class after updating.

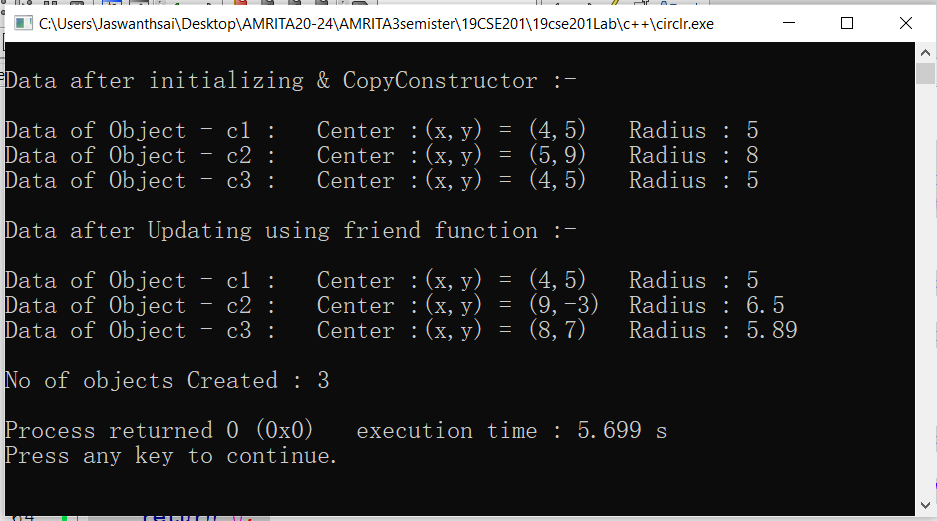
c3.display\_circle("c3");

cout<<"\nNo of objects Created : "<<c2.number\_of\_circles<<"\n";

return 0;

}

Output:



Result:

Thus, the C++ program created for the above problem is successfully executed.