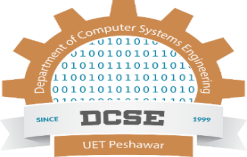
****

**University of engineering & technology Peshawar**

**Object oriented programming-lab**

**Lab report no#05**

**Fall 2020**

**Submitted by: Ashfaq Ahmad**

**Section: B**

**Reg No: 19PWCSE1795**

**Semester: 3rd**

**“On my honor, as a student of University of Engineering and Technology Peshawar, I have neither given nor received unauthorized assistance on this academic work”**

Student signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Submitted to:**

**Eng: summayyea salahuddin**

**Department Of Computer System Engineering**

**Activity 5.4.1**

Reuse Point class of Lab 1 (Activity 1.4.2) as base class. Make the member data protected. Write all

class member functions outside Point class.

* Derive a class Circle from this Point class that has an additional data member: radius of the circle. The point from which this circle is derived represents the center of circle. Provide a no-argument constructor to initialize the radius and center coordinates to 0. Provide a 2-argument constructor: one argument to initialize the radius of circle and the other argument to initialize the center of circle (provide an object of point class in the second argument). Provide a 3-argument constructor that takes three floats to initialize the radius, x-, and y-coordinates of the circle. Provide setter and getter functions for radius of the circle. Provide two functions to determine the radius and circumference of the circle. Write all class member functions outside Circle class.
* Also derive another class Cylinder from the Point class. This class contains an additional data member: radius and height of cylinder. Provide appropriate constructors to initialize the center, radius, and height of the cylinder. Provide functions to determine the area and volume of the cylinder. Area of a cylinder is 2πr\*(r + h). Volume of cylinder is 2πr\*r\*h. Use the findCircum() of circle class where required. Write a main function to test these classes.

**Source code:**

#include <iostream>

using namespace std;

class point

{

protected:

int x,y;

public:

point();

point(int a,int b);

int setX(int a);

int setY(int b);

int getX();

int getY();

void show() const;

};

point::point()

{

x=0;

y=0;

}

point::point(int a,int b)

{

x=a;

y=b;

}

int point::setX(int a)

{

x=a;

}

int point::setY(int b)

{

y=b;

}

int point::getX()

{

return x;

}

int point::getY()

{

return y;

}

void point::show() const

{

cout<<"point is : ("<<x<<","<<y<<")"<<endl;

}

class circle : public point

{

protected:

float re;

public:

circle();

circle(float r, point p);

circle(float r,float a, float b);

void setr(float r);

float getr(); //void

float redius(); //void

float circumferance(); //void

};

circle::circle()

{

re = 0, x = 0, y = 0;

}

circle::circle(float r, point p)

{

re = r;

x = p.getX();

y = p.getY();

}

circle::circle(float r, float a, float b)

{

re = r, x = a, y = b;

}

void circle::setr(float r) //float r

{

re=r;

}

float circle::getr()

{

return re;

}

float circle::circumferance()

{

return 2\*3.14\*re;

}

float circle::redius()

{

return circumferance()/(2\*3.14);

}

class cylander: public point

{

protected:

float re,height;

public:

cylander();

cylander(float r, float h, point p);

cylander(float r, float h, float a,float b);

cylander(float h, circle c);

float area();

float vol();

};

cylander::cylander(): re(0),height(0){}

cylander::cylander(float r, float h, point p)

{

re=r;

height=h;

x=p.getX();

y=p.getY();

}

cylander::cylander(float r, float h, float a, float b)

{

re=r;

height=h;

x=a;

y=b;

}

cylander::cylander(float h, circle c)

{

height=h;

re=c.getr();

x=c.getX();

y=c.getY();

}

float cylander::area()

{

circle c(re, x,y);

return c.circumferance() \* (re+height);

}

float cylander::vol()

{

circle c(re, x,y);

return c.circumferance() \* re \* height;

}

int main()

{

int x,y;

cout<<"Please Enter x-corrdinate: ";

cin>>x;

cout<<"Please Enter y-corrdinate: ";

cin>>y;

point p(x,y);

circle c(x,p);

p.show();

cout<<"Redius is: "<<c.getr()<<endl;

cout<<"circumferance is: "<<c.circumferance()<<endl;

cylander cy(x,c);

cout<<"area of cylander : "<<cy.area()<<endl;

cout<<"voloum of cylander : "<<cy.vol()<<endl;

return 0;

};

Compilation:

