**Jeremy Howell, Nhi Pham**

**CECS 282 – Sec 07**

**9/23/2021**

**Lab 6-1: Inheritance Hierarchy**

Text

Description automatically generated

Demonstrated at 11:24 am on September, 23rd 2021

**pointType.h**

#ifndef H\_PointType

#define H\_PointType

class pointType {

public:

void setPoint(double x, double y);

void print() const;

double getX() const;

double getY() const;

pointType(double x = 0.0, double y = 0.0);

protected:

double xCoordinate;

double yCoordinate;

};

#endif

**pointType.cpp**

#include <iostream>

#include "pointType.h"

using namespace std;

void pointType::setPoint(double x, double y) {

xCoordinate = x;

yCoordinate = y;

}

void pointType::print() const {

cout << "(" << xCoordinate << ", " << yCoordinate << ")" << endl;

}

double pointType::getX() const {

return xCoordinate;

}

double pointType::getY() const {

return yCoordinate;

}

pointType::pointType(double x, double y) {

xCoordinate = x;

yCoordinate = y;

}

**circleType.h**

#ifndef H\_CircleType

#define H\_CircleType

#include "pointType.h"

class circleType: public pointType {

public:

void print() const;

void setRadius(double r);

double getRadius() const;

double getCircumference() const;

double getArea() const;

circleType(double x = 0.0, double y = 0.0, double r = 0.0);

protected:

double radius;

};

#endif

**circleType.cpp**

#include <iostream>

#include "circleType.h"

using namespace std;

void circleType::print() const {

cout << "Center: ";

pointType::print();

cout << endl;

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

cout << "Circumference: " << getCircumference() << endl;

cout << "Area: " << getArea() << endl;

}

void circleType::setRadius(double r) {

radius = r;

}

double circleType::getRadius() const {

return radius;

}

double circleType::getCircumference() const {

return (2 \* 3.1416 \* radius);

}

double circleType::getArea() const {

return (3.1416 \* radius \* radius);

}

circleType::circleType(double x, double y, double r):pointType(x, y) {

radius = r;

}

**cylinderType.h**

#ifndef H\_CylinderType

#define H\_CylinderType

#include "circleType.h"

class cylinderType: public circleType {

public:

void print() const;

void setHeight(double h);

void setBaseCenter(double x, double y);

void setCenterRadiusHeight(double x, double y, double r, double h);

double getHeight() const;

double getVolume() const;

double getSurfaceArea() const;

cylinderType(double x = 0.0, double y = 0.0, double r = 0.0, double h = 0.0);

protected:

double height;

};

#endif

**cylinderType.cpp**

#include <iostream>

#include "cylinderType.h"

using namespace std;

void cylinderType::print() const {

cout << "Base Center: ";

pointType::print();

cout << endl;

cout << "Base Radius: " << circleType::getRadius() << endl;

cout << "Base Circumference: " << circleType::getCircumference() << endl;

cout << "Base Area: " << circleType::getArea() << endl;

cout << "Cylinder height: " << height << endl;

cout << "Cylinder surface area: " << getSurfaceArea() << endl;

cout << "Cylinder volume: " << getVolume() << endl;

}

void cylinderType::setHeight(double h) {

height = h;

}

void cylinderType::setBaseCenter(double x, double y) {

pointType::setPoint(x, y);

}

void cylinderType::setCenterRadiusHeight(double x, double y, double r, double h) {

pointType::setPoint(x, y);

circleType::setRadius(r);

setHeight(h);

}

double cylinderType::getHeight() const {

return height;

}

double cylinderType::getVolume() const {

return (3.1416 \* radius \* radius \* height);

}

double cylinderType::getSurfaceArea() const {

return ((2 \* 3.1416 \* radius \* height) + (2 \* 3.1416 \* radius \* radius));

}

cylinderType::cylinderType(double x, double y, double r, double h):circleType(x,y,r) {

height = h;

}

**main.cpp**

#include <iostream>

#include <iomanip>

#include "cylinderType.h"

using namespace std;

int main() {

cylinderType cylinder1(3, 2.5, 4, 2.5);

cylinderType cylinder2;

cout << fixed << showpoint;

cout << setprecision(2);

cout << "\*\*\*\*\* Cylinder 1 \*\*\*\*\*" << endl;

cylinder1.print();

cout << endl;

cylinder2.setPoint(-2.5, 7);

cylinder2.setRadius(4);

cylinder2.setHeight(3.9);

cout << "\*\*\*\*\* Cylinder 2 \*\*\*\*\*" << endl;

cylinder2.print();

cout << endl;

double x, y;

double r;

double h;

cylinderType cylinder3;

cout << "Enter x Coordinates of the center: ";

cin >> x;

cout << endl;

cout << "Enter y Coordinate of the center: ";

cin >> y;

cout << endl;

cout << "Enter base radius: ";

cin >> r;

cout << endl;

cout << "Enter cylinder height: ";

cin >> h;

cout << endl;

cylinder3.setCenterRadiusHeight(x, y, r, h);

cout << "\*\*\*\*\* Cylinder 3 \*\*\*\*\*" << endl;

cylinder3.print();

cout << endl;

return 0;

}