#include <iostream>

#include <cmath>

using namespace std;

class Shape {

public:

// Конструктор

Shape() : length(0), width(0) {}

Shape(float length) : length(length), width(0) {}

Shape(float length, float width) : length(length), width(width) {}

virtual float calculateArea() { return 0; }

protected:

float length, width;

};

class Circle : public Shape {

public:

// Конструктор

Circle(float radius) : Shape(radius) {}

float calculateArea() override { return 3.14 \* pow(length, 2); }

};

class Rectangle : public Shape {

public:

// Конструктор

Rectangle(float length, float width) : Shape(length, width) {}

float calculateArea() override { return length \* width; }

};

class Triangle : public Shape {

public:

// Конструктор

Triangle(float base, float height) : Shape(base, height) {}

float calculateArea() override { return 0.5 \* length \* width; }

};

int main() {

// Пример использования:

Circle circle(5); // Круг с радиусом 5

Rectangle rectangle(4, 6); // Прямоугольник со сторонами 4 и 6

Triangle triangle(3, 8); // Треугольник с основанием 3 и высотой 8

// Вызов метода calculateArea() для каждой фигуры

cout << "Area of circle: " << circle.calculateArea() << endl;

cout << "Area of rectangle: " << rectangle.calculateArea() << endl;

cout << "Area of triangle: " << triangle.calculateArea() << endl;

return 0;

}