**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра МО ЭВМ**

**ОТЧЕТ**

**по лабораторной работе № 2**

**по дисциплине «Объектно-ориентированное программирование»**

**Тема: Наследование.**

|  |  |  |  |
| --- | --- | --- | --- |
| Студент гр.7303 | |  | Швец А.А. |
| Преподаватель |  |  | Размочаева Н.В. |

г. Санкт-Петербург

2019 г.

**Цель работы:**

Необходимо спроектировать систему классов для моделирования геометрических фигур прямоугольник, эллипс, сектор эллипса. Задание предполагает использование виртуальных функций в иерархии наследования, проектирование и использование абстрактного базового класса. Разработанные классы должны быть наследниками абстрактного класса Shape, содержащего методы для перемещения в указанные координаты, поворота на заданный угол, масштабирования на заданный коэффициент, установки и получения цвета, а также оператор вывода в поток. Необходимо также обеспечить однозначную идентификацию каждого объекта.

Решение должно содержать:

* условие задания;
* UML диаграмму разработанных классов;
* текстовое обоснование проектных решений;
* реализацию классов на языке С++.

**Ход работы:**

Для выполнения поставленной задачи были реализованы следующие структуры данных:

1. Структура point содержит два поля, описывающие координаты х и у точки.
2. Перечисление color, содержащее возможные цвета.
3. Абстрактный класс Shape, содержащий следующие поля: цвет, id фигуры, координаты центра фигуры, вектор, хранящий координаты вершин

фигуры, угол поворота фигуры. Класс Shape содержит следующие методы:

3.1. void setColor(color \_clr) для установления заданного цвета фигуры;

3.2. color getColor() для получения информации о цвете фигуры;

3.3. void move(point \_coordinates) для смещения фигуры в заданную точку;

3.4. void rotate(double \_angle) для поворота фигуры на заданный угол;

3.5. virtual void scale(double resizeCoefficient)=0 - чисто виртуальный метод для масштабирования фигуры;

3.6. virtual void makeCoordinates() = 0 – чисто виртуальный метод для установки координат;

3.7. friend ostream& operator << (ostream& outputStream, Shape& shape) для переопределения оператора вывода на экран.

1. Класс Square, который наследуется от абстрактного класса Shape. Класс имеет поле, характеризующее сторону квадрата. В классе были переопределены методы scale и makeCoordinates.
2. Класс Ellipse, который наследуется от абстрактного класса Shape. Класс Ellipse имеет поля, характеризующие длины диаметров эллипса. В классе был переопределены методы scale и makeCoordinates.
3. Класс Rhombus, который наследуется от абстрактного класса Shape. Класс Rhombus имеет поля, характеризующие длину его стороны и обе диагонали.

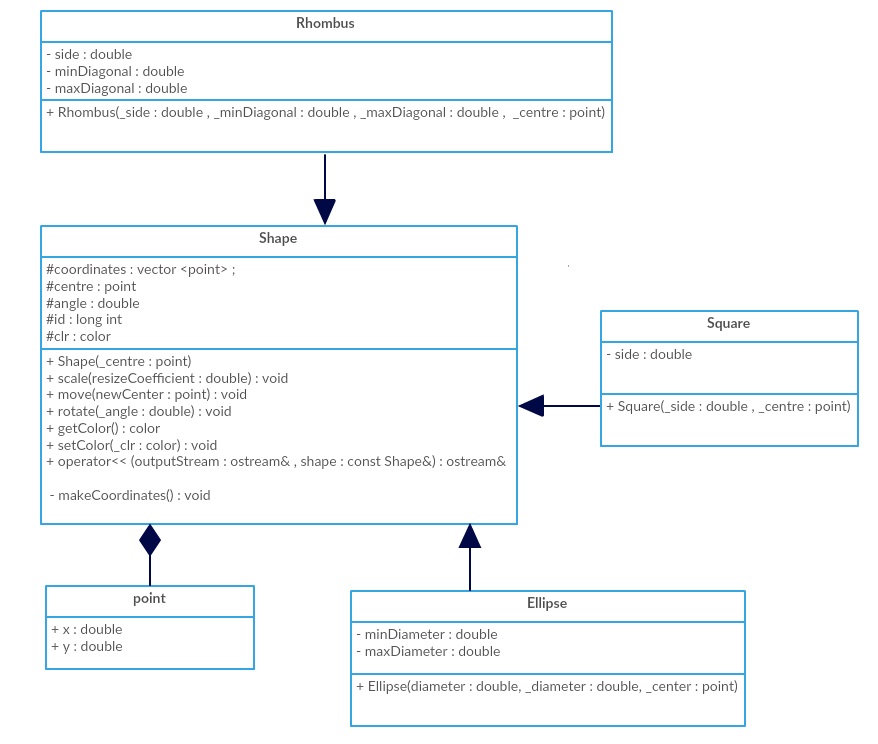
В классе были переопределены методы scale и makeCoordinates.

**Обоснование решения:**

* + данной лабораторной работе был реализован абстрактный класс Shape.
* нем содержатся поля, которые являются общими для всех фигур: id, цвет, координаты вершин, угол поворота фигуры, координаты центра. Также в нем описаны методы, реализация которых общая для всех фигур: move, в котором происходит изменения координат центра в соответствии с новыми с последующим вызовом функции makeCoordinates, чтобы заново построить там фигуру; rotate, в котором все координаты умножаются на матрицу поворота, для получения координат повернутой фигуры. Также описаны методы задания и получения цвета.

Для реализации прямоугольника потребовались дополнительное поле - side. Для реализации ромба потребовались дополнительные поля с диагоналями ромба и его стороной. Для реализации эллипса потребовались дополнительные поля, задающие его диаметры.

**UML диаграмма классов**



**Заключение**

* ходе выполнения данной лабораторной работы была изучена тема наследование. Была спроектирована система классов для моделирования геометрических фигур: квадрат, эллипс, ромб. Были использованы виртуальные функции в иерархии наследования, а также были разработаны классы, которые являются наследниками абстрактного класса Shape.