

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.

Студентка гр.7304

Дегтярева 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 C++.

Ход работы:

Для выполнения поставленной задачи были реализованы следующие структуры данных:

1. Структура Point содержит два поля, описывающие координаты x и y 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 set_color(color _clr) для установления заданного цвета фигуры;
 - 3.2. color get_color() 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 coefficient)=0 - чисто виртуальный метод для масштабирования фигуры;
 - 3.6. virtual void make_coordinates() = 0 – чисто виртуальный метод для установки координат;
 - 3.7. friend ostream& operator << (ostream& out, Shape& shape) для переопределения оператора вывода на экран.
4. Класс Rectangle, который наследуется от абстрактного класса Shape. Класс имеет поля, характеризующие длину и ширину прямоугольника. В классе был переопределены методы scale и make_coordinates.
5. Класс Ellipse, который наследуется от абстрактного класса Shape. Класс Ellipse имеет поля, характеризующие длины диаметров эллипса. В классе был переопределены методы scale и make_coordinates.
6. Класс EllipseSector который наследуется от абстрактного класса Shape. В классе определена структура radius, содержащая радиус сектора и его угол отклонения от оси x. Класс EllipseSector имеет поля, характеризующие

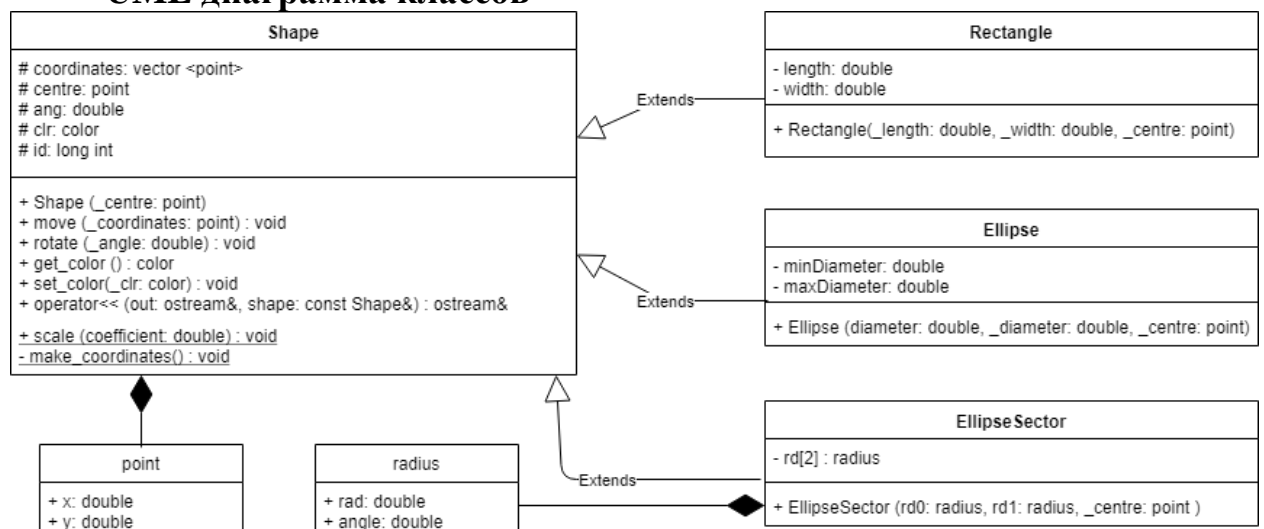
ограничивающие его радиусы. В классе был переопределены методы `scale` и `make_coordinates`.

Обоснование решения:

В данной лабораторной работе был реализован абстрактный класс `Shape`. В нем содержатся поля, которые являются общими для всех фигур: `id`, цвет, координаты вершин, угол поворота фигуры, координаты центра. Также в нем прописаны методы, реализация которых общая для всех фигур: `move`, в котором происходит передвижение координат центра в заданную точку и вызов функции `make_coordinates`, чтобы заново построить там фигуру; `rotate`, в котором все координаты умножаются на матрицу поворота. Также методы задания и получения цвета.

Для реализации прямоугольника потребовались дополнительные поля - `length` и `width`. Для реализации сектора эллипса потребовались дополнительные поля с диаметрами эллипса. Для реализации сектора эллипса потребовались дополнительные поля с радиусами, задающими эллипс.

UML диаграмма классов



Заключение

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