Государственное бюджетное общеобразовательное учреждение «Президентский физико-математической лицей № 239»

Поиск такой пары точек, что проведенные по ним две прямые пересекаются с окружностью на минимальном расстоянии друг от друга

Годовой проект по информатике

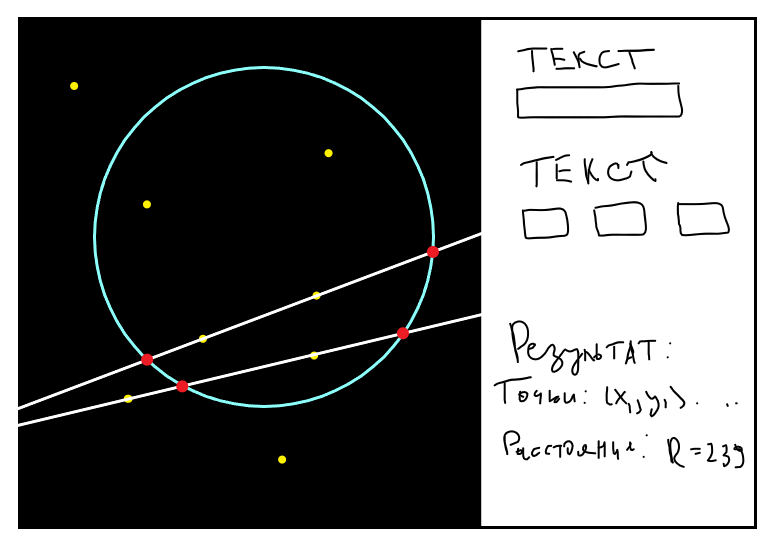
Козлова Таисия, 10–2 класс

Санкт-Петербург 2021

**1. Постановка задачи**

На плоскости имеется окружность и некоторое множество точек. Нужно найти две такие пары точек (точки не должны повторяться), что построенные по ним две прямые пересекаются с окружностью на минимальном расстоянии друг от друга. Требуется графически выделить найденные пары точек, построить соответствующие им прямые и их пересечения с окружностью.

**2. Планируемый внешний вид программы**



**3. Уточнение исходных и выходных данных:**

*Исходные данные:*

Множество точек, задающихся или вручную, или случайно (их координаты лежат от -1 до 1 по Ox и Oy); окружность с определенным центром (координаты от -1 до 1 по Ox и Oy) и радиусом, не большим 1.

*Выходные данные:*

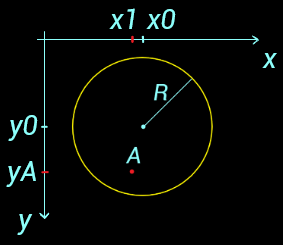
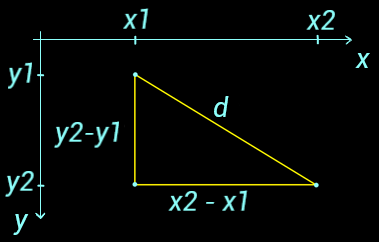
Нарисованы прямые и четыре выделенные другим цветом точки, по которым эти прямые построены. Выведены координаты этих точек в отдельном поле.

**4. Математическая модель**

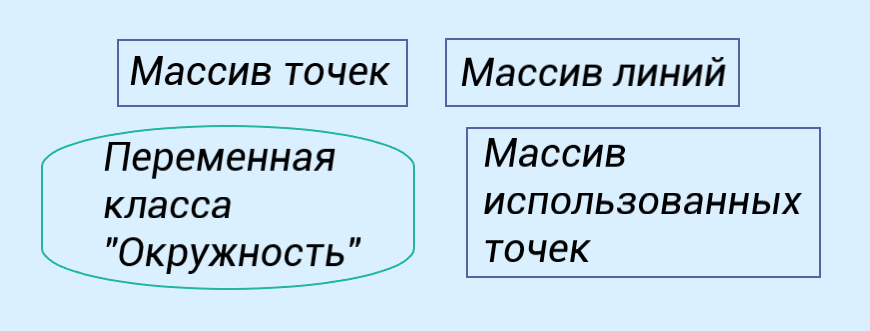
Теорема Пифагора:

Уравнение окружности:

Уравнение прямой:



**5. Анализ используемой структуры данных**

****

Использование отдельного класса для описания окружности облегчает работу с поиском пересечений. Также использование такой структуры данных, как java.util.ArrayList дает возможность избежать ограничений на количество точек (эти массивы расширяются сами при добавлении в них элементов).

**6. Метод решения**

Решение данной задачи можно условно разбить на 4 этапа:

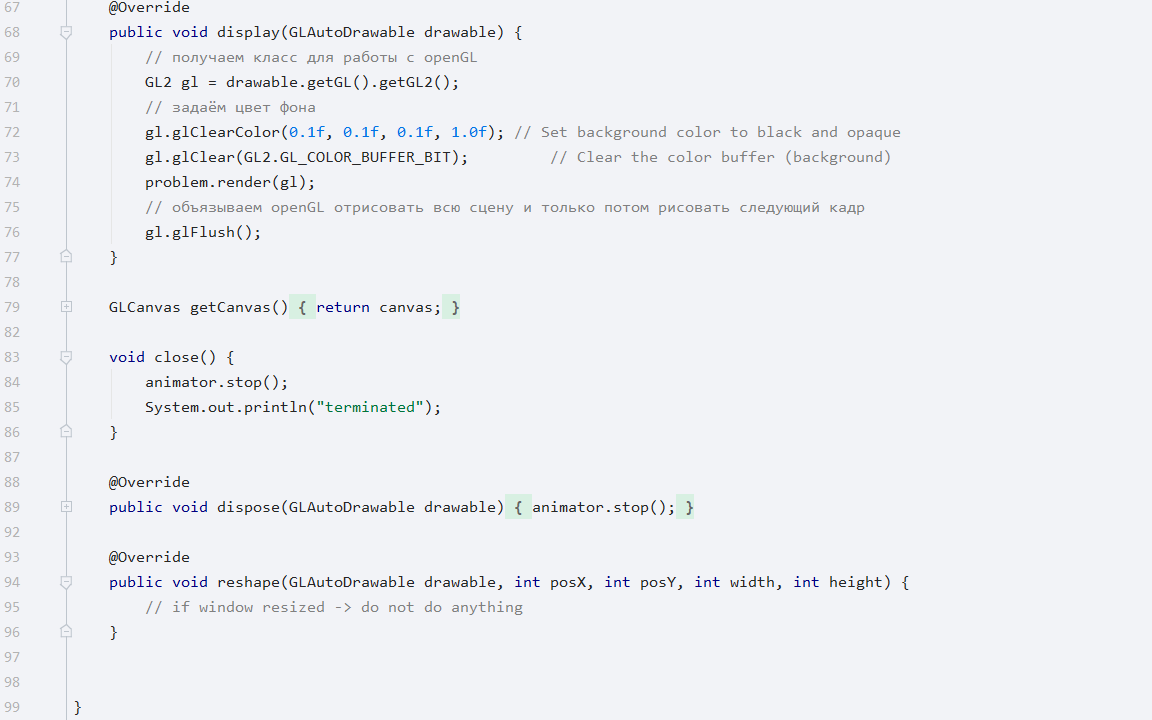
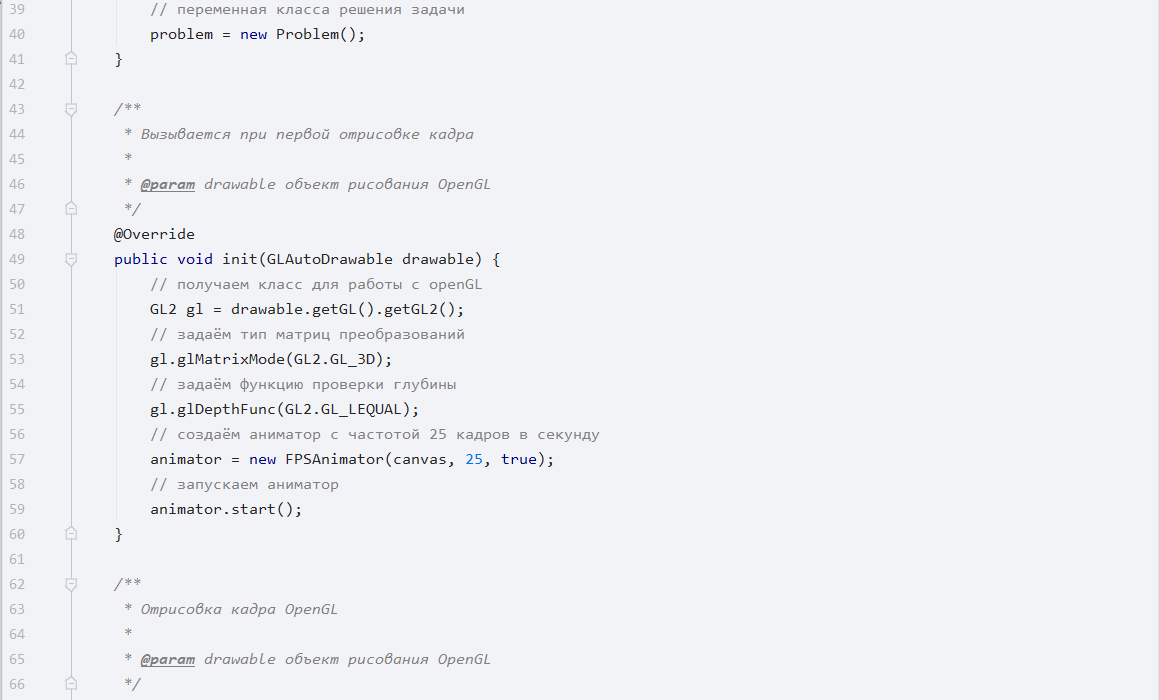
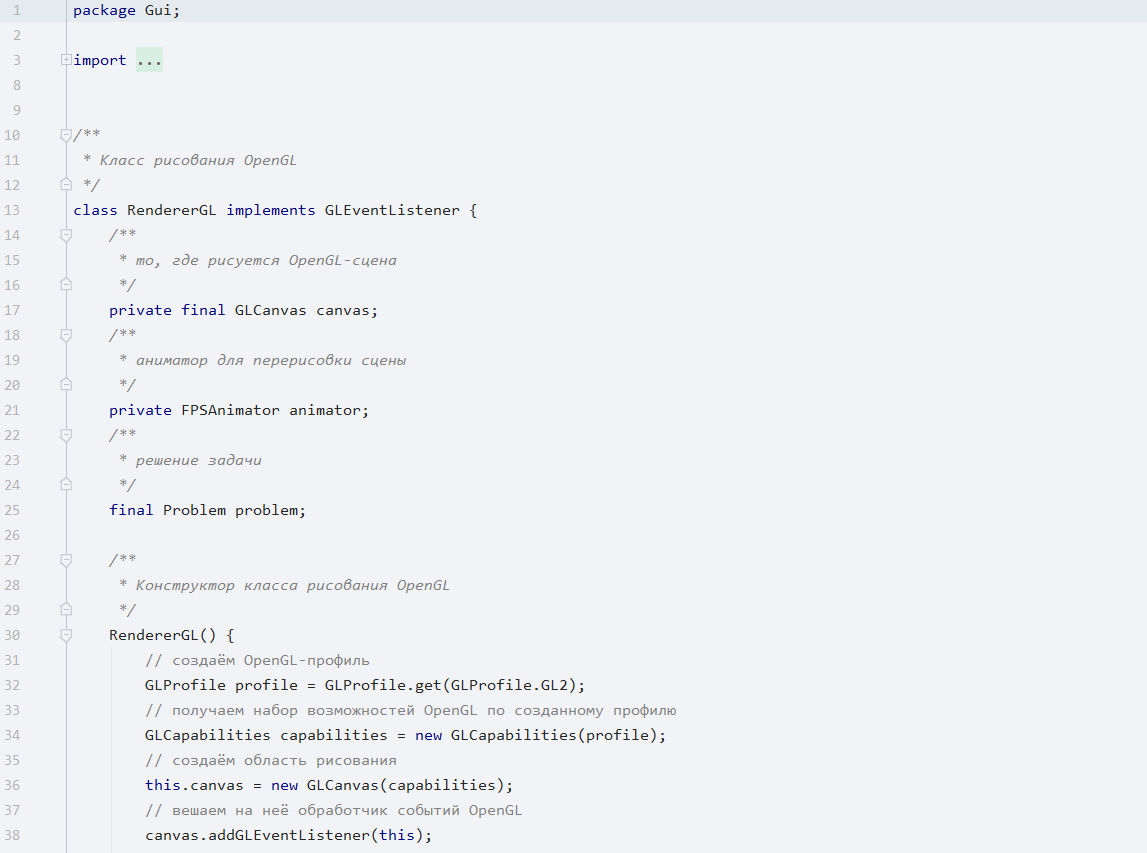
1. Построение прямых по каждой паре точек (нахождение их уравнений) и занесение в отдельный массив
2. Нахождение пересечений этих прямых с окружностью
3. Поиск двух прямых в соответствующем массиве, для которых расстояние (дуга) между их пересечениями с окружностью минимальна
4. Построение найденных двух прямых, выделение соответствующих им точек и точек их пересечения с окружностью цветом

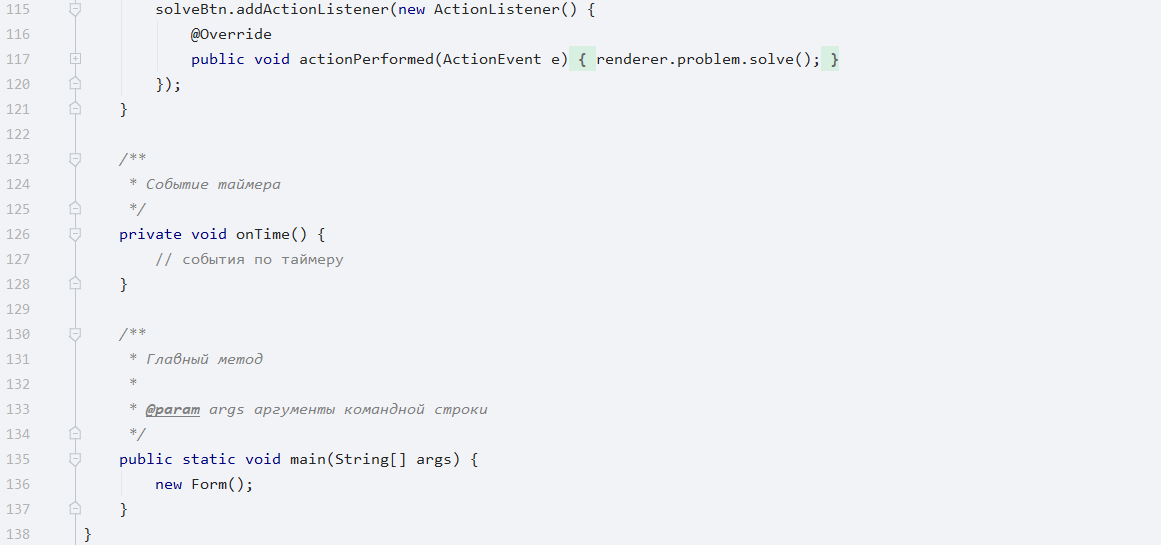
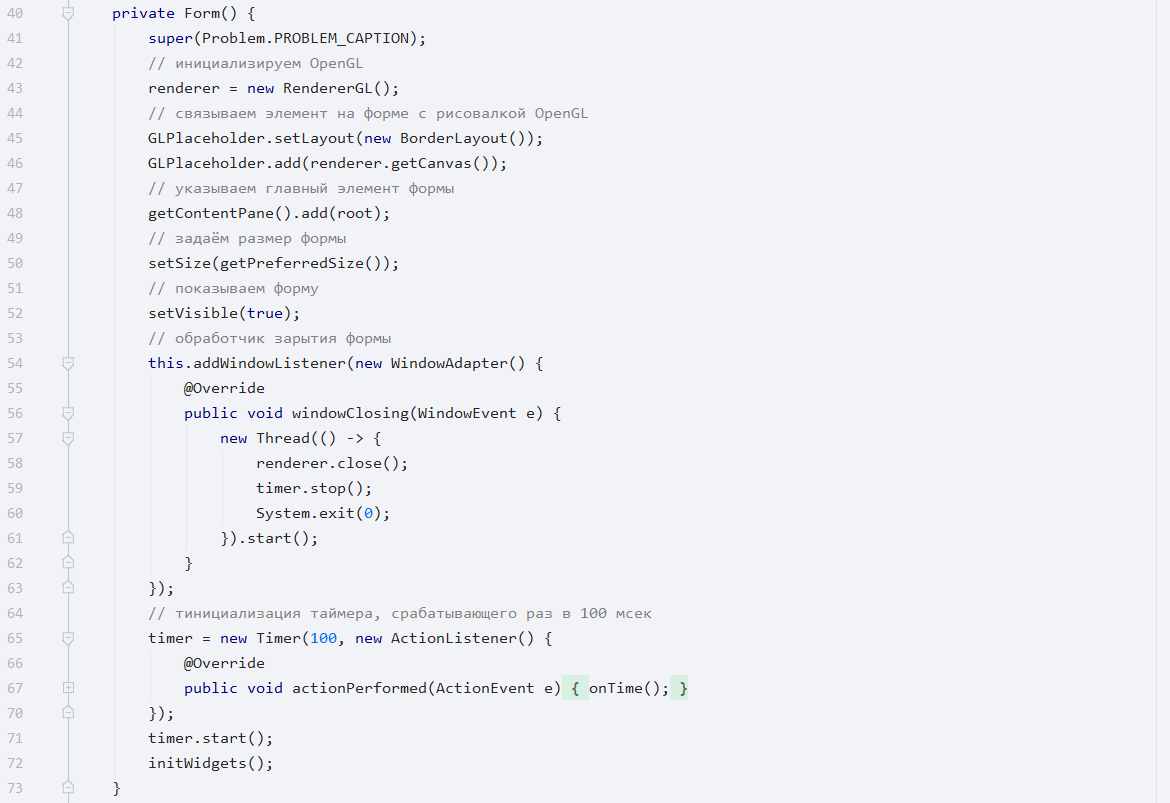
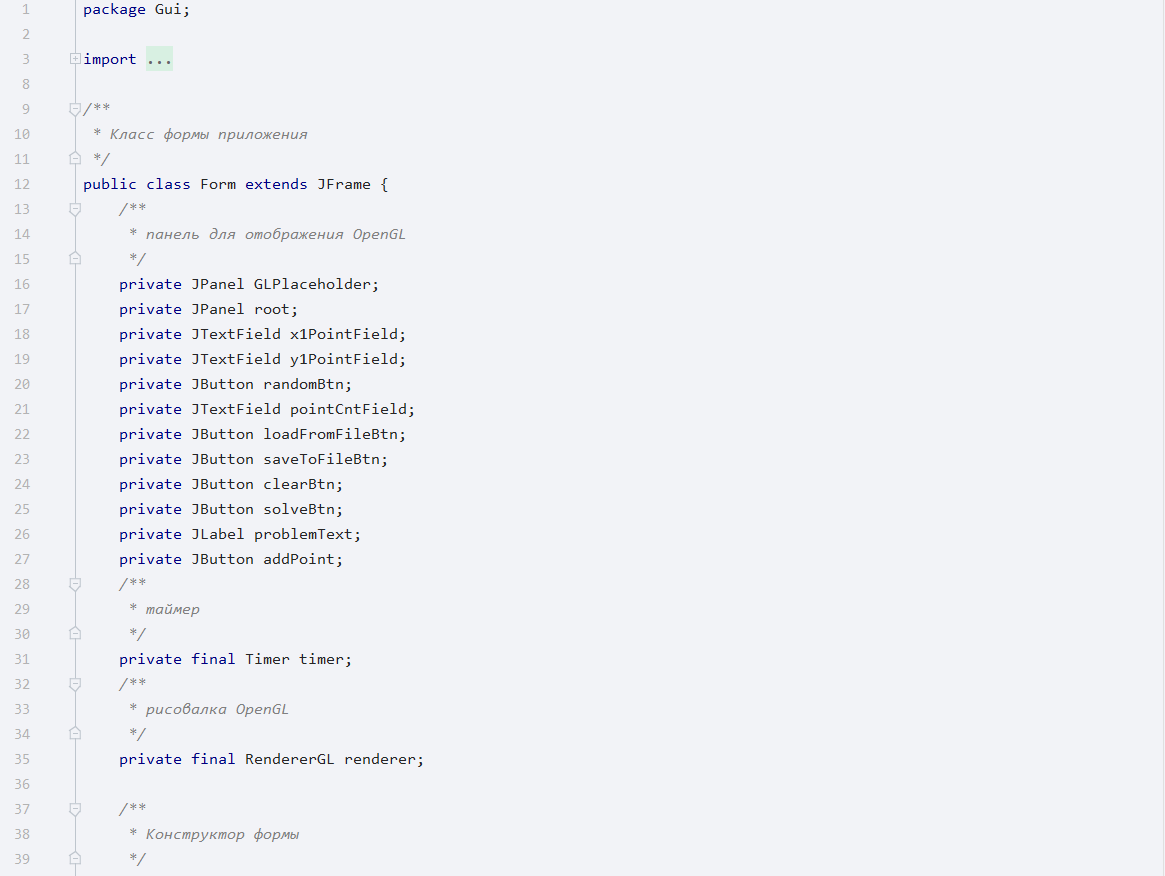
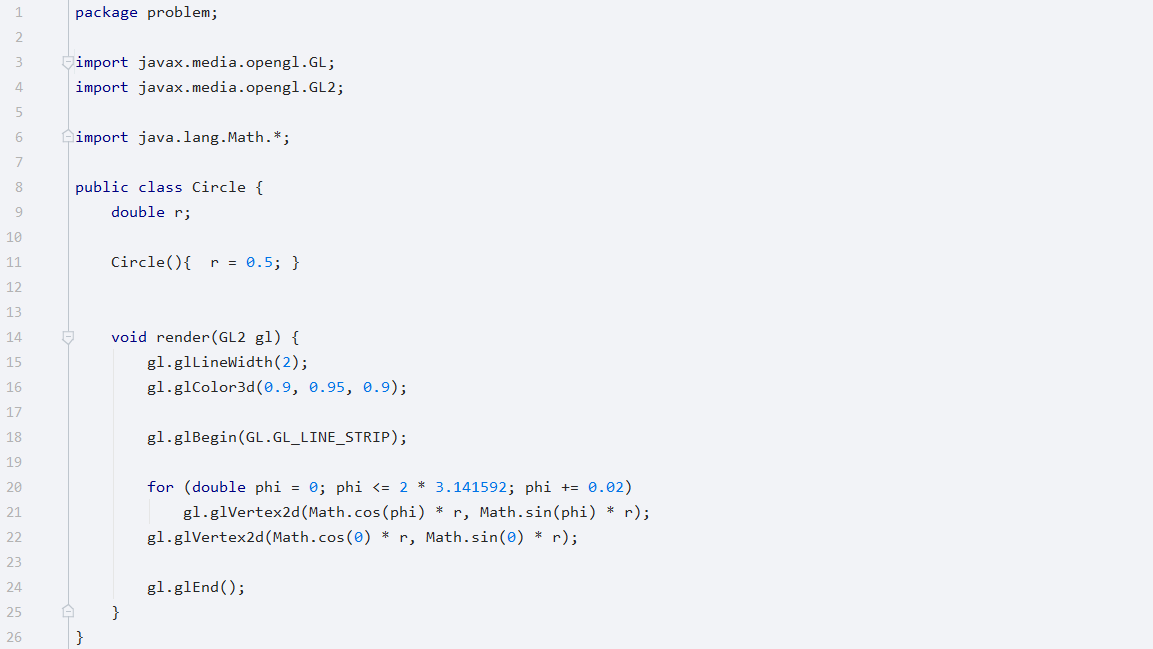
**7. Структура программы**

****

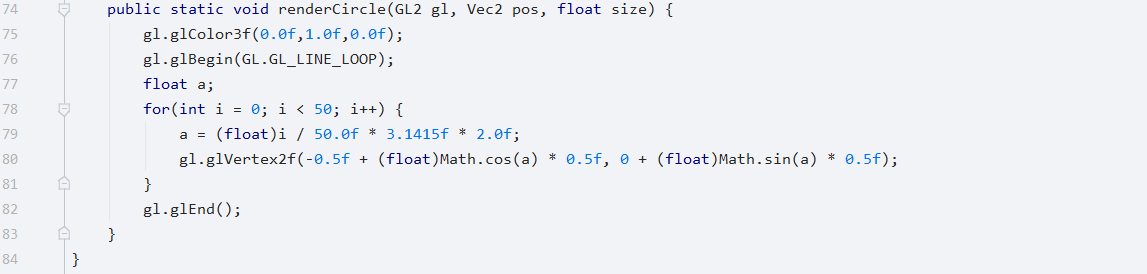
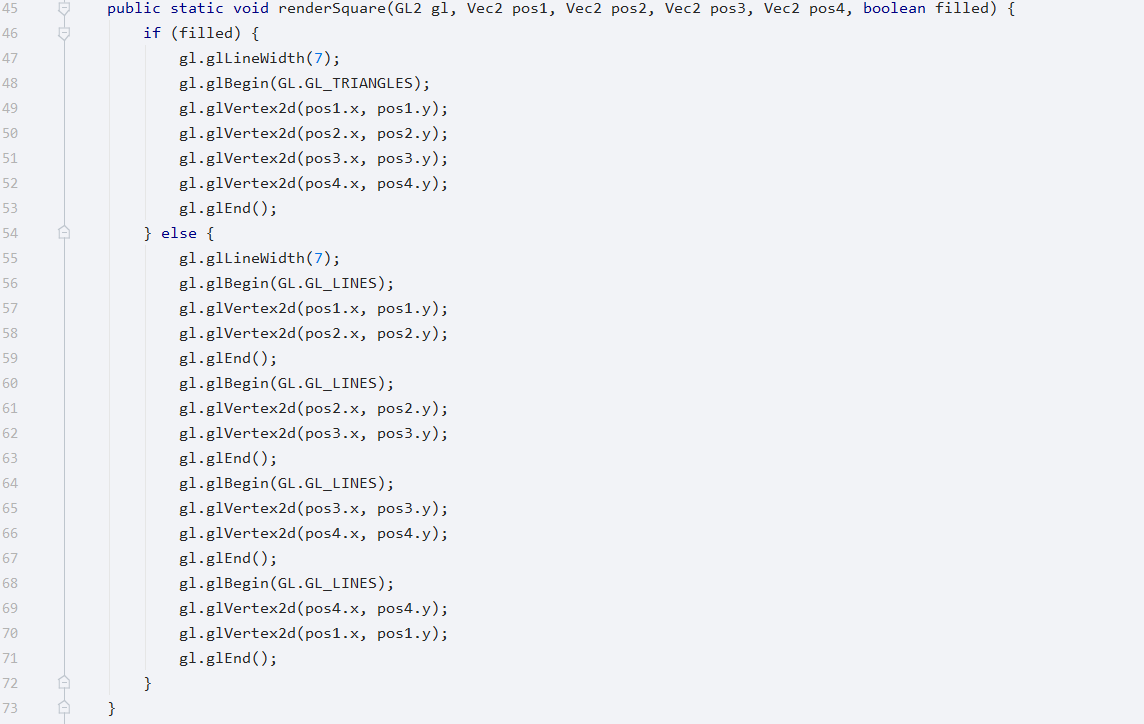
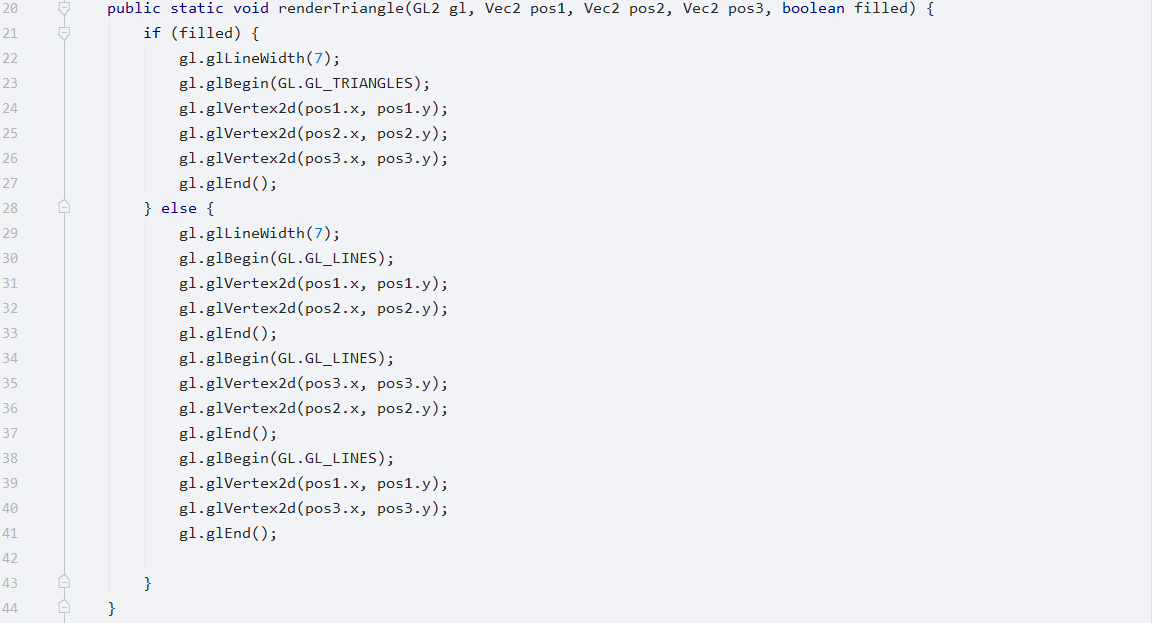
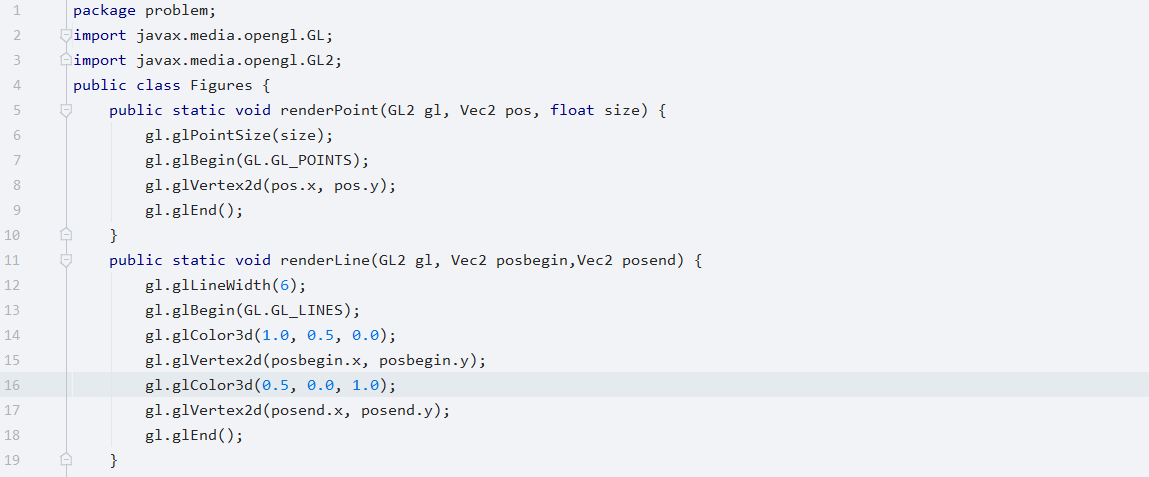
**8. Листинг программы**

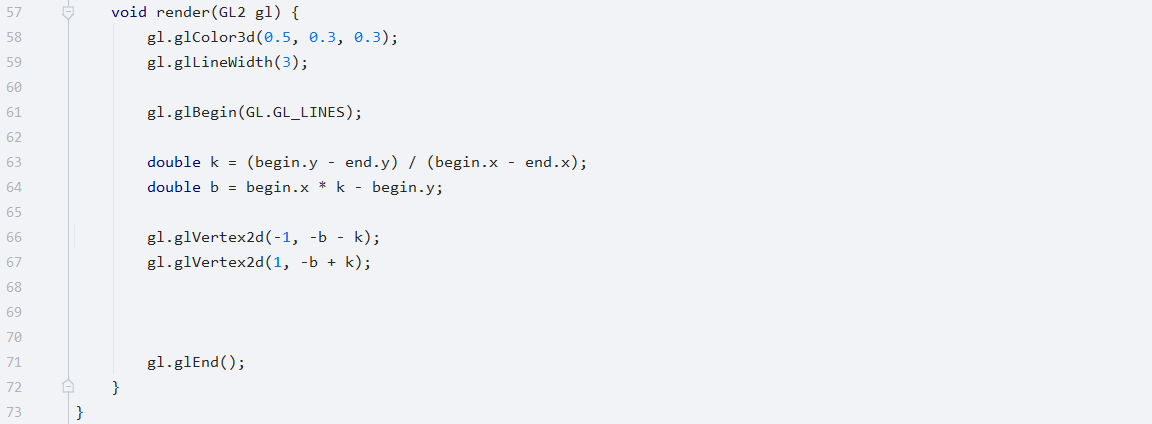
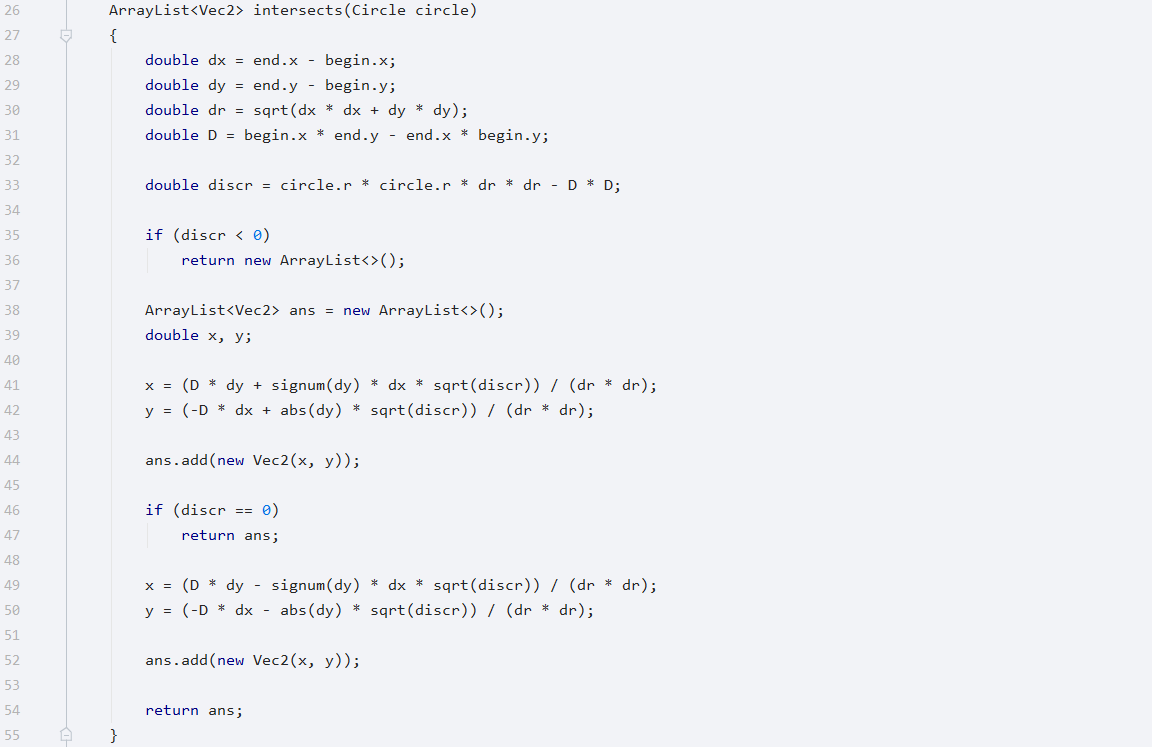
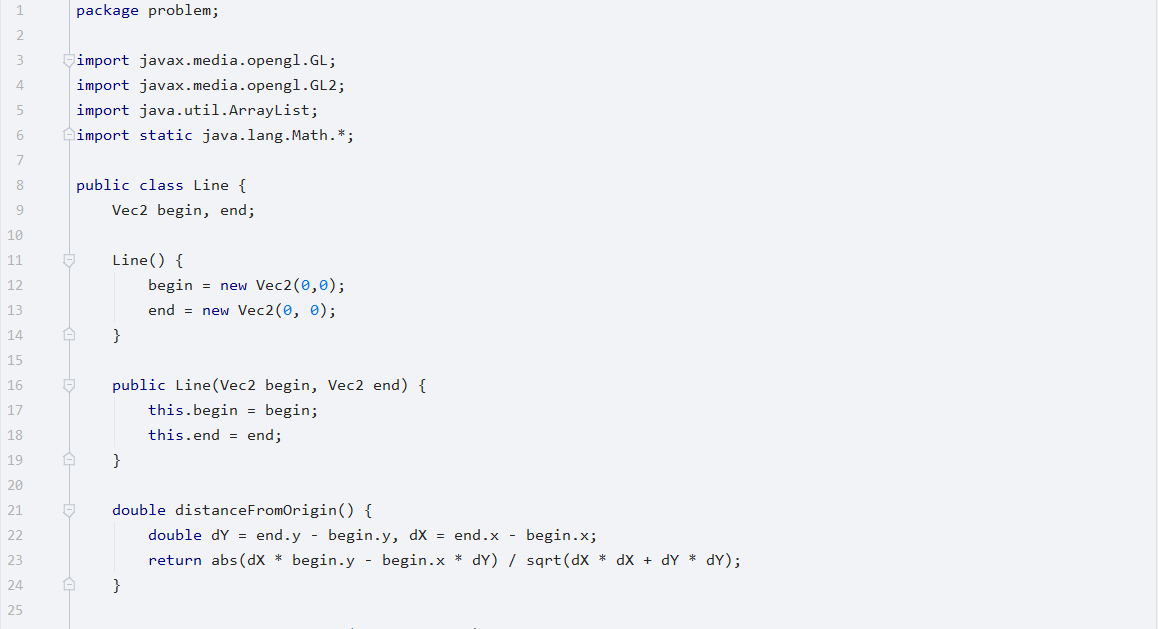
Класс RenderGL.java

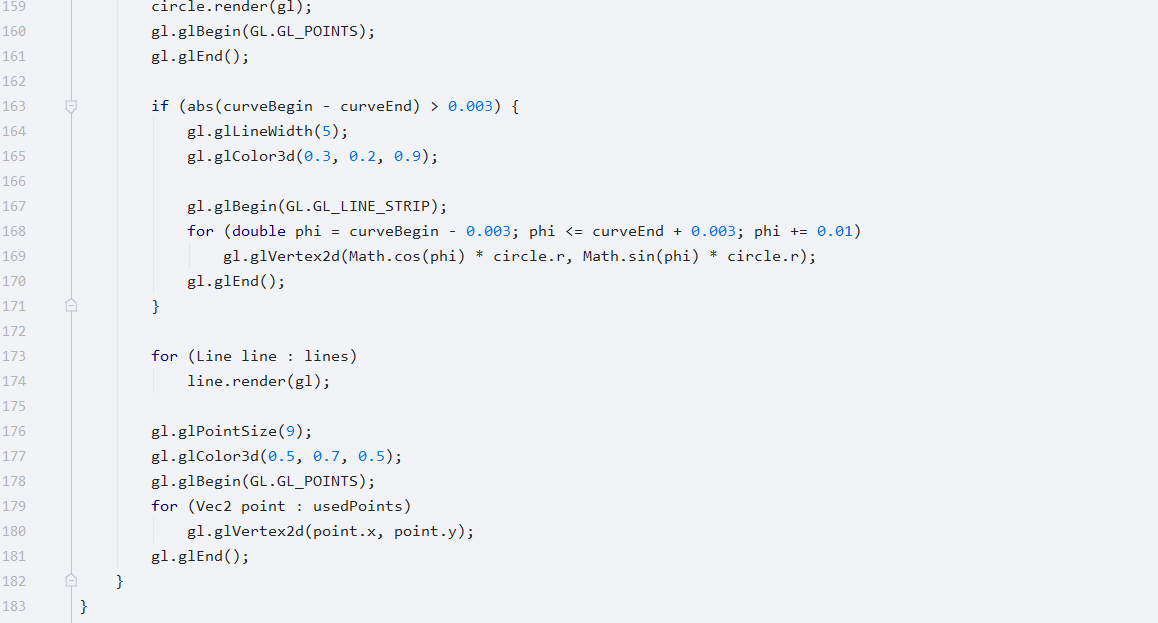
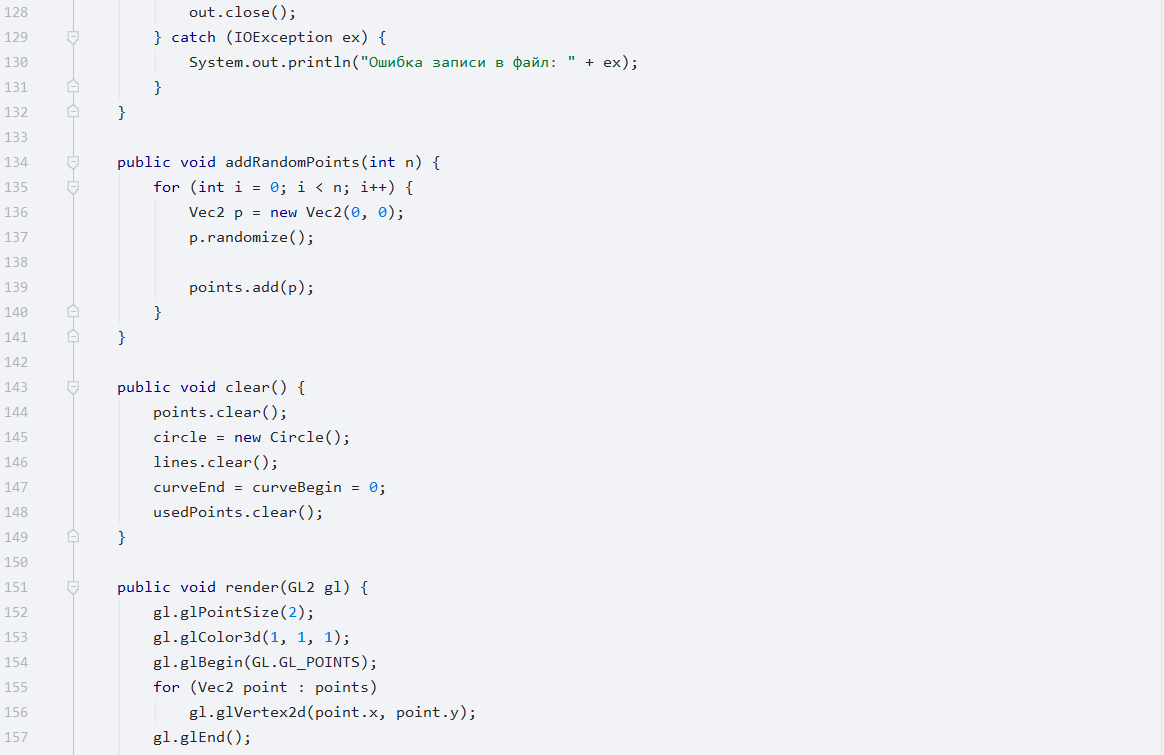
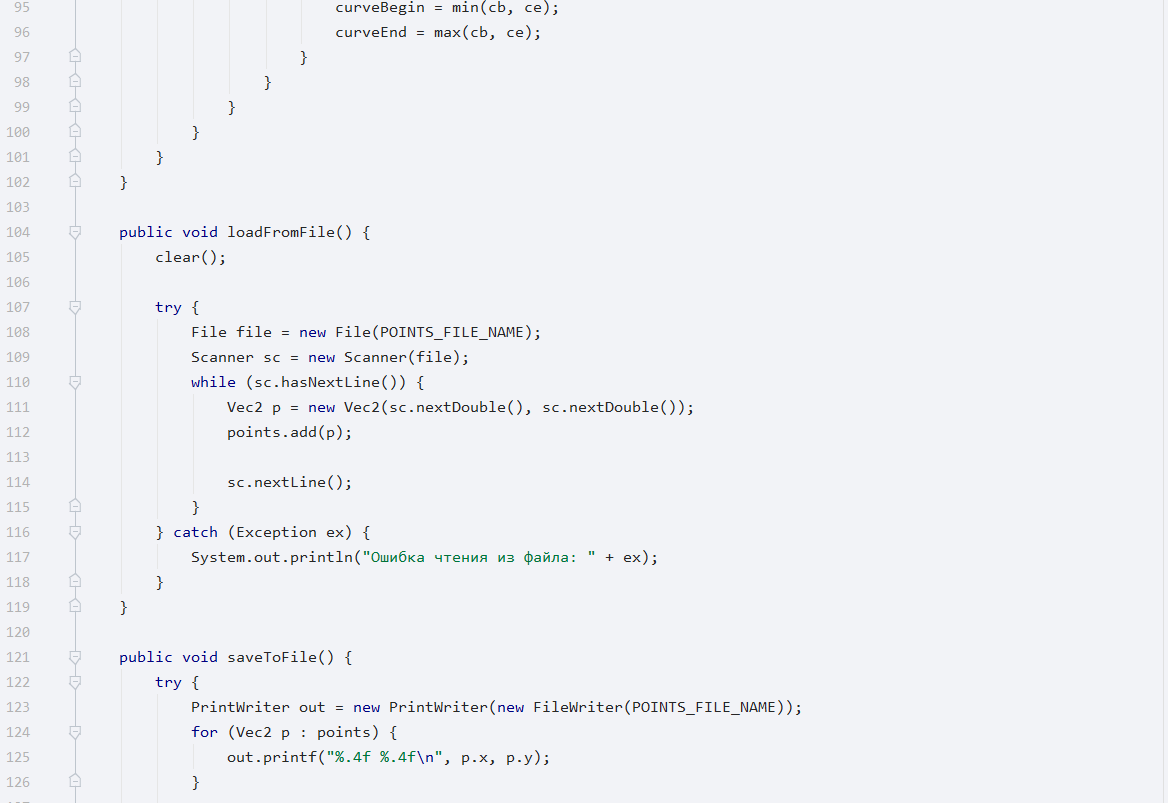
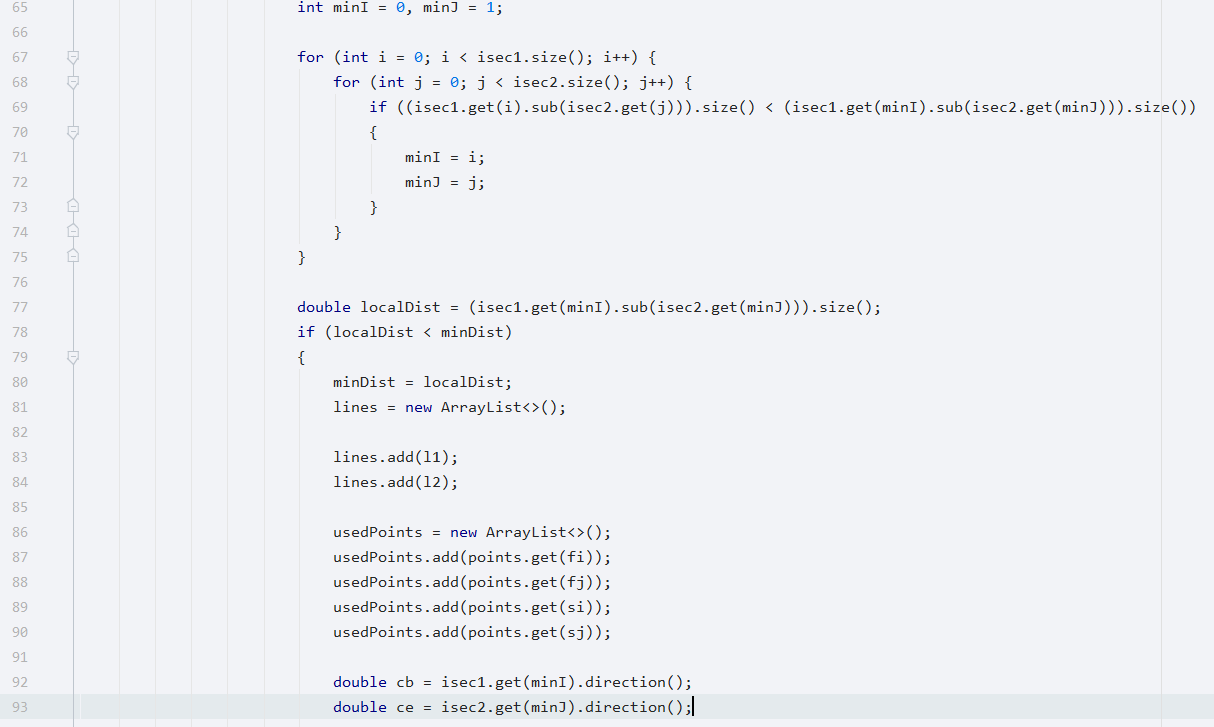
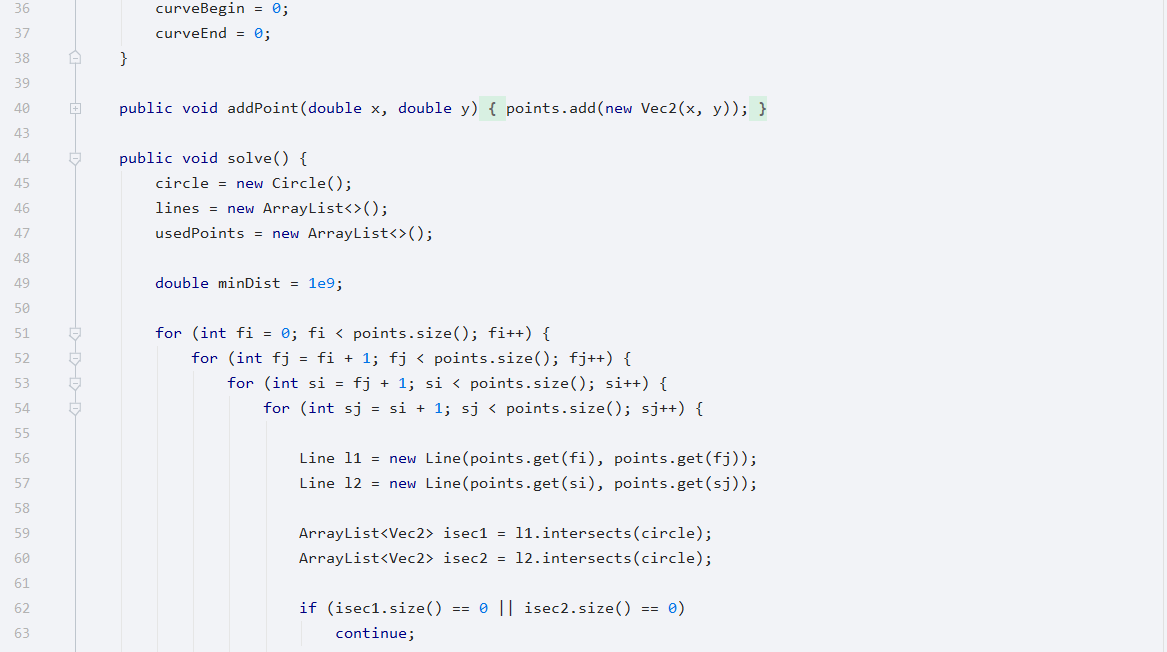
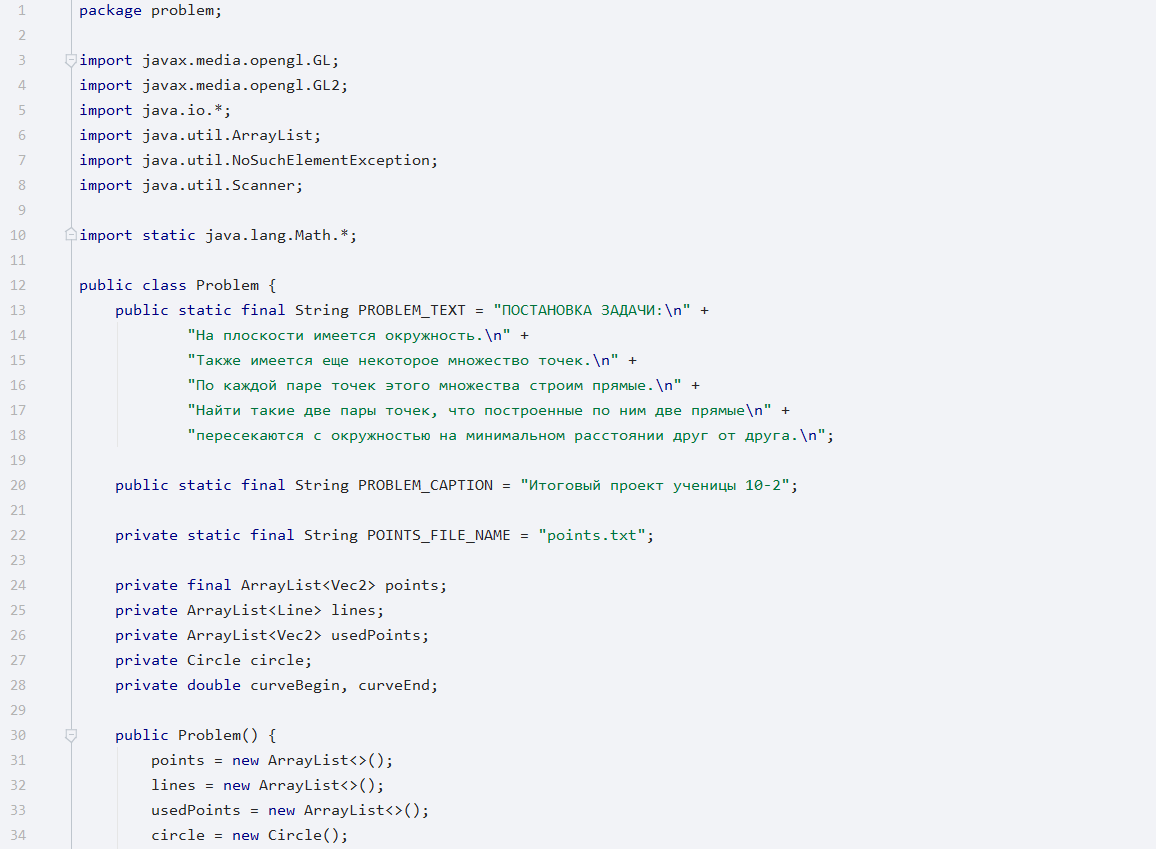
****

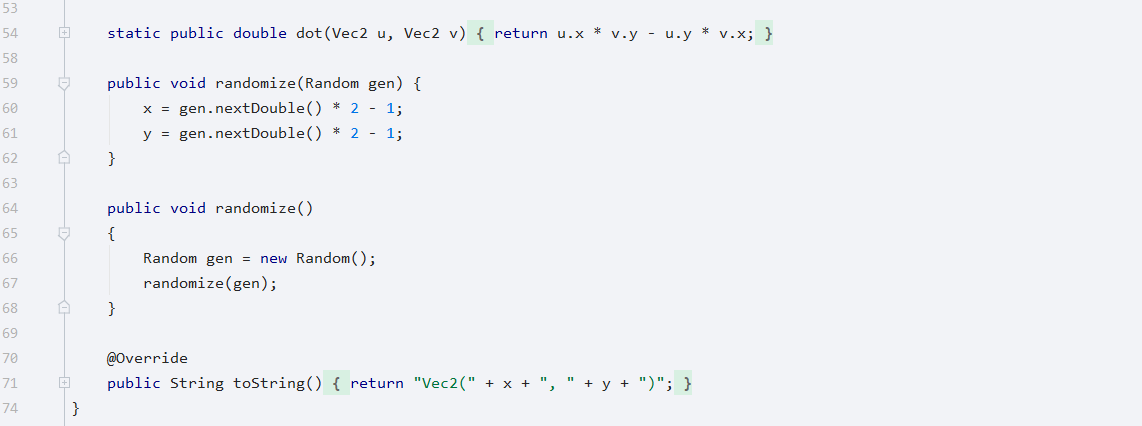
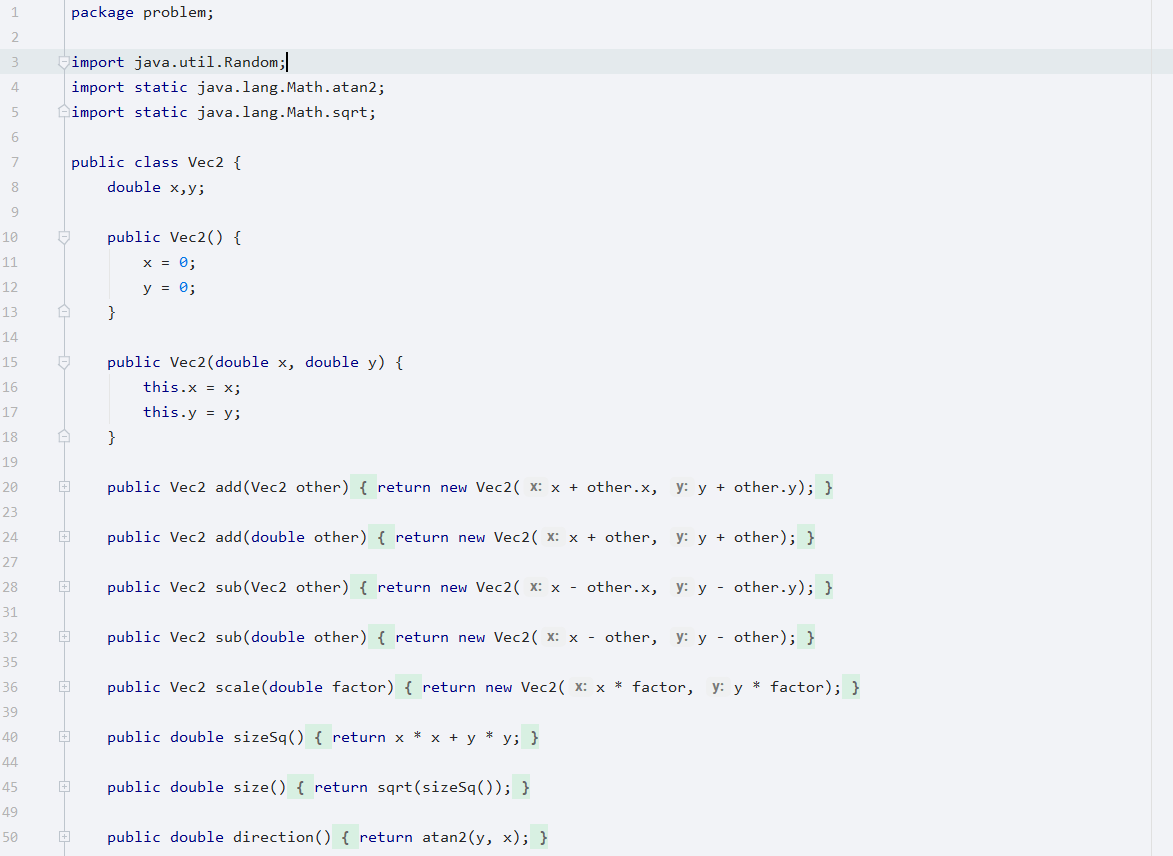
Класс Form.java****Класс Circle.java****

Класс Figures.java

****

Класс Line.java****

Класс Problem.java****

Класс Vec2.java****

**9. Пример работы программы**

**10. Анализ правильности решения**

С помощью добавления случайной генерации точек легко проверить действия программы на случайных данных. Во всех зафиксированных мной случаях программа работала корректно. Так же с помощью задания точек из файла можно проверить, насколько хорошо работают граничные случаи – например, случай касания одной из найденных прямых окружности. Такие проверки тоже прошли успешно.

**11. Использованные источники**

http://esate.ru/uroki/OpenGL/uroki-OpenGL-c-sharp/osnovy-vizualizatsii-2D-primitivov-openGL/

https://stackoverflow.com/