**Комитет по образованию г. Санкт-Петербург**

**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**ПРЕЗИДЕНТСКИЙ ФИЗИКО-МАТЕМАТИЧЕСКИЙ**

**ЛИЦЕЙ №239**

**Отчет о практике**

**«Создание графических приложений на языке Java»**

Учащийся 10-3 класса

Иванов А.В.

Преподаватель:

Клюнин А.О.

Санкт-Петербург – 2023 год

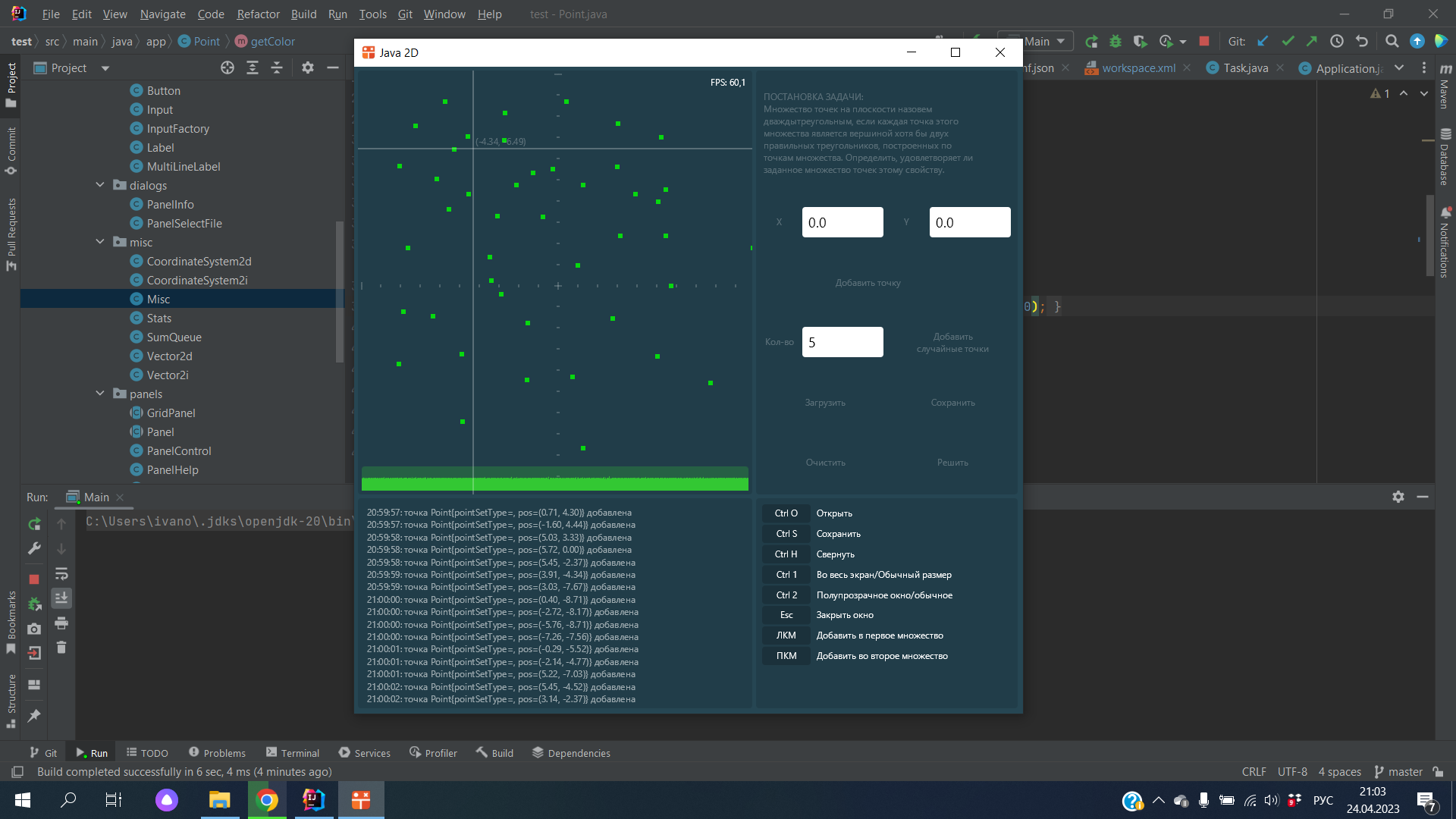
# 1. Постановка задачи

Множество точек на плоскости назовём дваждытреугольным, если каждая точка этого множества является вершиной хотя бы двух правильных треугольников, построенных по точкам множества. Определите, удовлетворяет ли заданное множество точек этому свойству.



# 2. Элементы управления

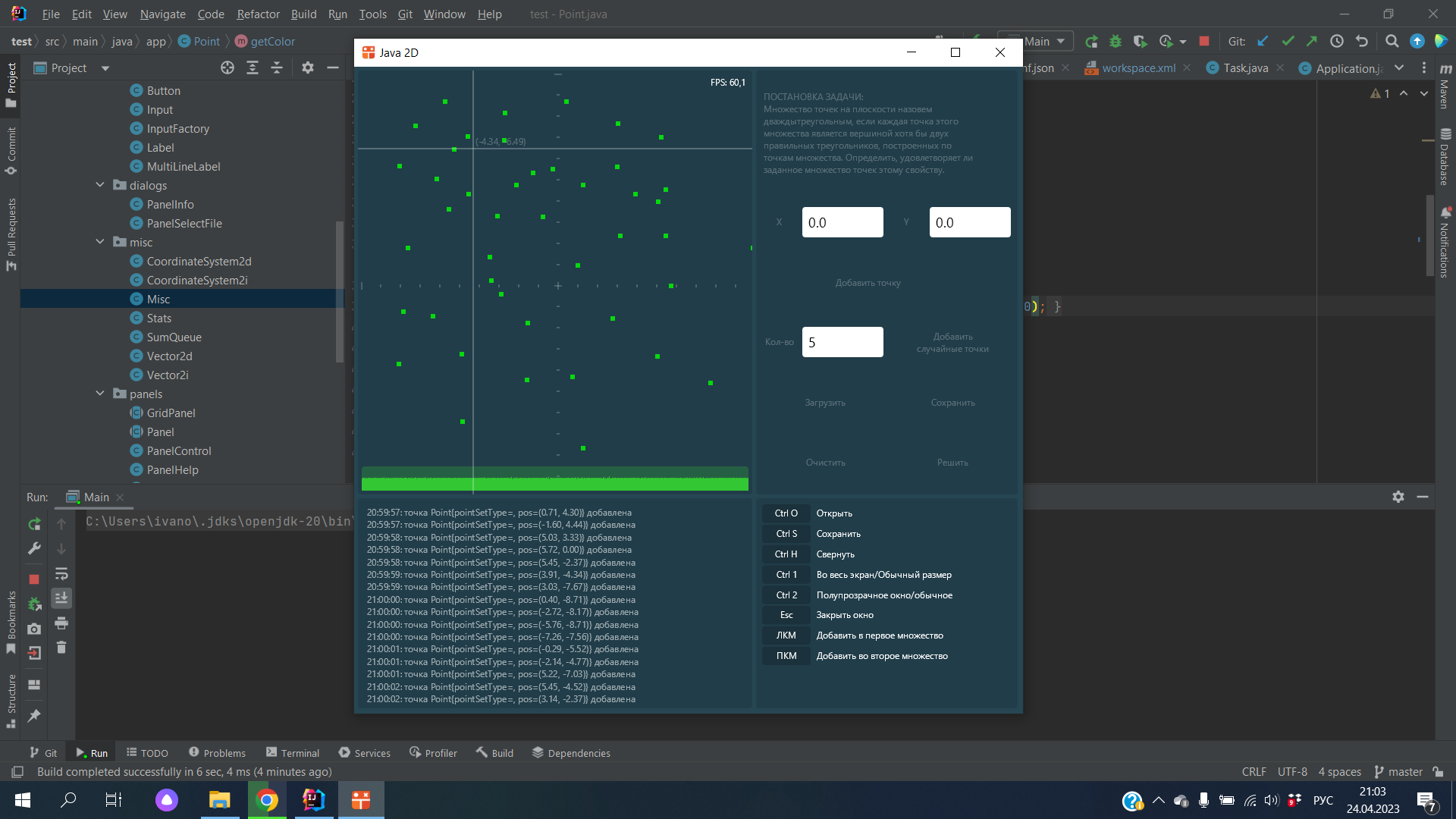
В рамках данной задачи необходимо было реализовать следующие элементы управления:



Для добавления точки по координатам было создано два поля ввода: «X» и «Y», а также кнопка «Добавить точку», по нажатию которой в область рисования добавляется точка с введёнными координатами.

Т.к. задача предполагает ввод только одного вида геометрических объектов, то для добавления случайных элементов достаточно одного поля ввода. В него вводится количество случайных точек, которые будут добавлены.

Также программа позволяет добавлять точки с помощью клика мышью по области рисования.



При клике левой кнопкой мыши по области рисования в месте клика создаётся точка.

# 3. Структуры данных

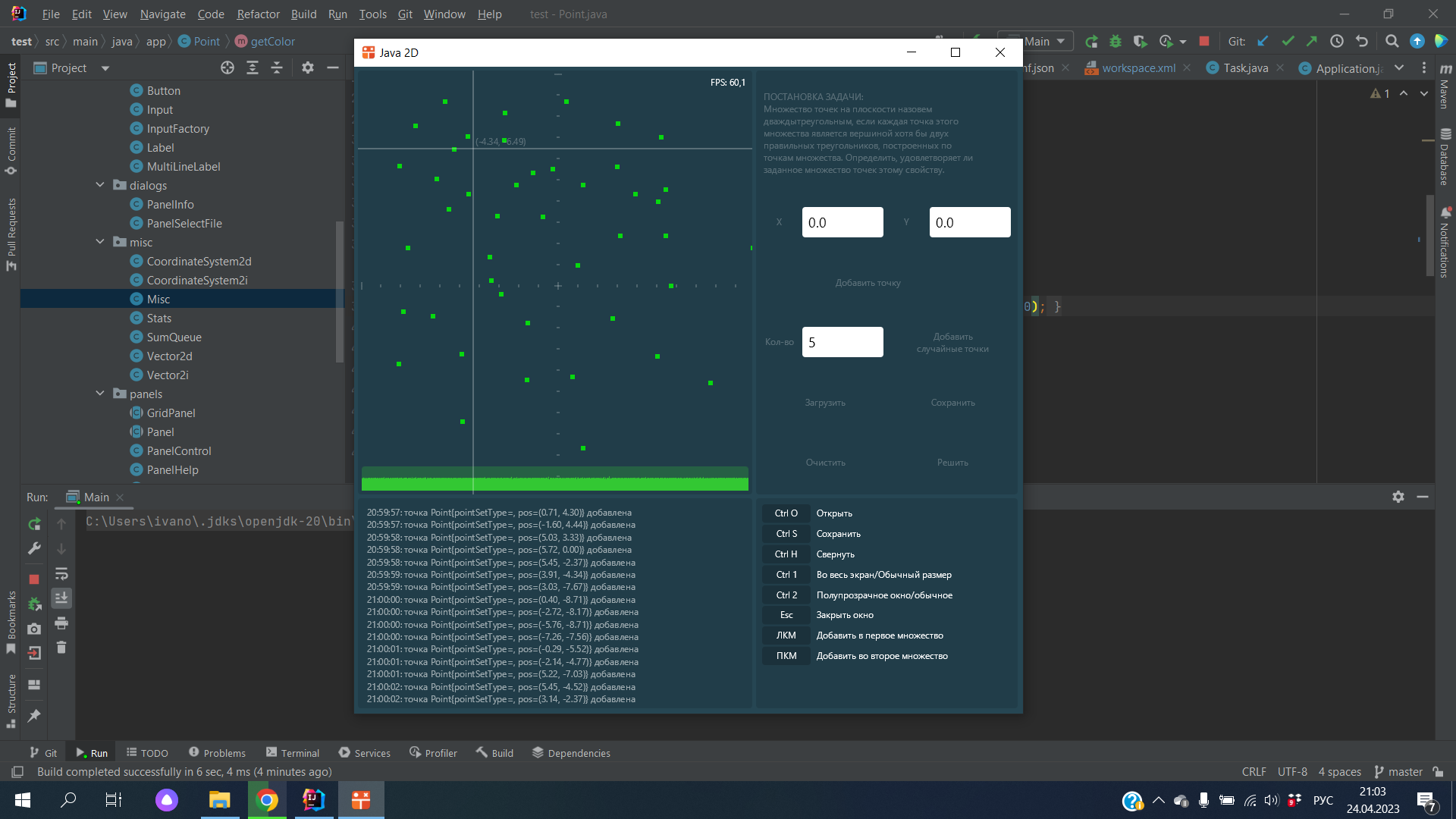
Для того чтобы хранить точки, был разработан класс **Point.java.** Его листинг приведён в приложении А. В него было добавлено поле **pos**, соответствующее положению точки в пространстве задачи.

Для хранения треугольников был разработан класс **Triangle.java.** Его листинг

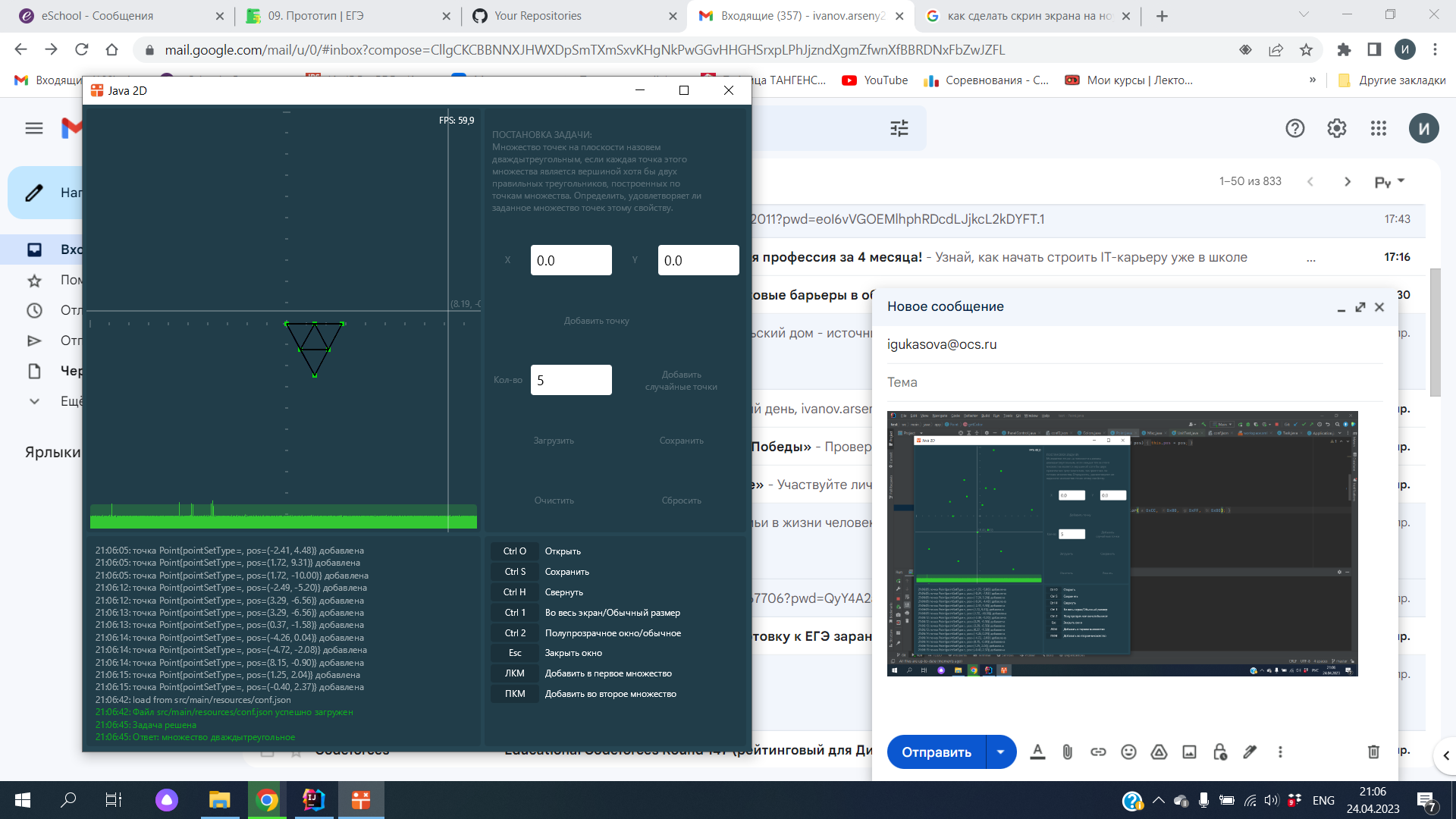
приведён в приложении Б. В него были добавлены метод **Triangle** для хранения вершин треугольника и метод **render** для рисования треугольника.

# 4. Рисование

Чтобы нарисовать точку, использовалась команда рисования прямоугольников **canvas.drawRect().**

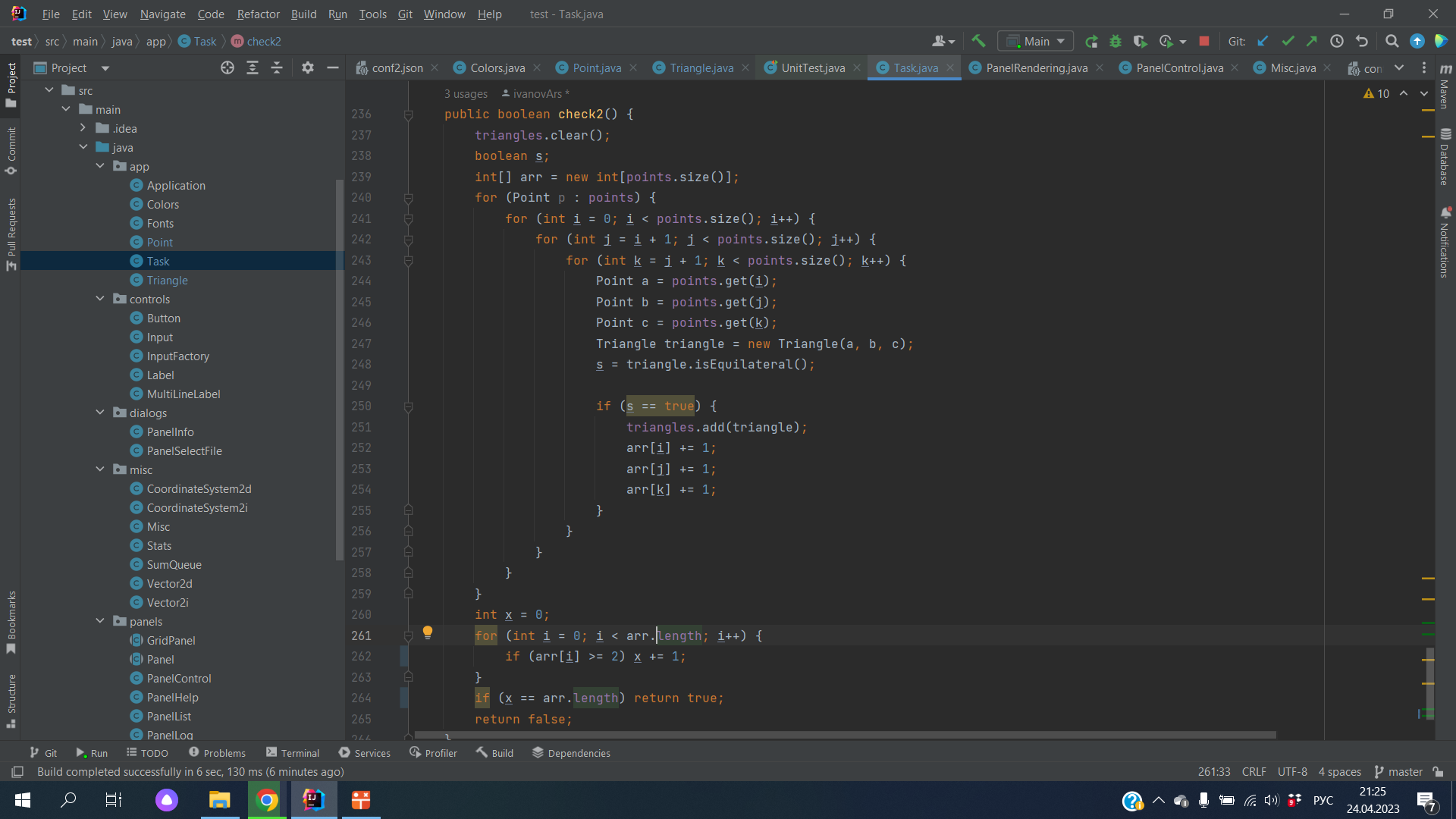


Для рисования треугольников использовалась команда рисования прямых между двумя точками с заданными координатами **canvas.drawLine().**

****

# 5. Решение задачи

Для решения поставленной задачи в классе **Task** был разработан метод **check2().**



В нём создаётся целочисленный массив, отвечающий за количество равносторонних треугольников, вершиной которых является та или иная точка множества (по умолчанию все элементы начального массива равны 0). Затем в методе перебираются все тройки точек. Если треугольник, образуемый тремя какими-то точками, равносторонний (расстояния между всеми парами точек равны между собой), для всех этих точек счётчик правильных треугольников увеличивается на 1 (элементы массива с соответствующими индексами увеличиваются на 1).

Если после перебора всех возможных троек количество правильных треугольников для каждой точки больше 1, это означает, что каждая точка является вершиной минимум двух равносторонних треугольников, а значит данное множество дваждытреугольное. В противном случае – нет.

# 6. Проверка

Для проверки правильности решённой задачи были разработаны unit-тесты. Их листинг приведён в приложении В.

Тест 1

Точки : }

Ответ : множество дваждытреугольное

Тест 2

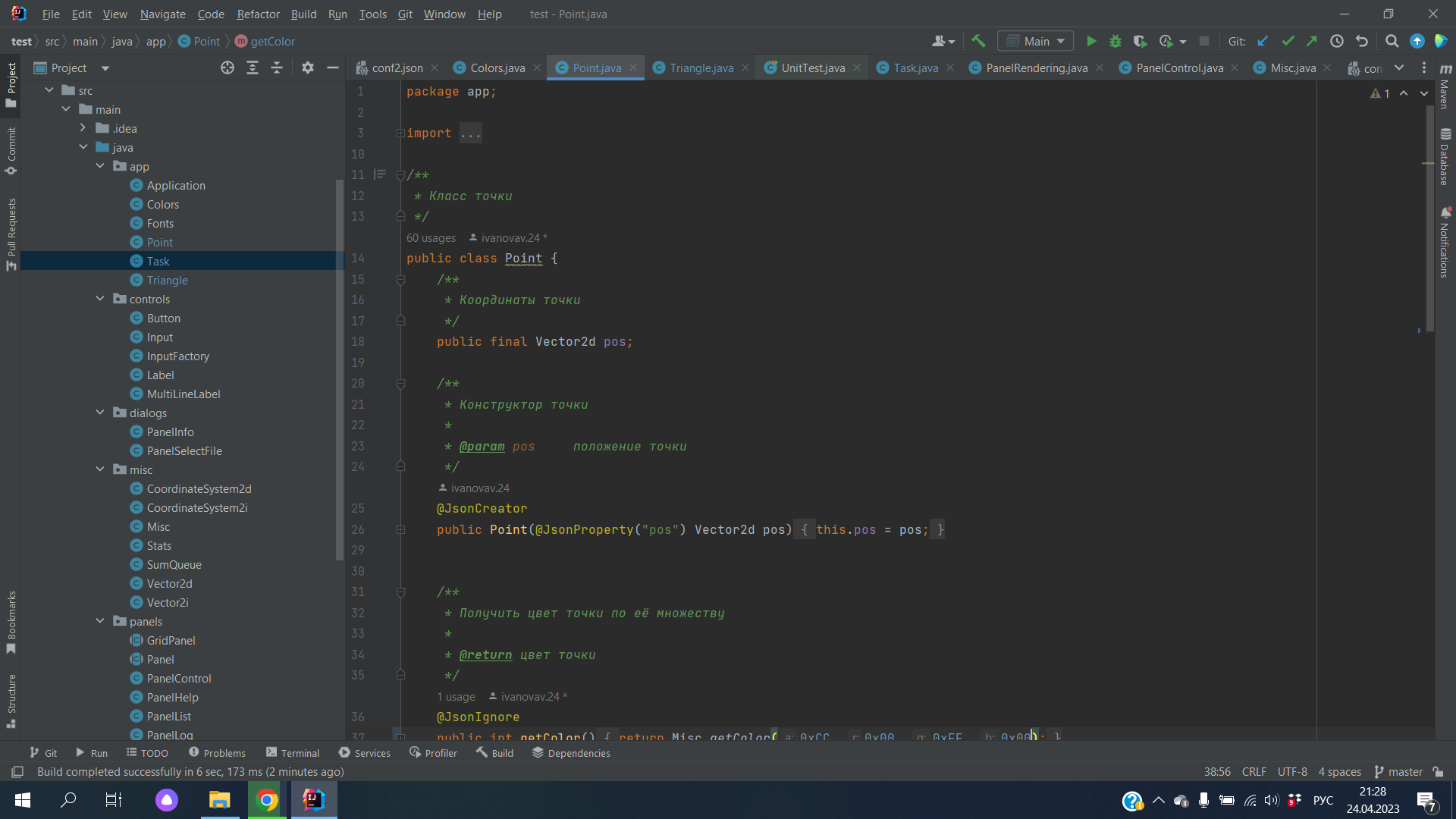
Точки : }

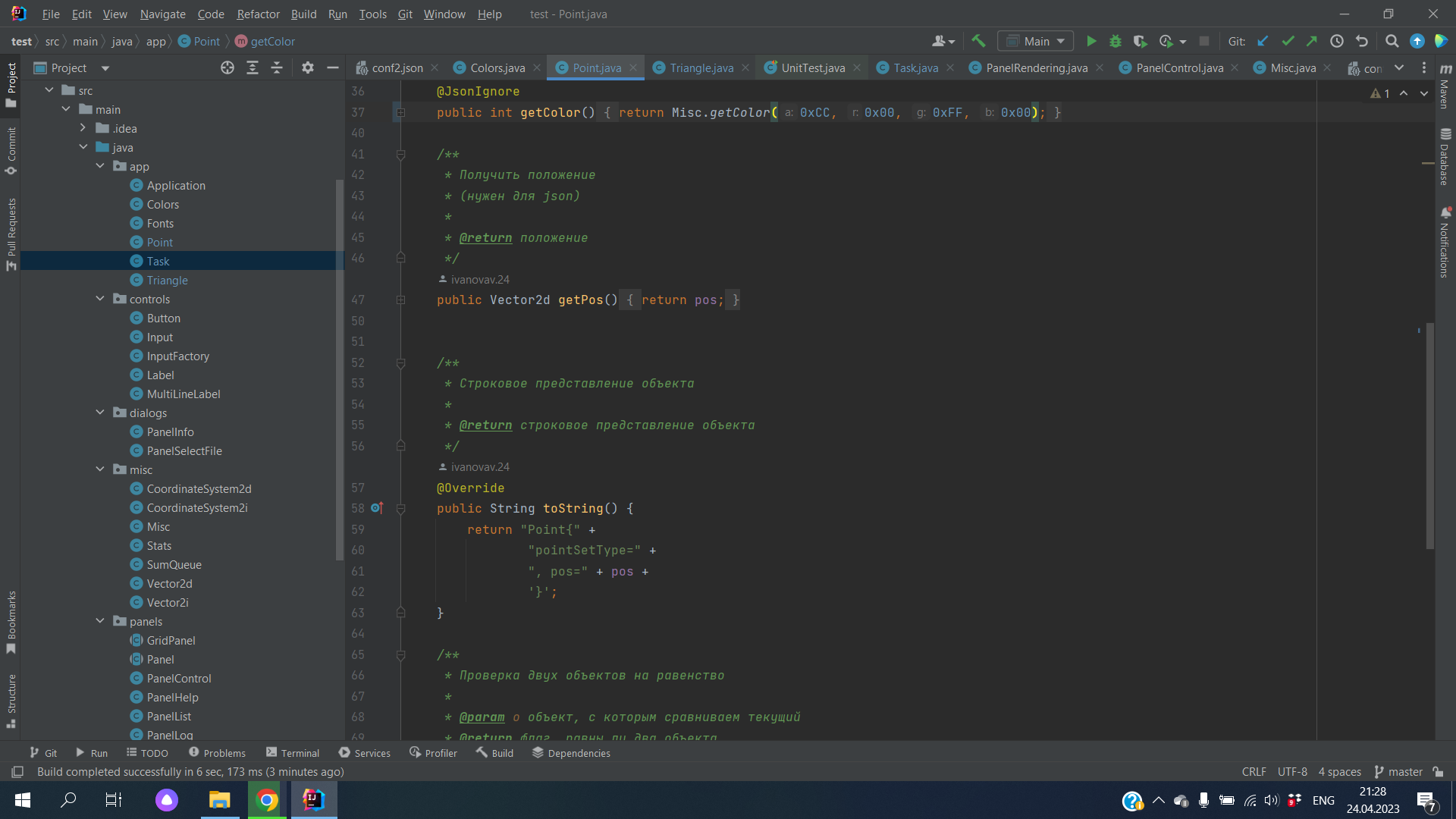
Ответ : множество недваждытреугольное

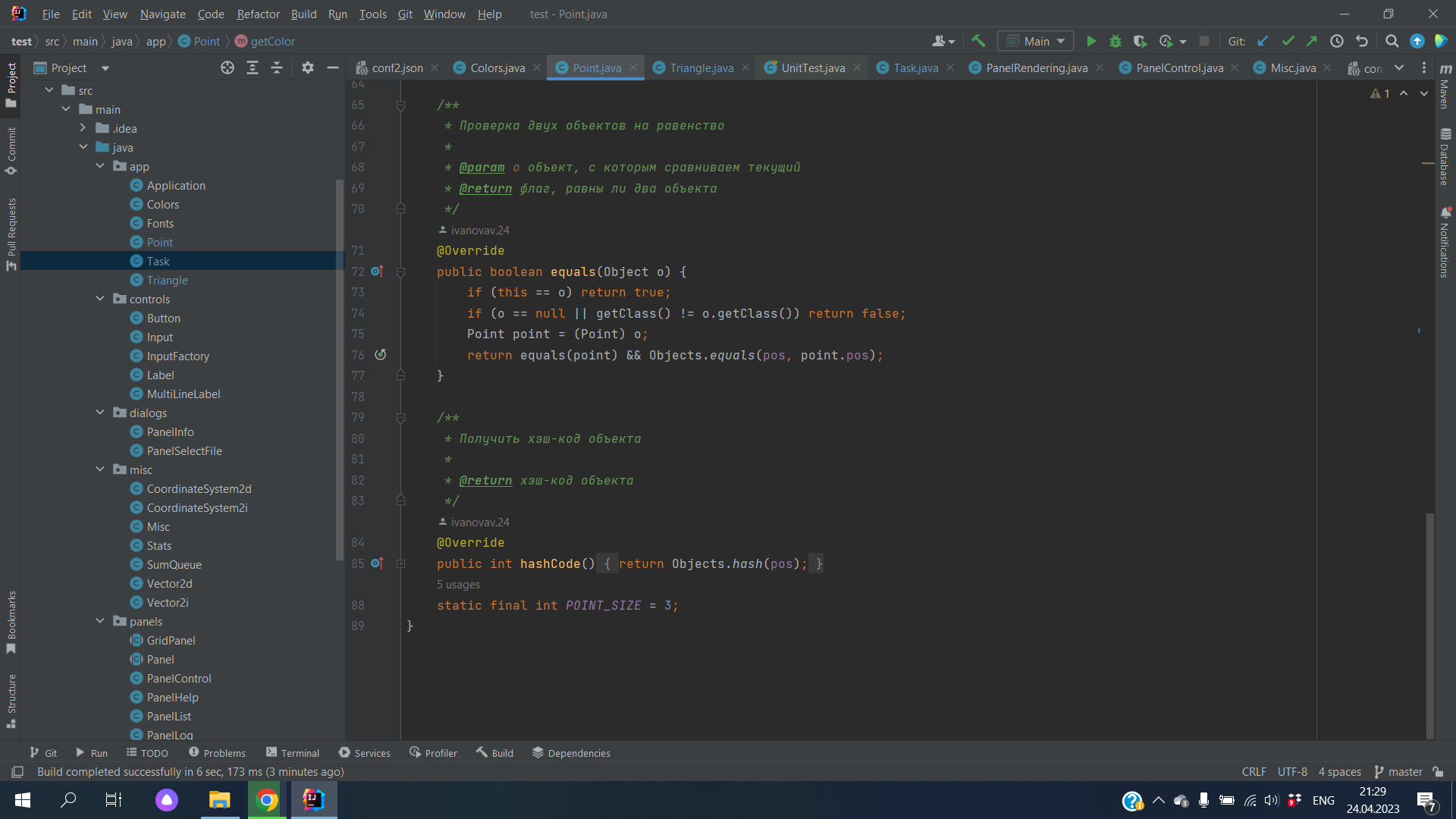
# 7. Заключение

В рамках выполнения поставленной задачи было создано графическое приложение с требуемым функционалом. Правильность решения задачи проверена с помощью юнит-тестов.

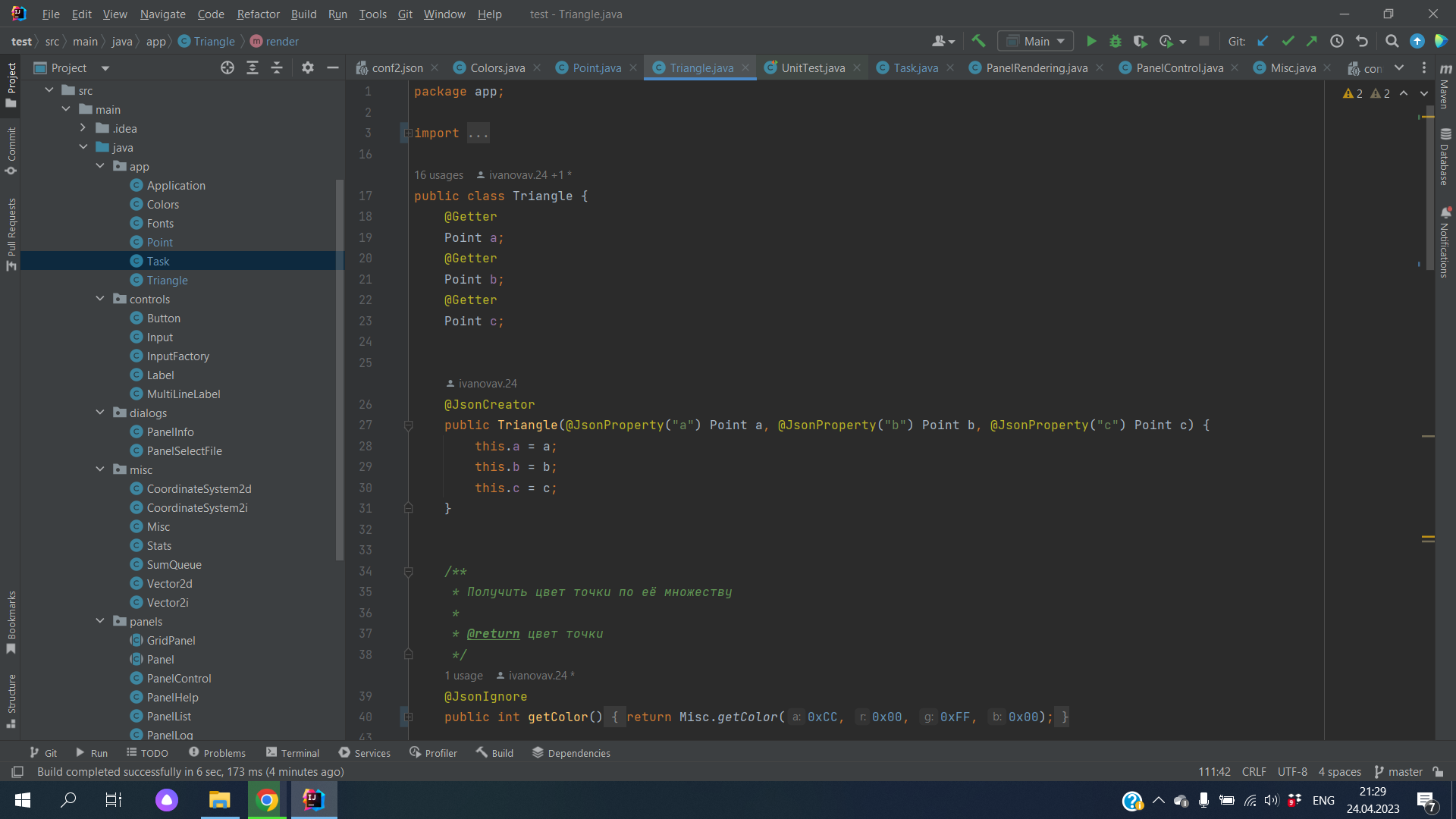
# Приложение А. Point.java

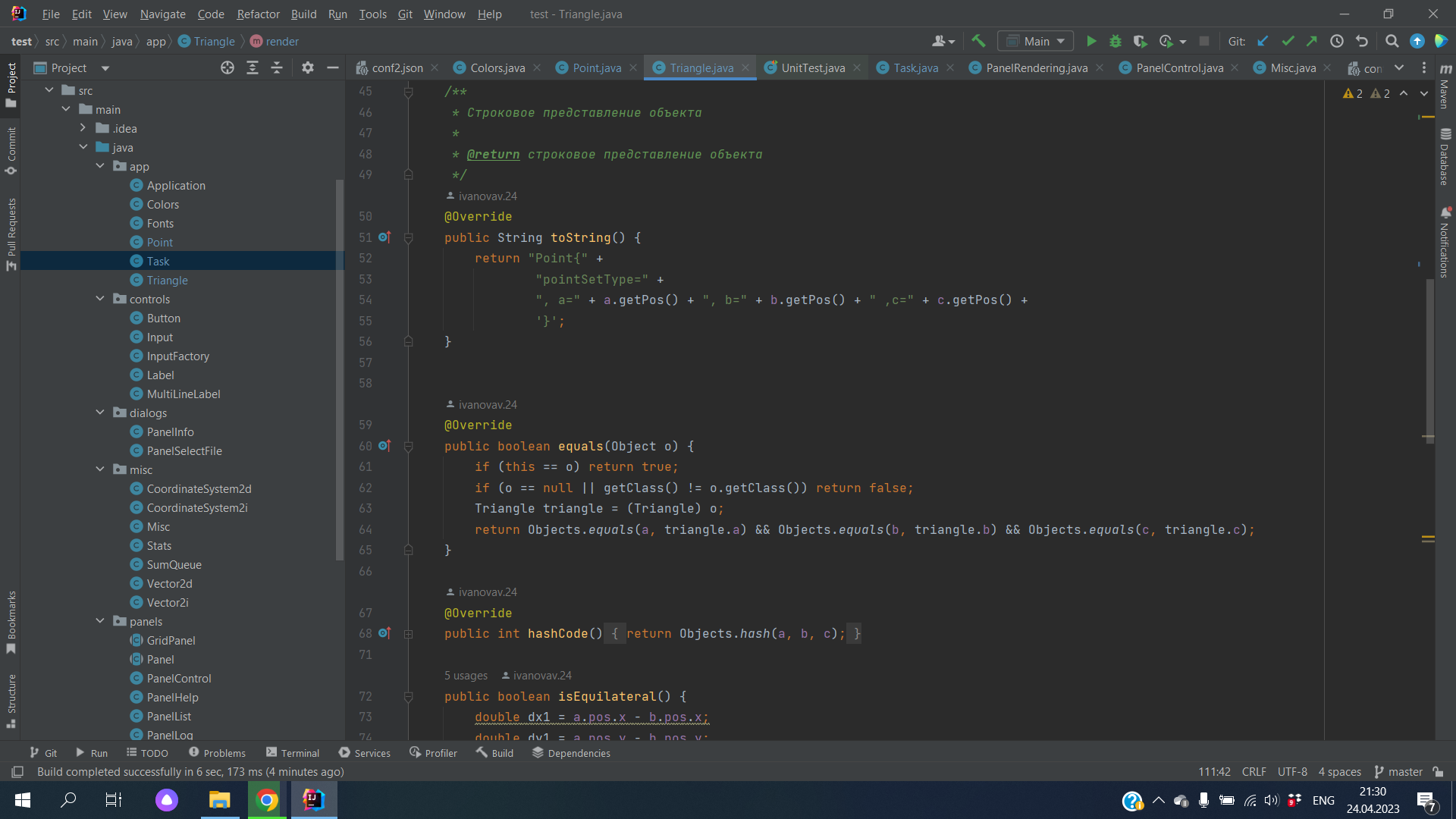






Приложение Б. Triangle.java







# Приложение В. UnitTest.java

