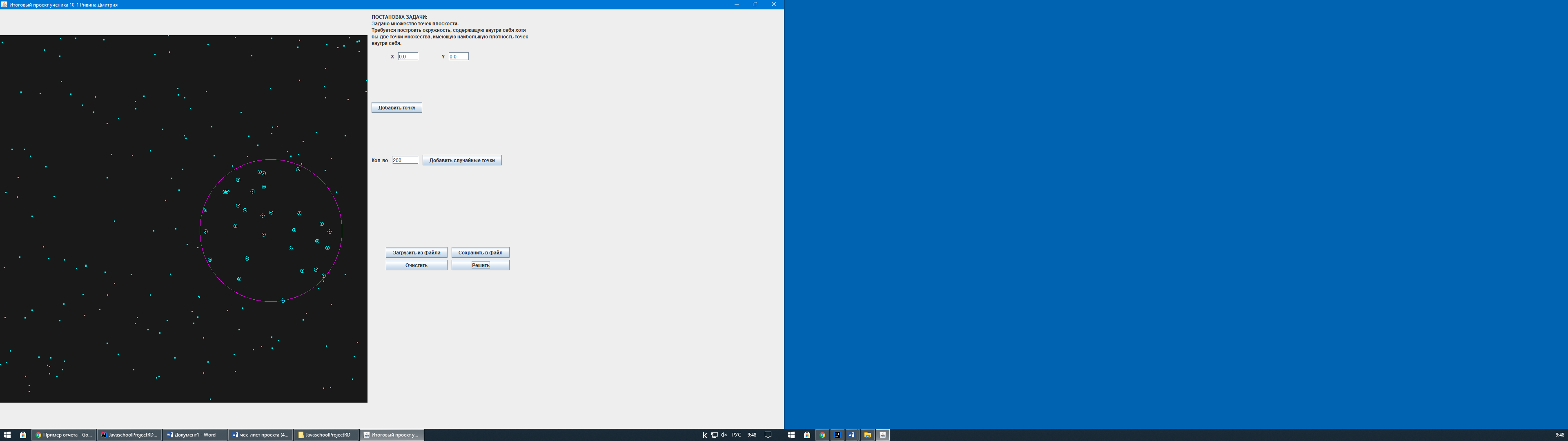
**Государственное бюджетное общеобразовательное учреждение "Президентский физико-математической лицей № 239"**

**Нахождение круга с наибольшей плотностью точек внутри.**



**Годовой проект по информатике.**

Ривин Дмитрий, 10-1 класс.

2021 год

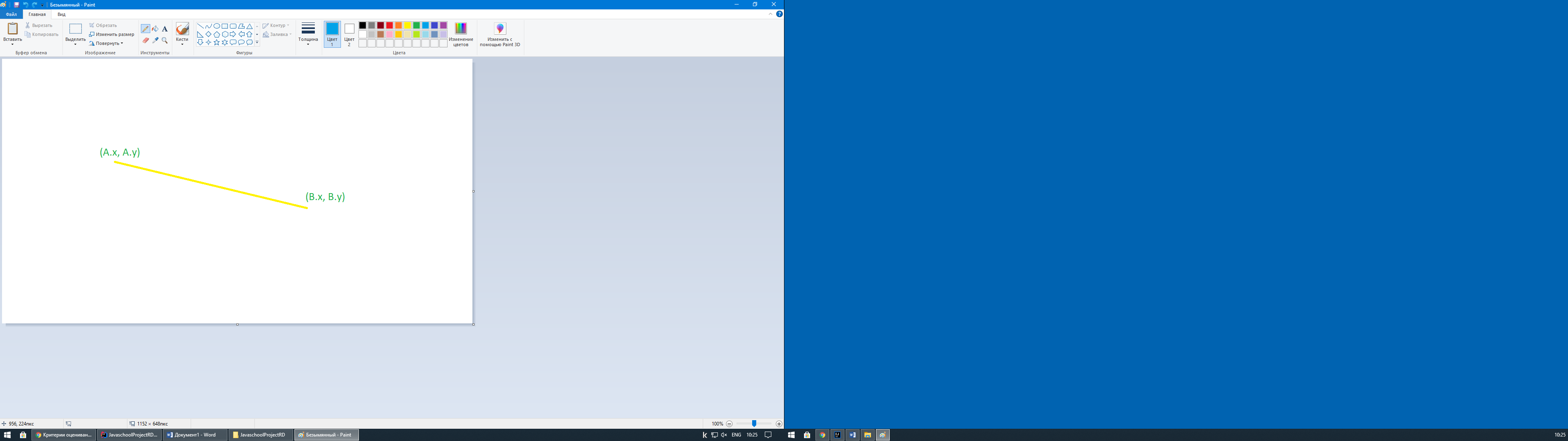
1. **Постановка задачи.** На плоскости задано множество точек. Найти окружность, содержащую внутри себя хотя бы две точки множества, имеющую наибольшую плотность точек внутри себя (количество точек на единицу площади). Нарисовать найденную окружность и выделить все точки, находящиеся внутри нее. Превратить их в кружочки.
2. **Уточнение исходных и выходных данных.** 
   1. **Исходные данные.**
   2. **Выходные данные.**

Необходимо нарисовать результирующую окружность и отметить, заключив их в маленькие круги, все точки, лежащие внутри.

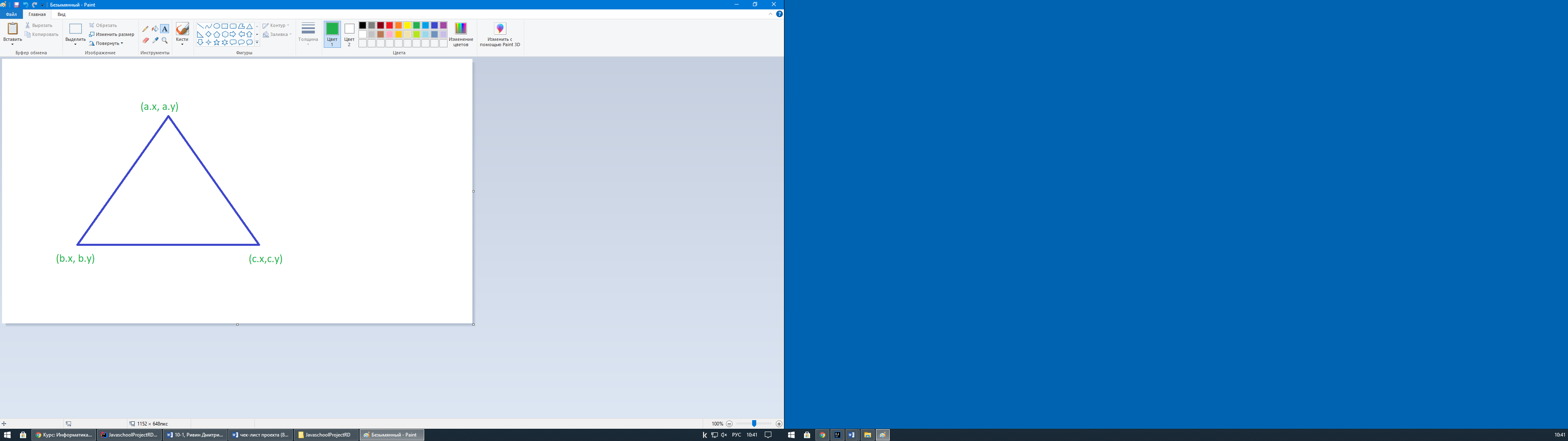
1. **Метод решения.**
   1. **Математическая модель.**

* Расстояние между двумя точками.

(A.x-B.x)\*(A.x-B.x)+(A.y-B.y)\*(A.y-B.y)



* Координаты точки на окружности.
* Когда точка лежит внутри окружности?
* Центр и радиус описанной окружности треугольник. Пусть даны координаты вершин a, b, c треугольника.



D = 2\*(a.x\*(b.y-c.y)+b.x\*(c.y-a.y)+c.x\*(a.y-b.y)), (вспомогательное обозначение)

Координаты центра: (((((triangle.a.x\*triangle.a.x)+(triangle.a.y\*triangle.a.y))\*(triangle.b.y-triangle.c.y)+((triangle.b.x\*triangle.b.x)+(triangle.b.y\*triangle.b.y))\*(triangle.c.y-triangle.a.y)+((triangle.c.x\*triangle.c.x)+(triangle.c.y\*triangle.c.y))\*(triangle.a.y-triangle.b.y))/D),

(((triangle.a.x\*triangle.a.x)+(triangle.a.y\*triangle.a.y))\*(triangle.c.x-triangle.b.x)+((triangle.b.x\*triangle.b.x)+(triangle.b.y\*triangle.b.y))\*(triangle.a.x-triangle.c.x)+((triangle.c.x\*triangle.c.x)+(triangle.c.y\*triangle.c.y))\*(triangle.b.x-triangle.a.x)/D));

* – площадь круга.
  1. **Используемая структура данных.**
* Входные данные: точки. Мы будем хранить их в динамическом массиве (points),
* Вспомогательные хранимые данные: динамический массив (circles), он возникает в процессе решения задачи
* Выходные данные: круг и точки, лежащие внутри него. Искомый круг хранится отдельно как объект класса Circle. Попавшие внутрь точки находятся и сразу выводятся в форму.

Входные точки

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| (x1, y1) | (x2, y2) | (x3, y3) | (x4, y4) | (x5, y5) | (x6, y6) | (x7, y7) | (x8, y8) |

с

Треугольник

|  |  |  |
| --- | --- | --- |
| A | B | C |
| (x2, y2) | (x6, y6) | (x8, y8) |

Окружность

Результирующая окружность

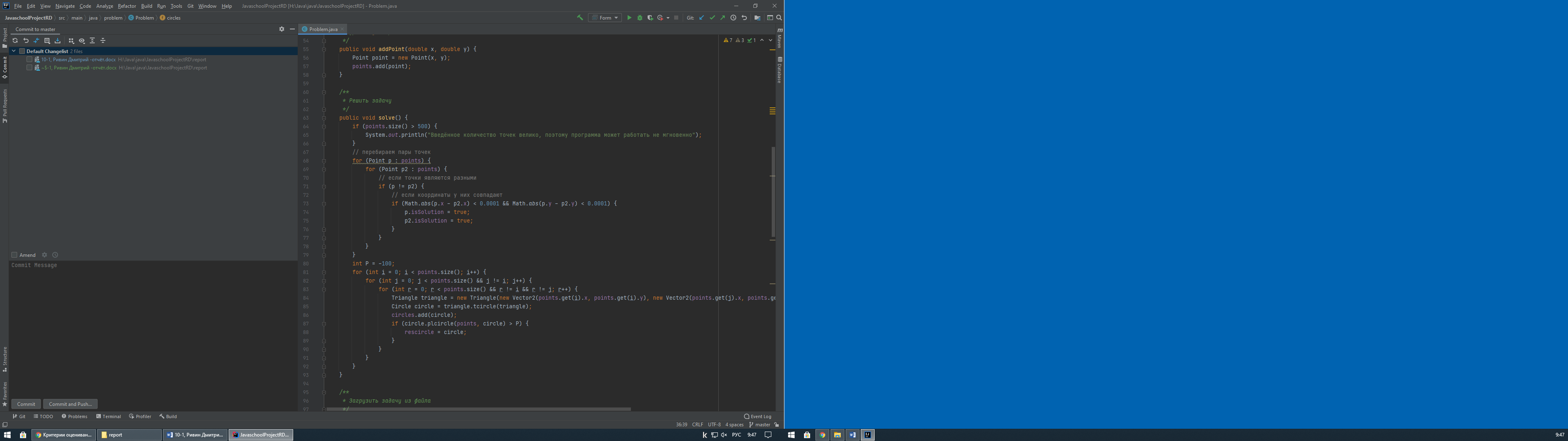
* 1. **Алгоритм решения.**

Переберём все треугольники и рассмотрим их описанные окружности, используя координаты. Напишем метод plcircle, который по данной окружности множества, лежащих внутри данной окружности. По всем рассмотренным окружностям посчитаем плотность точек для них. Мы перебрали вариантов, где n – количество точек. Выберем из них окружность с наибольшей плотностью. Она содержит внутри хотя бы 2 точки и является искомой. Нарисуем её цветом и нарисуем как маленькие цветные окружности точки, попавшие внутрь. Точки, попавшие на границу учтём и тоже покрасим. Организуем рисование в форме. (render)

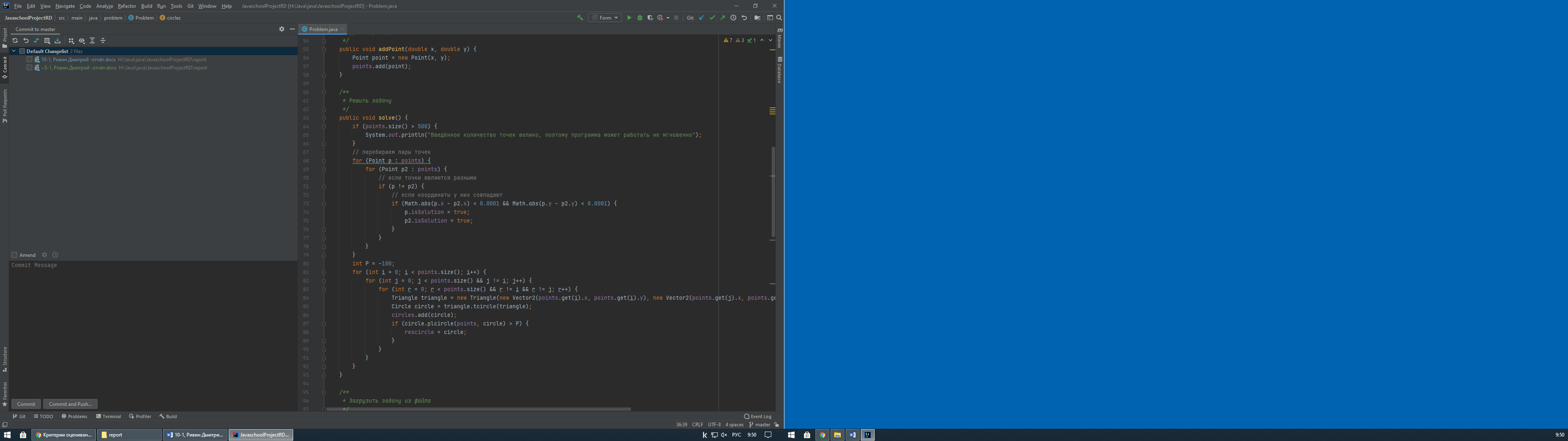
* 1. **Комментированный листинг.**

1. Метод solve: (решение)

Начало метода solve, перебор точек для их визуализации

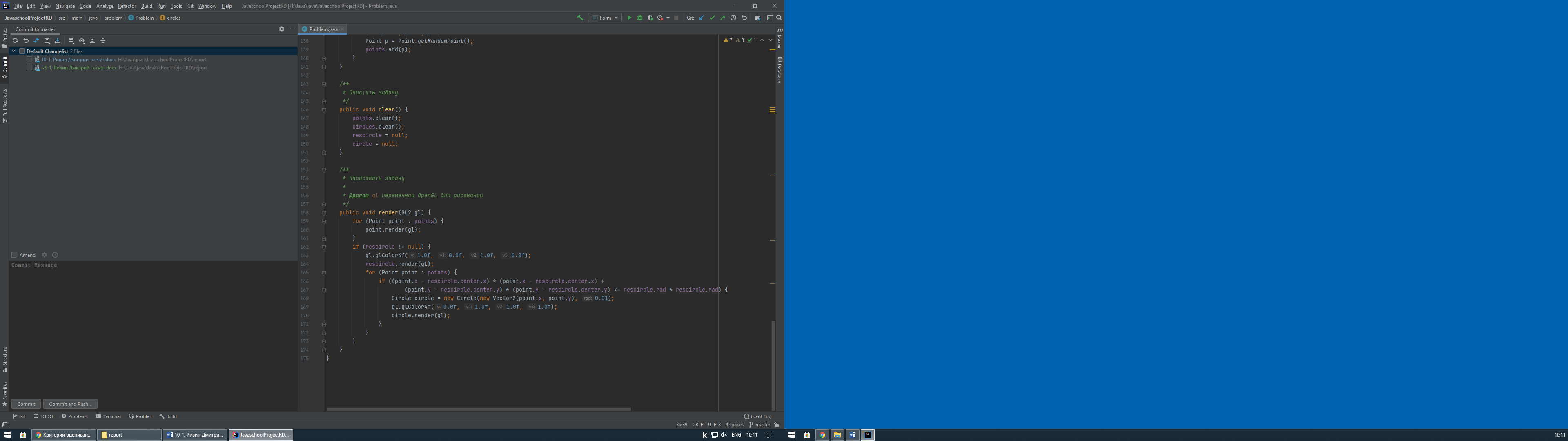


Основное действие метода, заключающееся в переборе треугольников и их описанных окружностей. Нахождение искомой с наибольше плотностью точек (P)



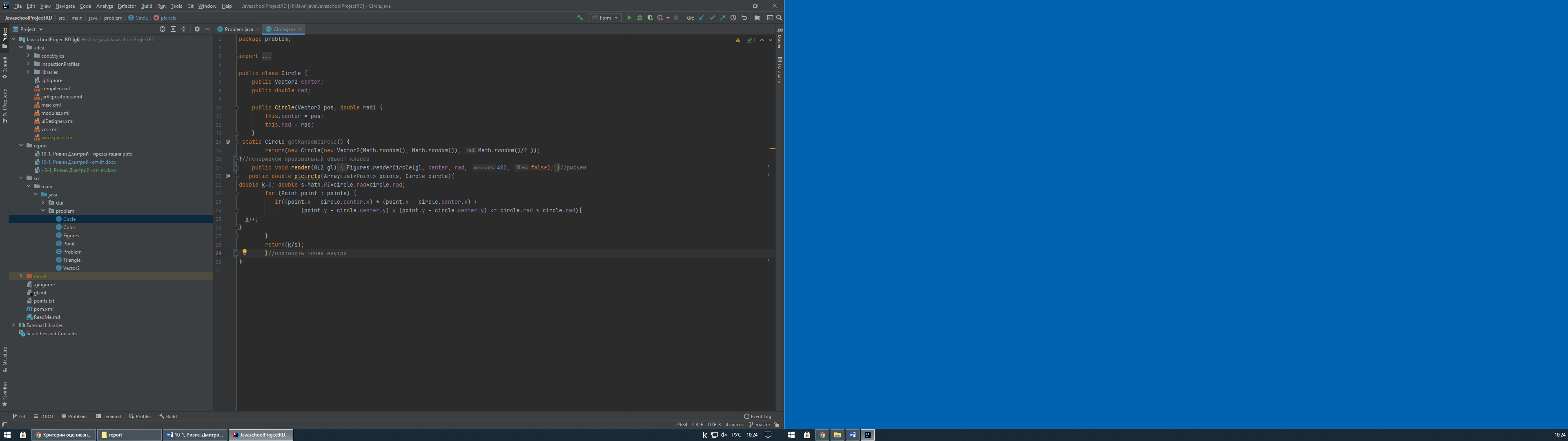
Рисование, попавшие внутрь окружности (rescircle) точки изображаем кружочками.

1. Метод render:

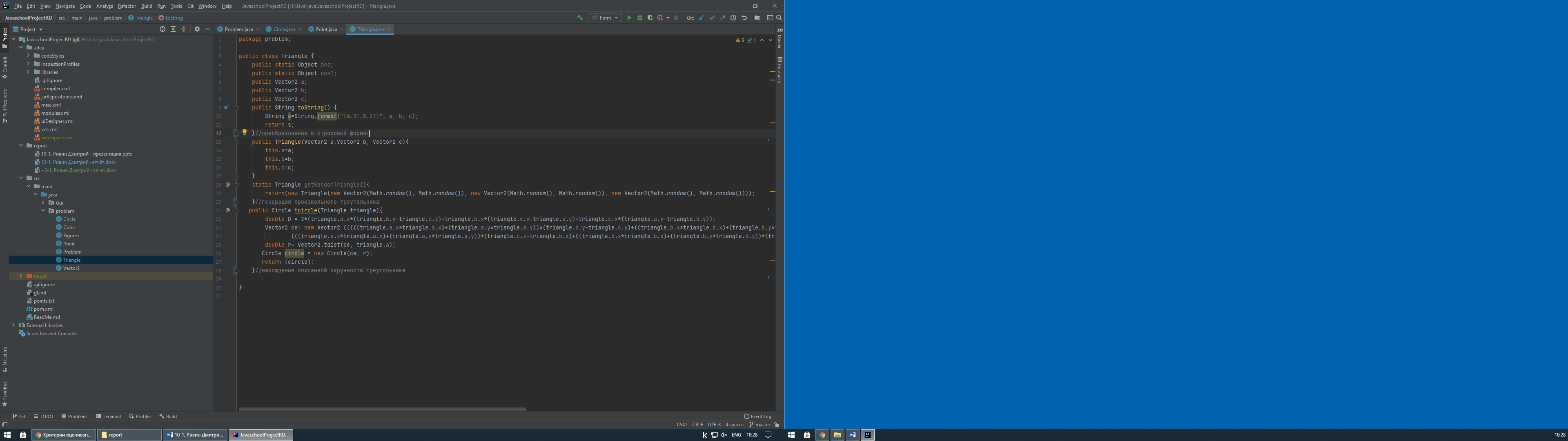


1. Классы:

* Circle



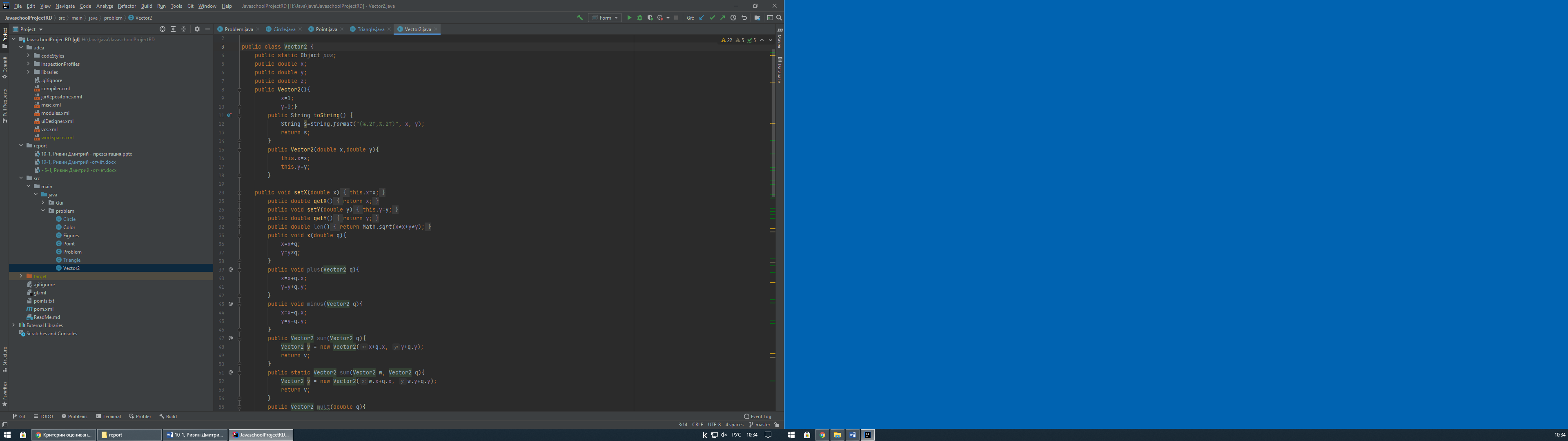
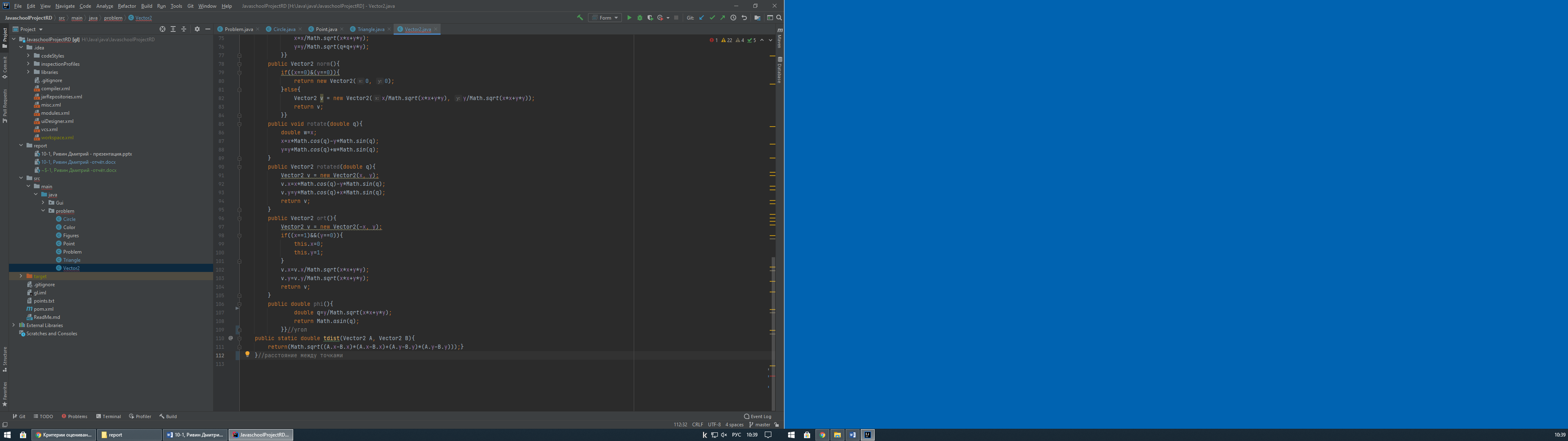
* Triangle



Переход к окружности

* Vector 2 (вектор)

Задание и операции

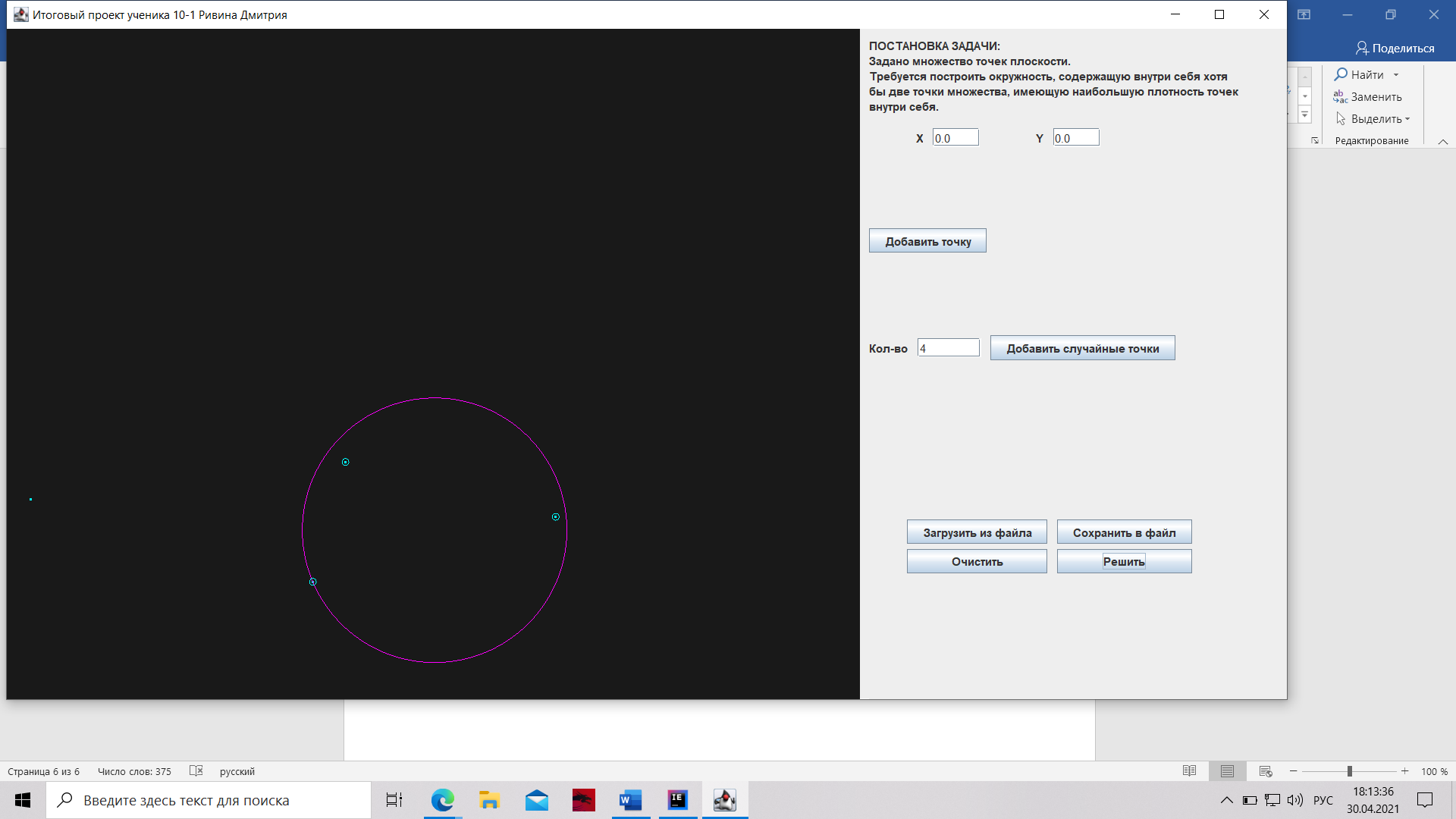
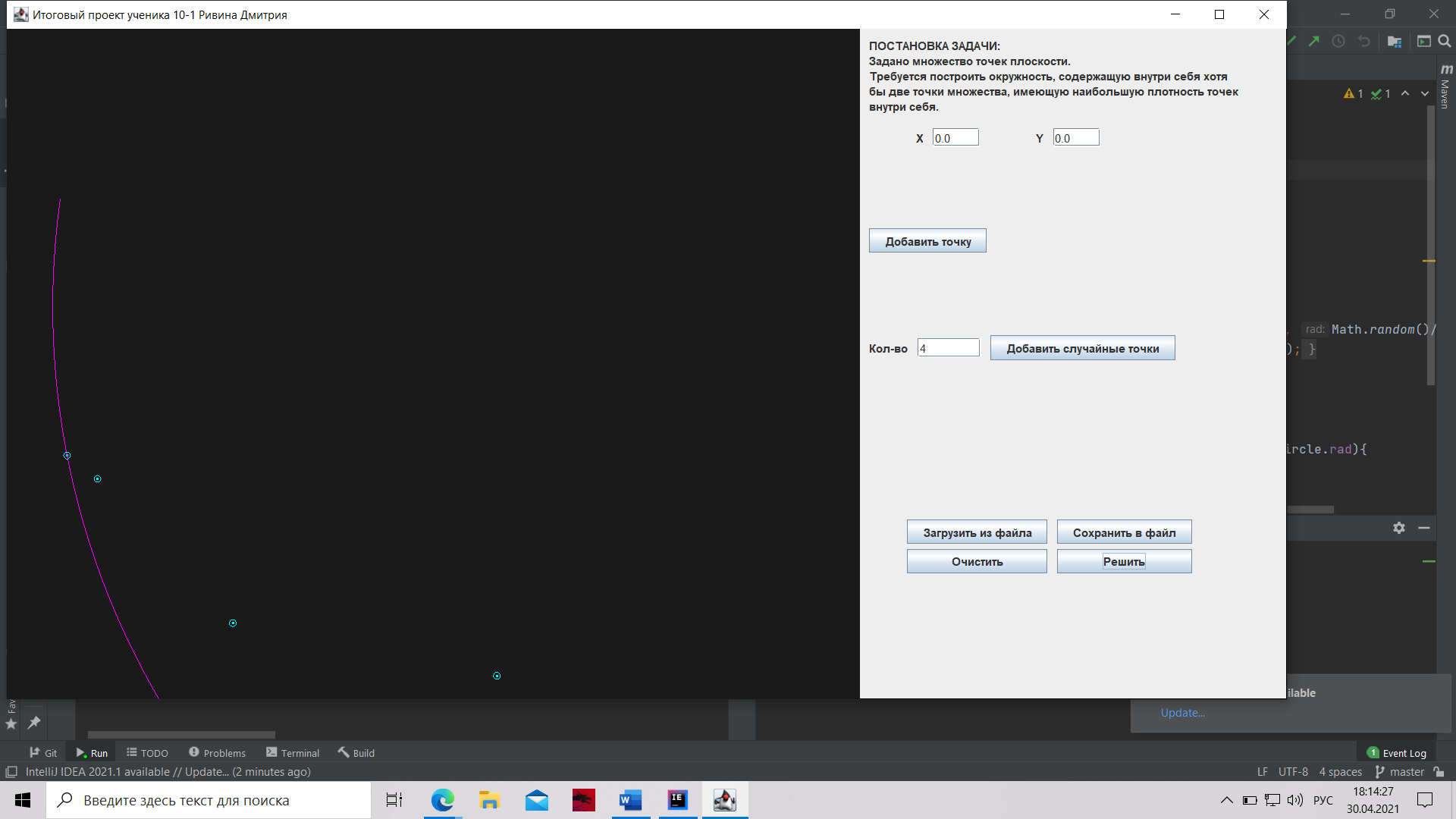
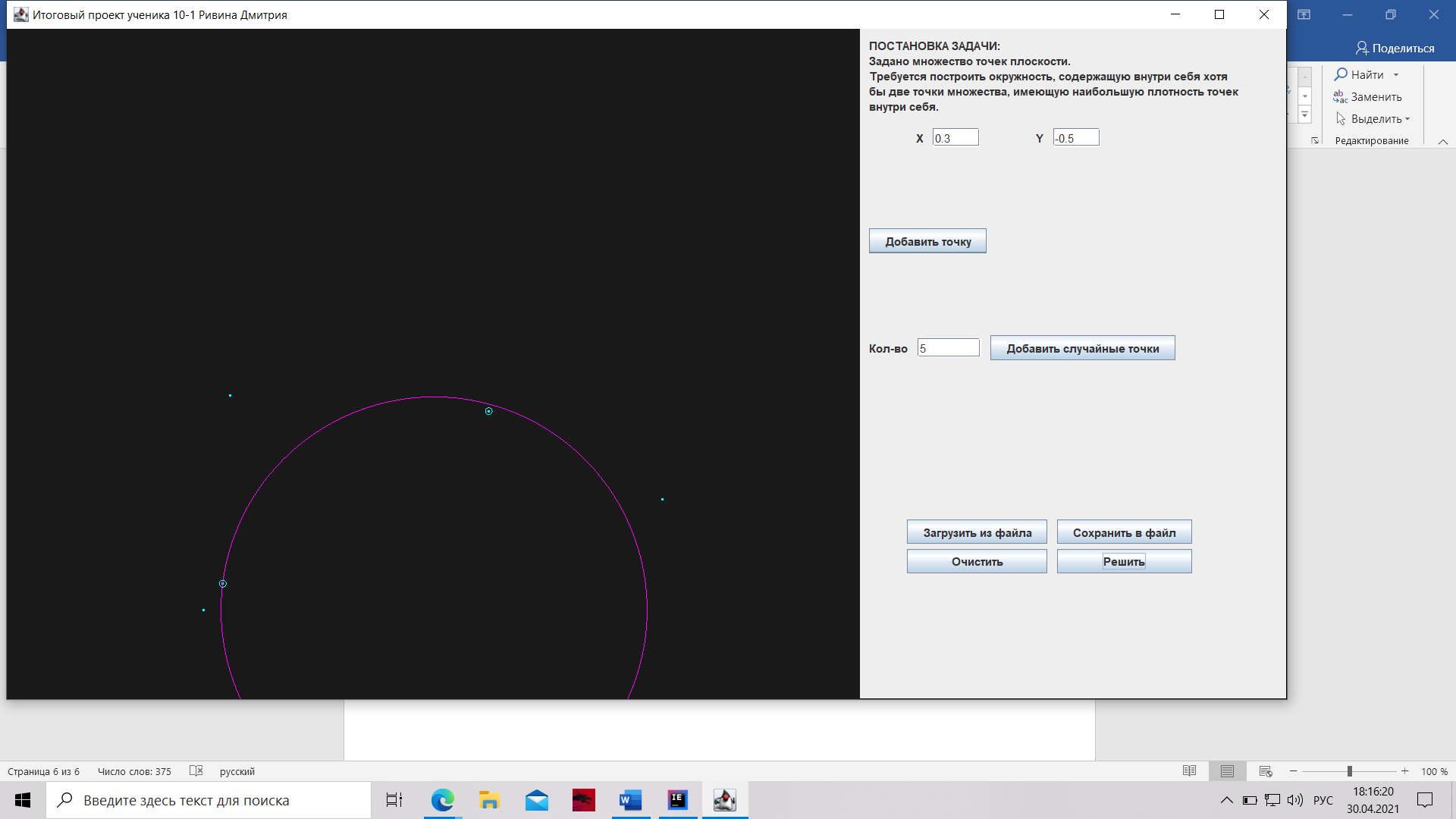
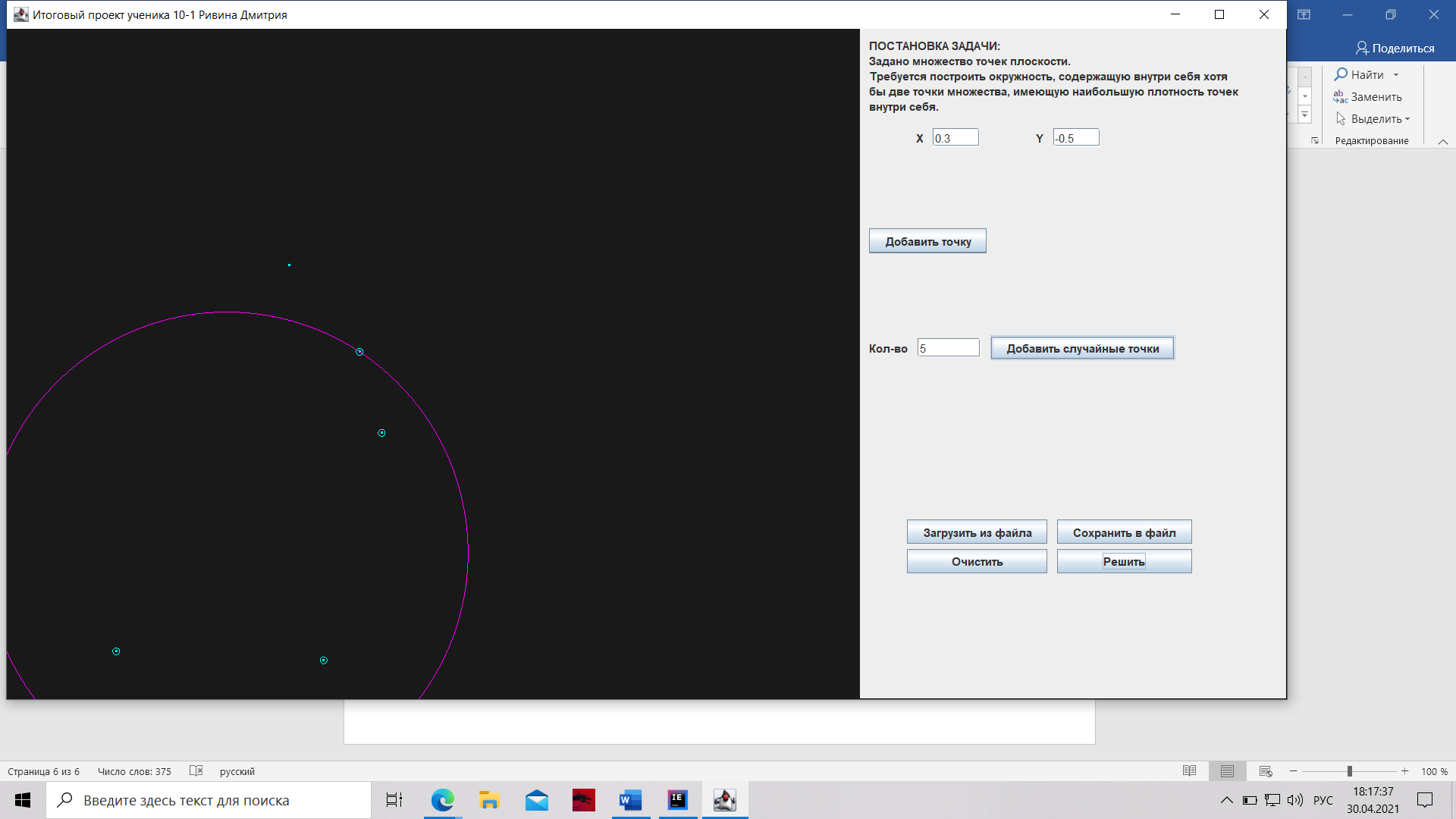
 

1. **Работа программы.**
   1. **Пример работы проекта.**

|  |  |
| --- | --- |
| **Входные данные**  Точки:  1. (0.1, 0.1),  2. (-0.1, -0.1),  3. (0.1, 0.2),  4. (0.2, 0.1). | **Результат**  Внутри 3 точки: (0.1, 0.1), (-0.1, 0.1) , (0.2, 0.1). |

* 1. **Анализ правильности решения.**

Проведём 4 теста: 2 для 4-x точек, 2 – для пяти.

* 
* 
* 
* 

На основании приведённого метода решения задачи и подтверждающих примеров работы делаем вывод о работоспособности программы.

***Спасибо за внимание.***