Муниципальное автономное общеобразовательное учреждение города Кургана «Гимназия № 30» (МАОУ «Гимназия № 30»)

640001, Российская Федерация, Курганская область, город Курган, улица Станционная, дом 26

тел.: (3522) 49-32-11 (3522) 49-28-46, (3522) 49-34-36 Email: [gimnazia30@mail.ru](mailto:gimnazia30@mail.ru) сайт: [gim30-r45.gosuslugi.ru](https://gim30-r45.gosuslugi.ru/)

Проектная работа «Создание приложения для построения геометрических конструкций на Python»

по геометрии

Выполнил ученик 10 «А» класса Русанов Евгений Васильевич

Руководитель Волосникова Ольга Николаевна, учитель математики

Курган, 2024 год

**Оглавление**

[**Введение** 3](#_Toc168441526)

[**ГЛАВА 1. ОСНОВЫ МЕТОДА КОМПЛЕКСНЫХ ЧИСЕЛ** 5](#_Toc168441527)

[**1.1.** **Геометрическая интерпретация комплексных чисел** 5](#_Toc168441528)

[**1.2.** **Операция перехода к сопряжённому числу.** 6](#_Toc168441529)

[**1.3.** **Векторная интерпретация комплексных чисел, их сложения и вычитания.** 6](#_Toc168441530)

[**1.4.** **Геометрический смысл умножения комплексных чисел.** 6](#_Toc168441531)

[**1.5.** **Параллельность и перпендикулярность векторов.** 7](#_Toc168441532)

[**1.6.** **Приведенное уравнение прямой и окружности** 8](#_Toc168441533)

[**1.7.** **Вывод необходимых формул для написания программы** 8](#_Toc168441534)

[**ГЛАВА 2. Создание приложения для построения геометрических конструкций** 12](#_Toc168441535)

[**2.1.** **Выбор библиотеки/фреймворка для создания графического интерфейса приложения** 12](#_Toc168441536)

[**2.2.** **Описание основных классов для работы приложения.** 12](#_Toc168441537)

[**2.3.** **Описание структуры работы созданного приложения** 13](#_Toc168441538)

[**2.4.** **Описание возможностей созданного приложения** 14](#_Toc168441539)

[**2.5.** **Пример работы приложения** 15](#_Toc168441540)

[**Заключение** 17](#_Toc168441541)

[**Список литературы** 18](#_Toc168441542)

# **Введение**

Геометрические конструкции являются неотъемлемой частью многих научных, инженерных и художественных дисциплин. Построение этих конструкций вручную может быть трудоемким и утомительным процессом, а также использование классических инструментов для построения приводит к достаточно большим погрешностям, особенно для сложных фигур. Поэтому для достижения высокого качества чертежей необходимо использовать специализированные программы.

**Цель проекта:** cоздать программу с графическим интерфейсом для построения и анализа геометрических конструкций.

Для достижения данной цели были поставлены следующие **задачи**:

1. Изучить теорию по теме “счет в комплексных координатах для решения геометрических задач”.
2. Вывести формулы и зависимости, необходимые для программного вычисления объектов и их взаимодействий.
3. Выбрать необходимые библиотеки и фреймворки для написания программы.
4. Разработать интерактивный графический интерфейс для создания и редактирования геометрических конструкций.
5. Создать приложение, используя выбранные библиотеки и имеющиеся наработки.

В работе над проектом были использованы следующие **методы**: анализ, моделирование, формализация, изучение, сравнение.

**Актуальность** проекта заключается в следующем: Во время школьного обучения приходится решать большое количество геометрических задач и также запоминать огромное количество теории, а создание подобного приложения помогло бы визуализировать пройденный материал и помогло бы в построение правильных геометрических чертежей. Также немало важным является то, что данное приложение будет иметь открытый исходный код, благодаря чему его можно будет использовать в дальнейших проектах.

**Теоретическая значимость** заключается в систематизации имеющейся информации по теме проекта.

**Практическая значимость** заключается в упрощении построения геометрических чертежей, а также ознакомление учеников с темой комплексная плоскость.

# **ГЛАВА 1. ОСНОВЫ МЕТОДА КОМПЛЕКСНЫХ ЧИСЕЛ**

## **1.1. Геометрическая интерпретация комплексных чисел**

**•** Плоскость комплексных чисел. Зададим на плоскости прямоугольную декартову систему координат Оху. Тогда каждому комплексному числу z, представленному в алгебраической форме (х, у - действительные числа; = — 1), можно однозначно поставить в соответствие точку M плоскости с координатами (х, у): . Комплексное число z называют комплексной координатой соответствующей точки M и пишут: M(z).

Следовательно, множество точек евклидовой плоскости находится во взаимно однозначном соответствии с множеством комплексных чисел. Эту плоскость называют плоскостью комплексных чисел. Начало О системы координат называют при этом начальной или нулевой точкой плоскости комплексных чисел.

При у = 0 число z — действительное. Действительные числа изображаются точками оси Ох, поэтому она называется действительной осью. При х = 0 число z — чисто мнимое: z = iy. Чисто мнимые числа изображаются точками на оси Oy, поэтому она называется мнимой осью. Нуль — одновременно действительное и чисто мнимое число.

**•** Расстояние от начальной точки О до точки M(z) называется модулем комплексного числа z и обозначается символом |z| или г:

Если — ориентированный угол, образованный вектором ОМ с осью Ох, то по определению синуса и косинуса

Такое представление комплексного числа называется его тригонометрической формой. Угол называется аргументом комплексного числа и обозначается .

## **1.2. Операция перехода к сопряжённому числу.**

**•** Если дано комплексное число , то комплексное число

называется комплексно сопряжённым (или просто сопряжённым) этому числу z. И обозначается .

**•** Из равенства следует y = 0 и обратно. Это значит, что числа, равные своим сопряжённым, являются действительными. Условие является критерием принадлежности числа к чисто мнимым.

Число, сопряжённое с суммой, произведением или же частным комплексных чисел, есть соответственно сумма, произведение или частное чисел, сопряжённых с данными комплексными числами. Эти равенства можно легко проверить, пользуясь формулами для операций над комплексными числами в алгебраической форме.

## **1.3. Векторная интерпретация комплексных чисел, их сложения и вычитания.**

**•**  Каждой точке M(z) плоскости комплексных чисел взаимно однозначно соответствует вектор ОМ с началом в нулевой точке О. Поскольку сложение и вычитание векторов, заданных своими координатами, выполняются по тем же формулам, что сложение и вычитание соответствующих им комплексных чисел, то сложению и вычитанию комплексных чисел, записанных в алгебраической форме, однозначно отвечает сложение и вычитание соответствующих им векторов.

## **1.4. Геометрический смысл умножения комплексных чисел.**

**•** Умножение двух комплексных чисел

полняется по формуле:

. Геометрически это означает, что точка С(ab) является образом точки A(a) при поворотной гомотетии в точке O(0) с коэффициентом и направленным углом , аналогичное можно сказать про образ точки B(b).

## **1.5. Параллельность и перпендикулярность векторов.**

**•** На плоскости комплексных чисел даны ненулевые векторы . Векторы сонаправлены тогда и только тогда, когда

, т.е. при . Очевидно также, что эти векторы противоположно направлены в том и только в том случае, если . Но комплексные числа с аргументами 0, являются действительными. Итак, для того, чтобы векторы были коллинеарны, необходимо и достаточно, чтобы частное было действительным числом. Значит, равенство есть критерий коллинеарности векторов.

**•** На плоскости комплексных чисел даны ненулевые векторы . Векторы перпендикулярны тогда и только тогда, когда

. Но комплексные числа с аргументами являются чисто мнимыми. Итак, для того, чтобы векторы были перпендикулярны, необходимо и достаточно, чтобы частное было чисто мнимым числом. Значит, равенство есть критерий перпендикулярности векторов.

## **1.6. Приведенное уравнение прямой и окружности**

**•** Выведем в комплексных координатах приведенное уравнение прямой, проходящей через заданные две точки A(a) и B(b) с комплексными координаты.

Из 1.5. следует, что уравнение равносильно тому, что точка

Z(z) AB. С помощью математических операций приведем данное уравнение к необходимому виду:

Таким образом мы получили приведенное уравнение прямой -

(В данном случае u = , а c = )

Также из расчетов видно, что u является осью нормали к прямой AB.

**•** Выведем в комплексных координатах приведенное уравнение окружности с заданным центром P(p) и радиусом r (

Из 1.1. следует, что уравнение равносильно тому, что точка Z(z) принадлежит искомой окружности. Это уравнение и называется приведенным уравнением окружности.

## **1.7. Вывод необходимых формул для написания программы**

**•** Выведем в комплексных координатах уравнение точки, являющейся пересечением двух заданных прямых и

.

Для этой точки z должна выполняться система уравнений:

(вычитаем из первого второе)

-) + (

, но при деление невозможно, но из 1.5. и 1.6. следует, что это равенство равносильно тому, что заданные две прямые параллельны. В итоге получаем:

**•** Выведем в комплексных координатах уравнение точки, являющейся пересечением заданной прямой и окружности и

.

Для этой точки z должна выполняться система уравнений:

Сделаем замену :

(видно, что

Возможно были получены мнимые точки пересечения (если объекты не пересекаются), для этого одну из точек нужно подставить во второе уравнение

**•** Выведем в комплексных координатах уравнение точки, являющейся пересечением заданных окружностей и

.

Для этой точки z должна выполняться система уравнений:

Сделаем замену и p =

(видно, что

Возможно были получены мнимые точки пересечения (если объекты не пересекаются), для этого одну из точек нужно подставить во второе уравнение

**•** Выведем в комплексных координатах уравнение точки, лежащей на заданной прямой .

Раз точка z ближайшая к А на заданной прямой, значит AZ перпендикулярна заданной прямой и параллельна . Для этой точки z должна выполняться система уравнений:

(где

*-*

**•** Выведем в комплексных координатах уравнение расстояния от заданной точки A(a) до заданной прямой .

Воспользуемся расчетами выше и получим, что z ближайшая к А точка на прямой, значит расстояние

**•** Все остальные формулы, которые используются в программе, считаются с помощью композиций результатов полученных выше (делается это не вручную)

# **ГЛАВА 2. Создание приложения для построения геометрических конструкций**

## **2.1. Выбор библиотеки/фреймворка для создания графического интерфейса приложения**

Для создания GUI (Graphics User Interface, графический интерфейс пользователя) приложения требуется выбрать некоторую уже созданную библиотеку либо фреймворк, предназначенную для создания GUI.

Для данного проекта была выбрана библиотека Pygame, поскольку в ней реализованы необходимые функции для отрисовки приложения, а также реализована простая работа с реактивным программированием (существует возможность легко выражать статические и динамические потоки данных), т.е. легко производить изменения в зависимости от действий пользователя.

## **2.2. Описание основных классов для работы приложения.**

Всего в приложении представлено 6 классов:

* Class Flatness – класс координатной плоскости или класс видимого пространства. Во время работы приложения, пользователь может перемещать, приближать и отдалять видимую область комплексной плоскости, и этот класс необходим для понимая, что именно показывать на экране программы.
* Class Point, Line, Circle – классы объектов, которые хранят все свойства данного объекта для дальнейших операций с ним (его координаты, цвет, тип отрисовки и все зависимости от других объектов), а также имеют функцию отрисовки и изменения своих свойств (в зависимости от действий пользователя координаты объектов, их цвет и т.д. могут меняться)
* Class Button – класс кнопки, который отвечает за ее отрисовку и за взаимодействие с пользователем (т.е. при наведение на кнопку это как-то отображается, а при нажатие производятся действия указанные при ее создание)
* Class List\_button – класс необходимый для автоматических расчетов координат кнопок, а также группировки их по тем или иным свойствам

## **2.3. Описание структуры работы созданного приложения**

Работу приложения можно разбить на три блока:

1. Взаимодействие с пользователем

На экране приложения представлено несколько возможностей для взаимодействия с программой, выбор одной из возможностей реализован с помощью кнопок (Сами возможности описаны ниже). После того как пользователь совершил какое-то действие, функции, отвечающие за работу с ним, считывают полученные данные и форматируют в нужную форму для передачи на следующий этап.

1. Проведение необходимых изменений и расчетов

После того как отформатированные данные попали на этот этап начинает совершаться расчет координат и зависимостей новых объектов, а также происходит перерасчет старых объектов (т.е. при изменение координаты одного объекта меняются все координаты объектов которые косвенно или непосредственно от него зависели). После всех действий на выходе мы получаем комплексную плоскость с данными о всех объектах, которые на ней находятся. Также происходит изменение характеристик объектов участвующих в дизайне приложения.

1. Отрисовка программы

На этом этапе происходит отрисовка видимой части плоскости и всех объектов участвующих в дизайне и работе приложения.

## **2.4. Описание возможностей созданного приложения**

1. Пользователь может ставить точки трех типов:

* Точка которая не будет зависеть от объектов созданных до нее
* Точка которая будет лежать на каком-то объекте
* Точка которая будет являться пересечением каких-то объектов

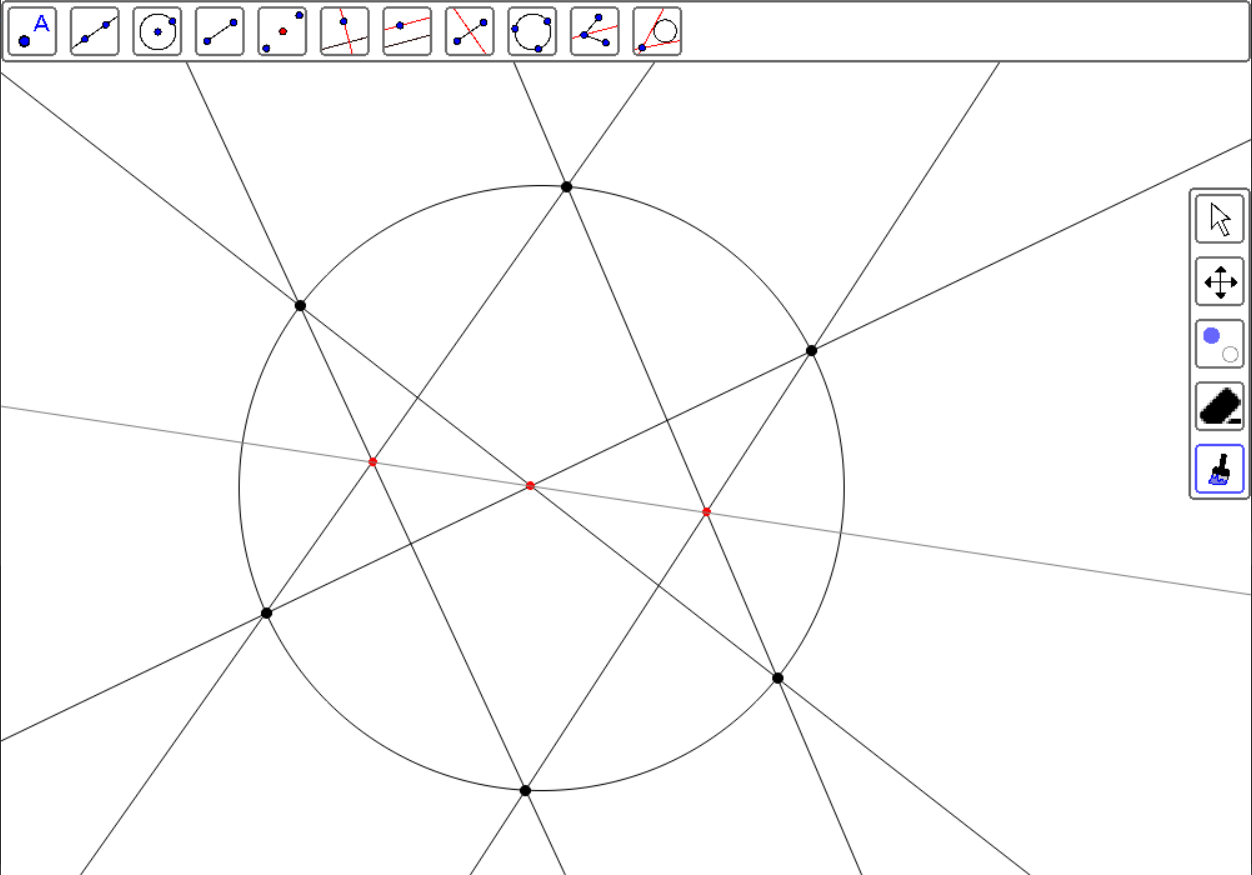
Также пользователь может выбрать уже существующею точку

(Программа сама после нажатия определит наиболее вероятное действие, которое пользователь хотел совершить).

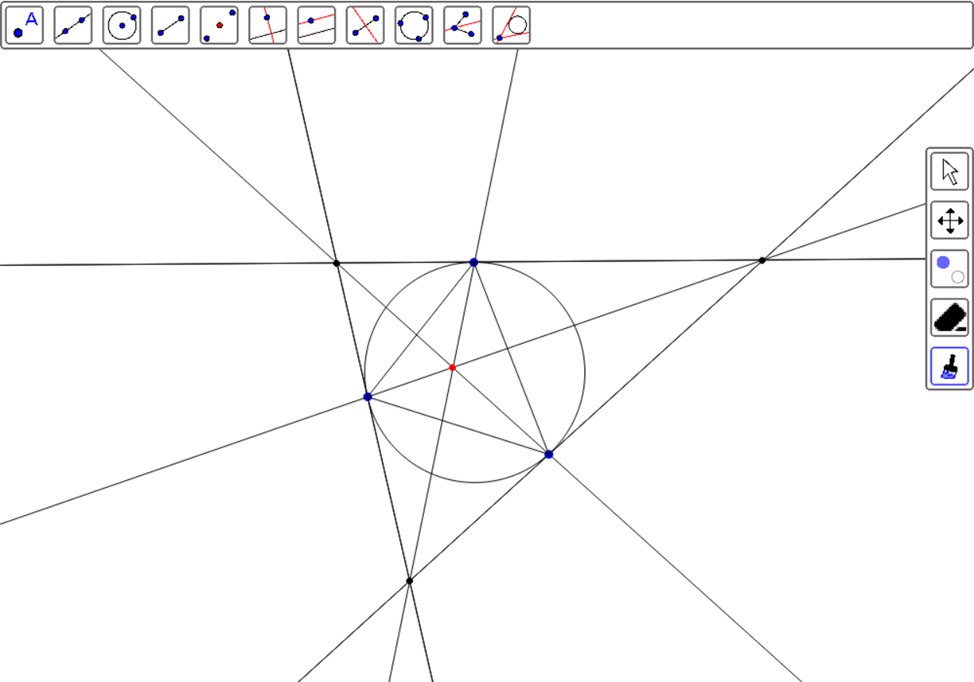
1. Пользователь может провести прямую или отрезок, указав две точки через которые объект должен проходить (если пользователь указывает в место, где нет существующих точек, то программа сама создаст ее).
2. Пользователь может провести окружность, указав ее центр и точку, через которую должна проходить окружность (если пользователь указывает в место, где нет существующих точек, то программа сама создаст ее).
3. Пользователь может указать отрезок, и программа построит точку, которая является ее серединой. (если пользователь указывает в место, где нет существующих точек, то программа сама создаст ее).
4. Пользователь может указать точку и прямую, и программа построит прямую проходящую через заданную точку и перпендикулярную заданной прямой. (если пользователь указывает в место, где нет существующих точек, то программа сама создаст ее).
5. Пользователь может указать точку и прямую, и программа построит прямую проходящую через заданную точку и параллельную заданной прямой. (если пользователь указывает в место, где нет существующих точек, то программа сама создаст ее).
6. Пользователь может указать отрезок, и программа построит прямую, являющиеся серединным перпендикуляром. (если пользователь указывает в место, где нет существующих точек, то программа сама создаст ее).
7. Пользователь может указать три точки, и программа построит окружность, проходящую через них. (если пользователь указывает в место, где нет существующих точек, то программа сама создаст ее).
8. Пользователь может указать угол, и программа построит прямую, являющиеся биссектрисой данного угла. (если пользователь указывает в место, где нет существующих точек, то программа сама создаст ее).
9. Пользователь может указать точку и окружность, и программа проведет касательные из этой точки к окружности. (если пользователь указывает в место, где нет существующих точек, то программа сама создаст ее).
10. Пользователь может перемещать уже существующие объекты, при этом будут сохраняться все зависимости созданные ранее.
11. Пользователь может перемещать, приближать и отдалять видимую часть комплексной плоскости.
12. Пользователь может удалять не нужные объекты, при этом будут удалены все объекты построение которых косвенно или непосредственно зависело от удаленного объекта.
13. Пользователь может делать некоторые объекты невидимыми и обратно, при этом другие объекты меняться не будут.
14. Пользователь может менять расцветку объектов.

## **2.5. Пример работы приложения**

Т.Паскаля - Если [шестиугольник](https://ru.wikipedia.org/wiki/%D0%9C%D0%BD%D0%BE%D0%B3%D0%BE%D1%83%D0%B3%D0%BE%D0%BB%D1%8C%D0%BD%D0%B8%D0%BA) вписан в [окружность](https://ru.wikipedia.org/wiki/%D0%9E%D0%BA%D1%80%D1%83%D0%B6%D0%BD%D0%BE%D1%81%D1%82%D1%8C) (или в любое другое [коническое сечение](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D1%81%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) — [эллипс](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%BB%D0%B8%D0%BF%D1%81), [параболу](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D1%80%D0%B0%D0%B1%D0%BE%D0%BB%D0%B0), [гиперболу](https://ru.wikipedia.org/wiki/%D0%93%D0%B8%D0%BF%D0%B5%D1%80%D0%B1%D0%BE%D0%BB%D0%B0_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) или даже в пару [прямых](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D1%8F%D0%BC%D0%B0%D1%8F)), то точки пересечения трёх пар противоположных сторон лежат на одной прямой. Эту прямую называют прямой Паскаля.



Точка Лемуана - точка пересечения прямых, соединяющих каждую вершину треугольника с точками пересечения касательных к описанной окружности, проведённых из двух других вершин.



# **Заключение**

В процессе создания приложения были изучены основные понятия и методы комплексной плоскости, были описаны и вычислены основные формулы для работы в комплексных координатах и на основе всего вышесказанного была достигнута **цель проекта**, а именно, было создано приложение для построения точных геометрических чертежей, которое уже протестировали 6 моих знакомых. Созданное приложение, а так же другие материалы проекта доступны на Github по ссылке <https://github.com/Pro100Zhenya/Geogebra>

# **Список литературы**

1. Понарин, Я. П. **Алгебра** **комплексных** чисел в геометрических задачах: **Книга** для учащихся математических классов школ, учителей и студентов педагогических вузов / Я. П. Понарин. — 2-е изд. — Москва : МЦНМО, 2014.
2. Documentation of pyinstaller: <https://pyinstaller.org/en/stable/>
3. [Алгоритм Брезенхэма — Википедия (wikipedia.org)](https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC_%D0%91%D1%80%D0%B5%D0%B7%D0%B5%D0%BD%D1%85%D1%8D%D0%BC%D0%B0)
4. Documentation of Pygame: [Pygame documentation — DevDocs](https://devdocs.io/pygame/)