|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ «Информатика и системы управления»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Лабораторная работа № 3**

|  |  |
| --- | --- |
| **Тема** Реализация и исследование алгоритмов построения отрезков  **Студент** Склифасовский Д. О.  **Группа ИУ 7-45**  **Оценка (баллы) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |  |

Москва.

2020 г.

**Цель работы:**

Научиться выполнять построение отрезков различными алгоритмами и проанализировать их.

**Техническое задание:**

1. Рисование отдельных отрезков и сравнение их визуальных характеристик:

1\* Алгоритм цифрового дифференциального анализатора;

2\* Алгоритмы Брезенхема;

3\* Алгоритм Ву;

4\* Библиотечный алгоритм;

1. Исследование визуальных характеристик для отрезка, расположенного во всем спектре изменения углов;

**Теоретический материал:**

Общие требования:

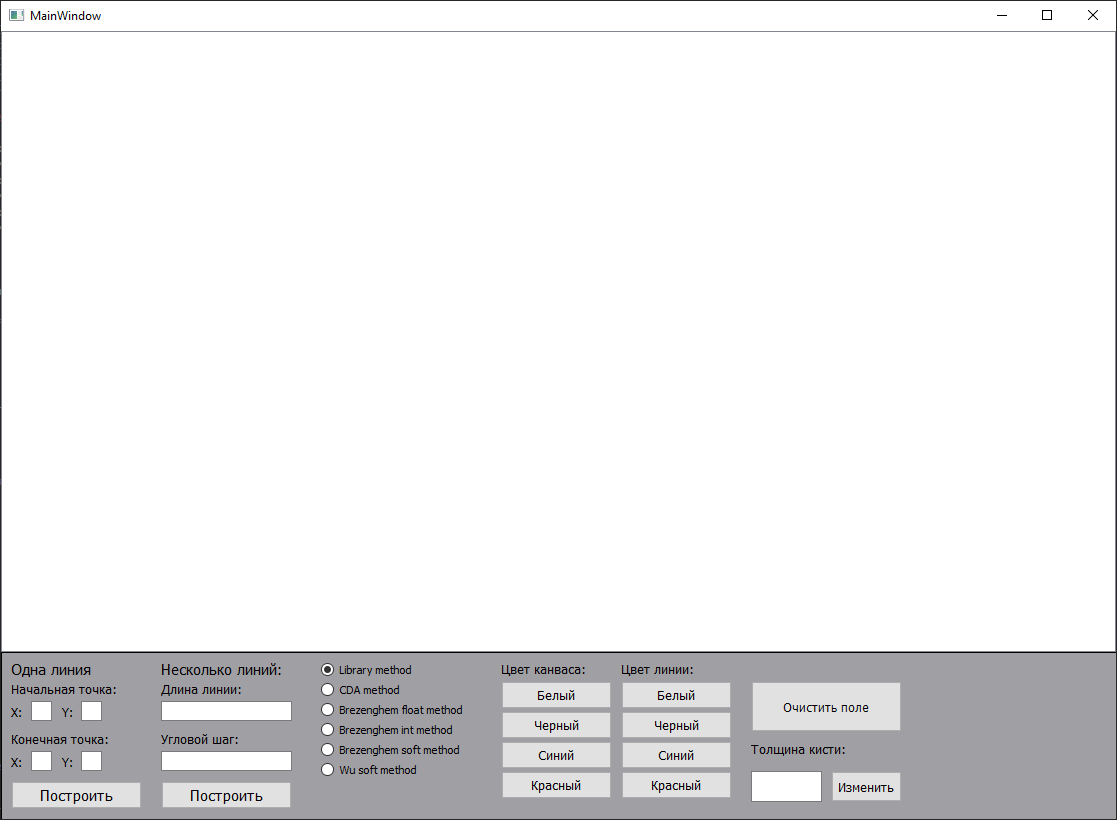
1. Отрезок должен выглядеть как отрезок прямой, начинаться и заканчиваться в заданных точках

2. Интенсивность (яркость) вдоль отрезка должна быть постоянной. Отрезки, имеющие разные углы наклона, должны быть одной интенсивности. Восприятие человека зависит не только от интенсивности свечения объекта, но и от расстояния между светящимися объектами //чтобы удовлетворить этому требованию, надо высвечивать точки с переменной интенсивностью от расстояния – потребует дополнительных вычислений, без особой нужды не используется

3. Алгоритмы (особенно нижнего уровня) должны работать быстро

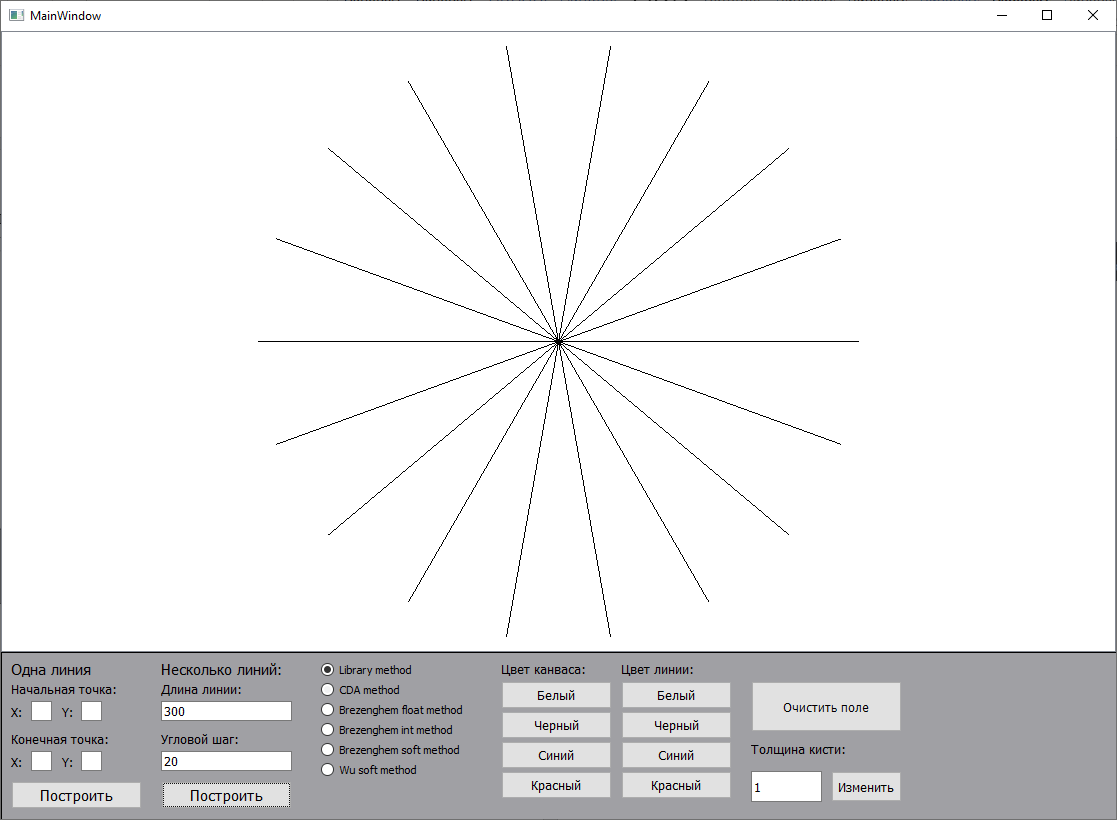
Все алгоритмы имеют пошаговый характер – на очередном шаге высвечиваем пиксель, и производим вычисления, используемые в следующем шаге.

**Интерфейс:**



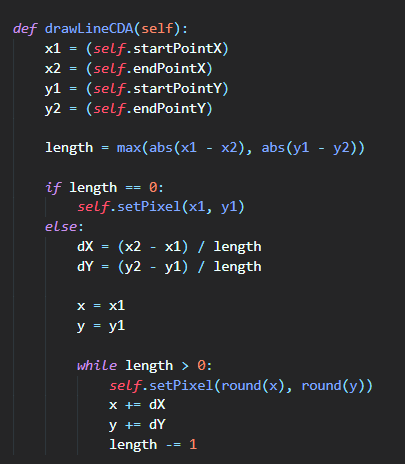
**Библиотечный метод:**

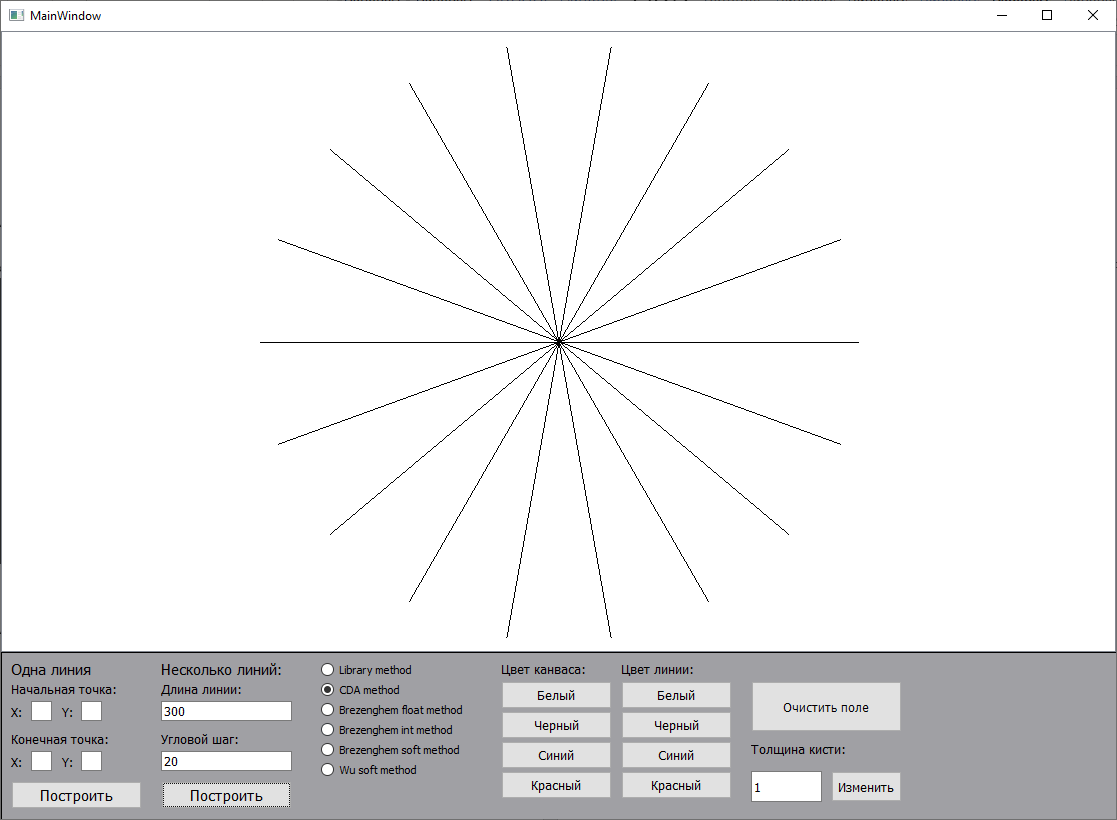




**Алгоритм цифрового дифференциального анализатора:**

Недостатки – работает с целочисленной арифметикой. Работает медленнее за счет операции округления.



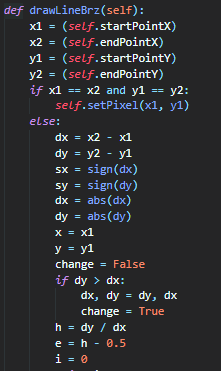


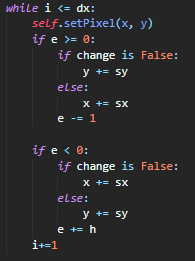
Алгоритм:

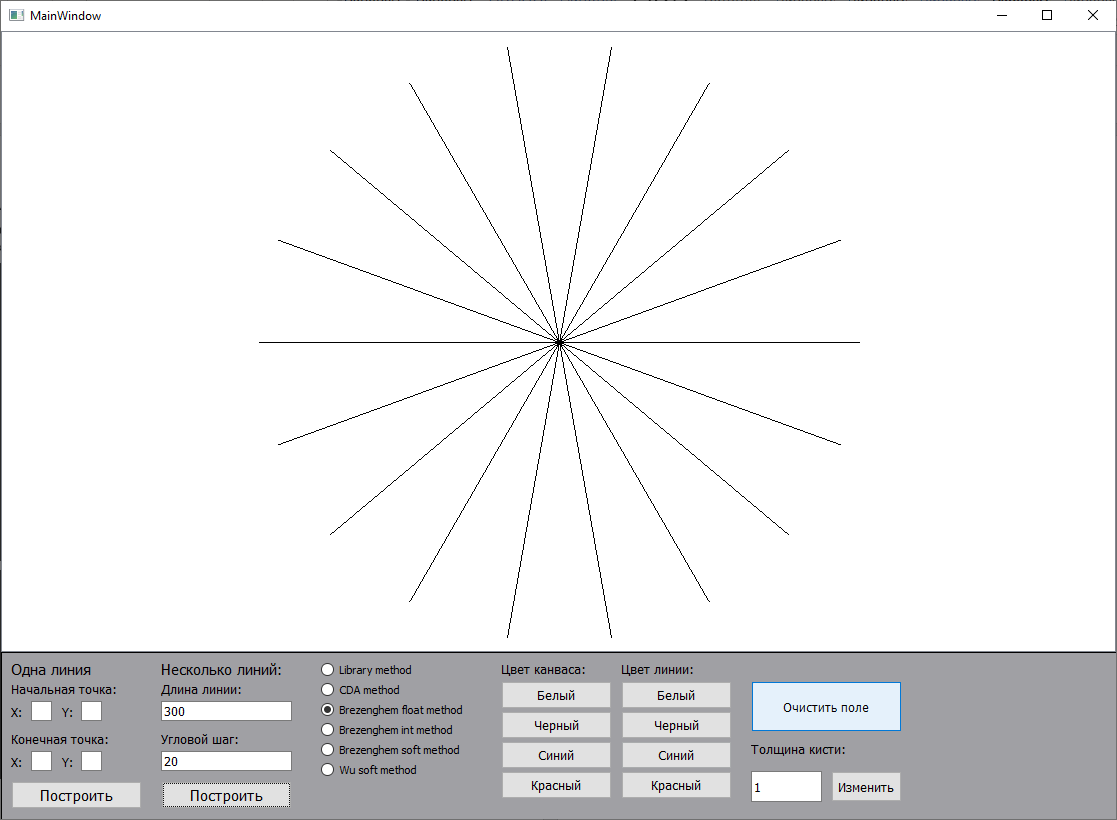
1. Ввод Xн, Yн, Xк, Yк
2. Проверка на вырожденность
3. Если abs(dx) > abs(dy), то длина L = abs(dy), иначе L = abs(dx)
4. dX = (Yк – Yн) / L, dY = (Yк – Yн) / L
5. Xтек = Xн, Yтек = Yн
6. Цикл по i=1 до L + 1
   1. Вычвечивание точки Xт;Yт
   2. Вычисление следующих координат Xт += dX, Yт += dY

**Алгоритм Брезенхема с действительными коэффициентами:**

Ошибка еi – величина изменяющаяся на каждом шаге , расстояние между точкой самого отрезка и точкой, аппроксимирующей его на очередном шаге





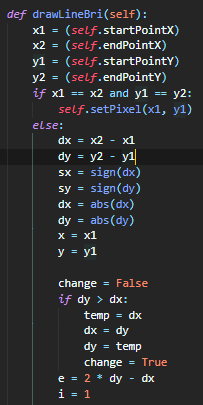


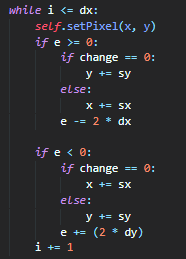
Алгоритм:

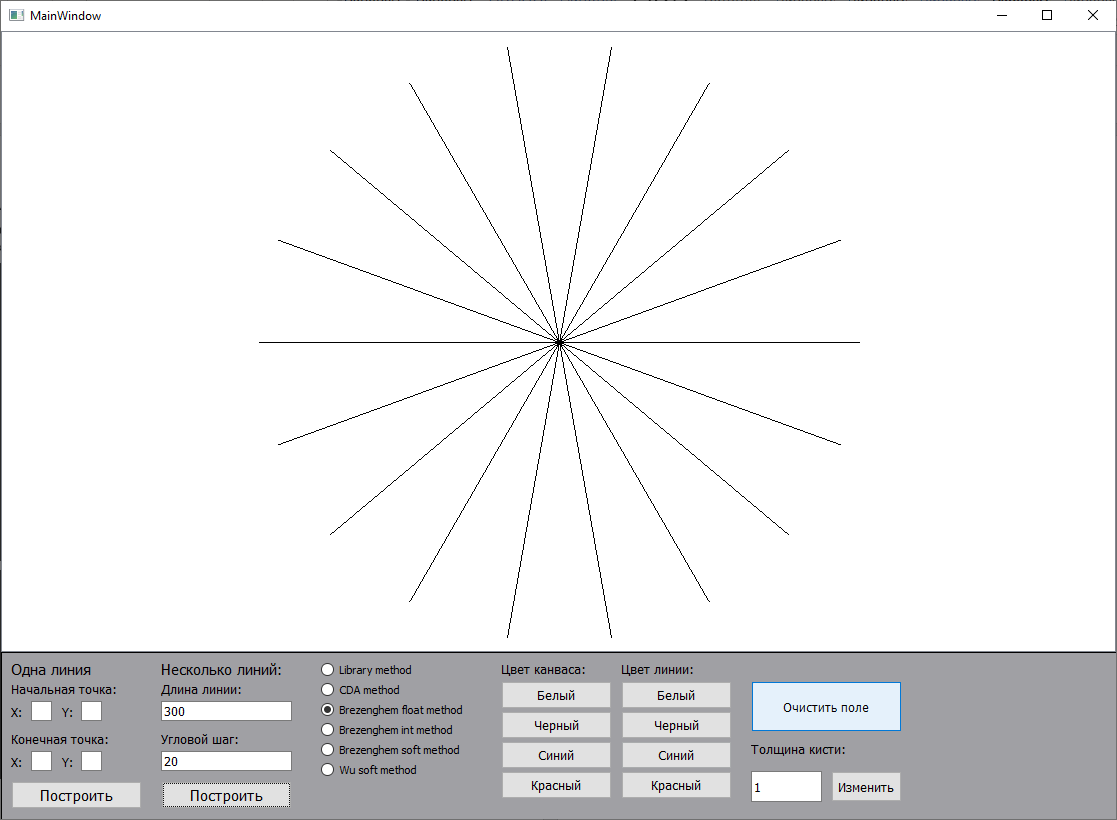
1. Начало
2. Ввод хн, ун, хк, ук
3. Проверка отрезка на вырожденность: если вырожденный то высветить точку хн,ун и переход на конец
4. Xтек = Xн, Yтек = Yн
5. dX = (Yк – Yн), dY = (Yк – Yн)
6. Sx=sign(dx), sy=sign(dy)
7. dx=|dx|, dy=|dy|
8. Если dx>dy то обмен=0, иначе обмен=1 {t=dy; dy=dx; dx=t}
9. ; .
10. Цикл построения отрезка (по i=1 to dx+1)
    1. Высвечивание точки (xt,yt)
    2. Если (e>=0)   
        если (обмен=0) yt=yt+sy  
        иначе xt=xt+sx  
        e=e-1 иначе если (обмен=0) xt=xt+sx  
        иначе уt=уt+sy  
        e=e+m
11. Конец

Недостатки: не все переменные являются переменными целого типа (e,m - действительные).

**Алгоритм Брезенхема с целыми коэффициентами:**





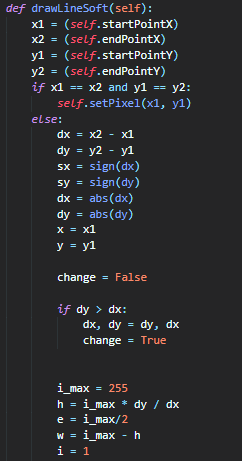


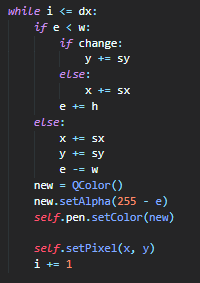
Алгоритм:

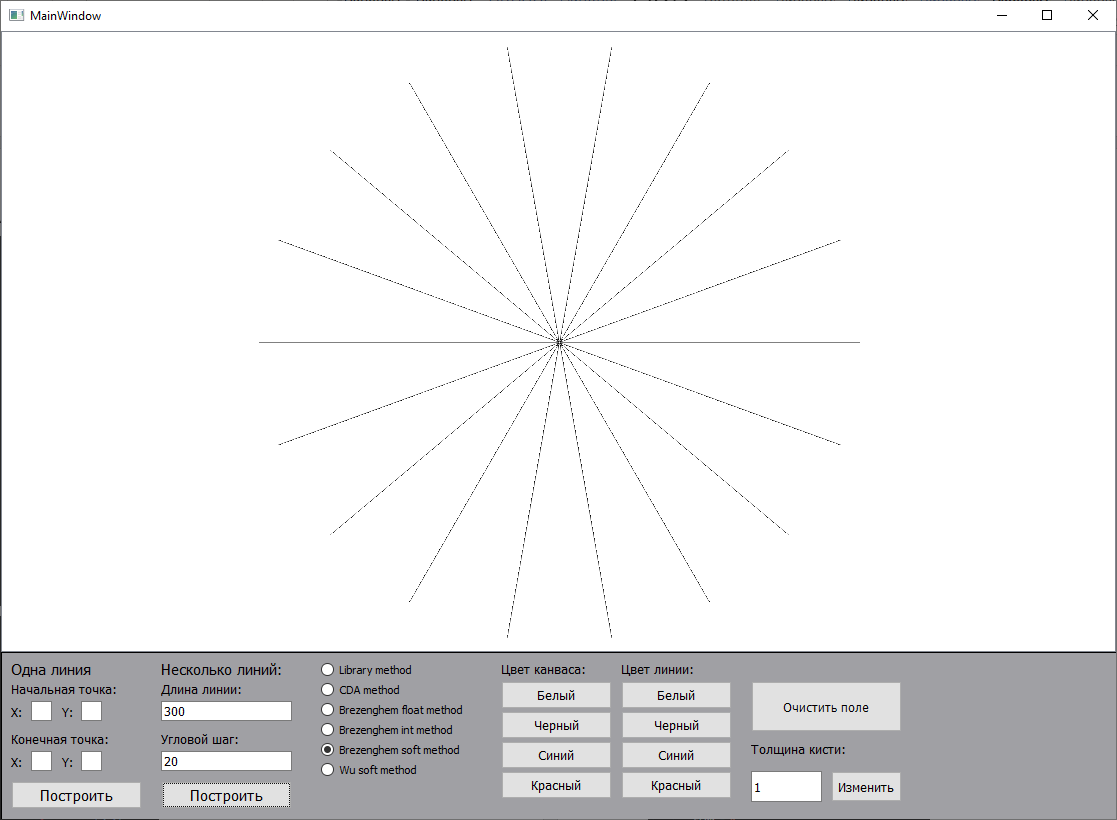
Чтобы перейти от предыдущего алгоритма к алгоритму, работающему с целыми:

. Тогда в цикле будет e=e-2dx; e=e+2dy.

**Алгоритм Брезенхема построения отрезка с устранением ступенчатости:**

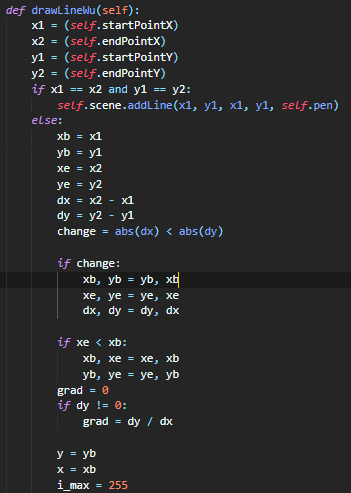


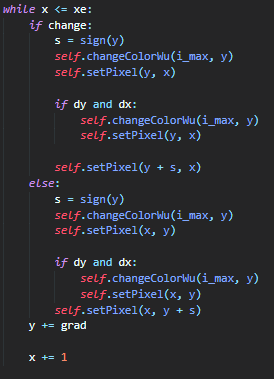


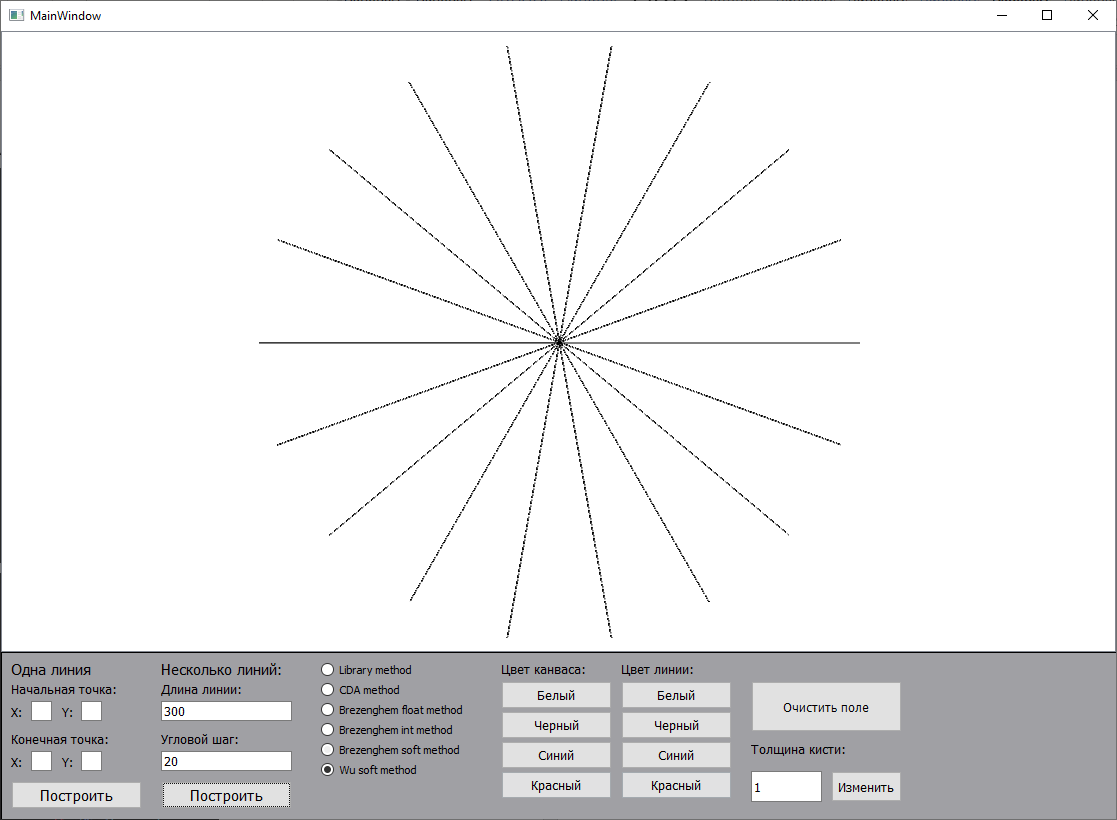


Сглаживание основывается на том, что каждый пиксель высвечивается со своим уровнем интенсивности.

**Алгоритм Ву:**



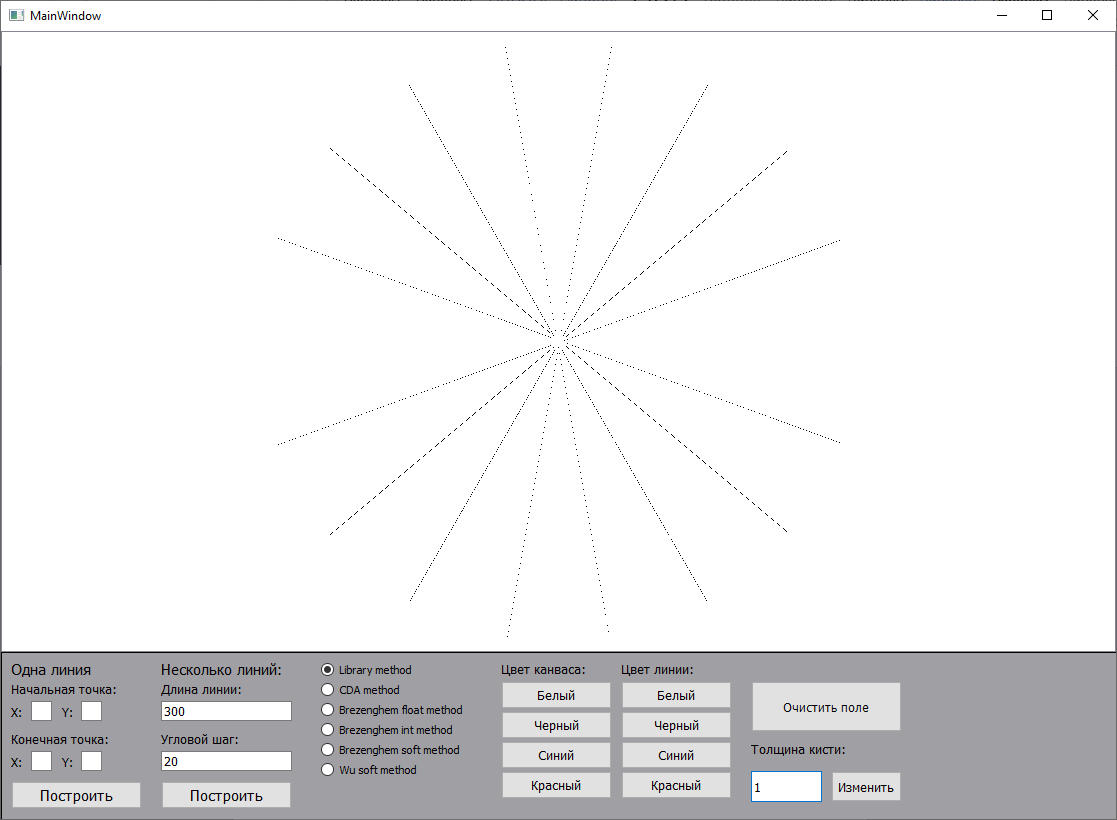




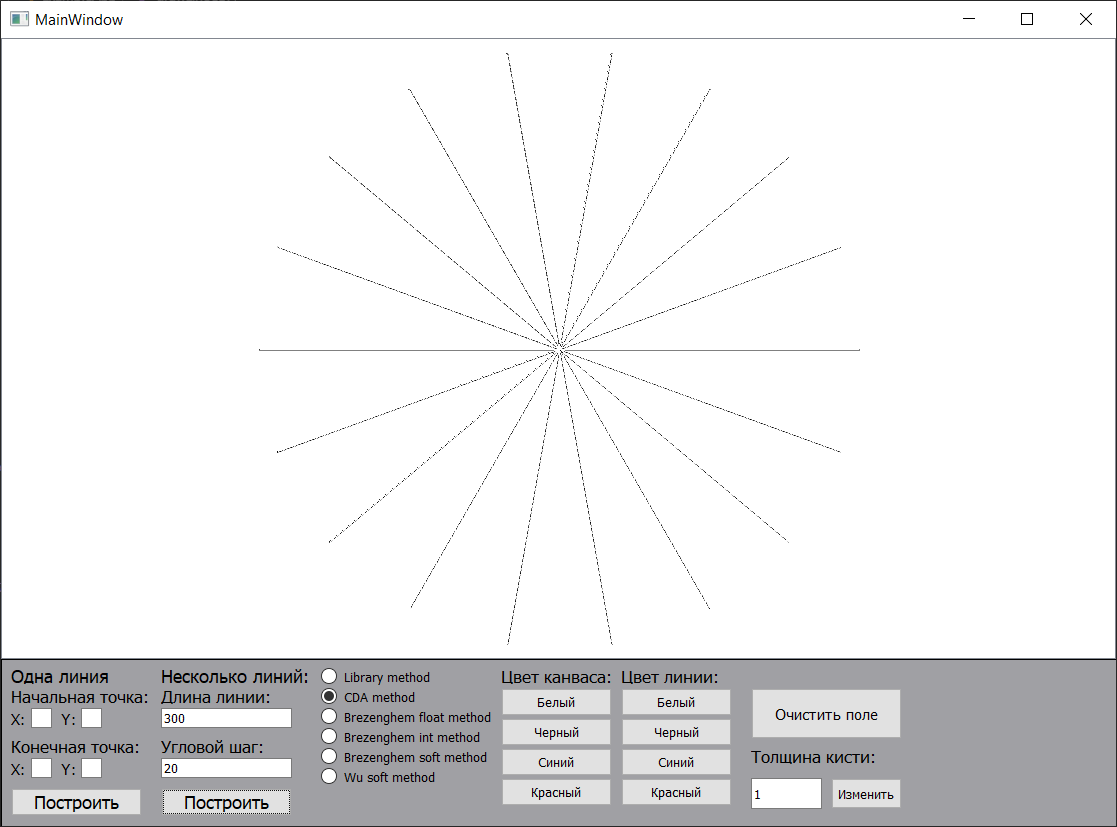
Чем ближе пиксель к точке идеального отрезка, тем с большей интенсивностью высвечивается.

Больше всего библиотечный алгоритм похож на алгоритм ЦДА:

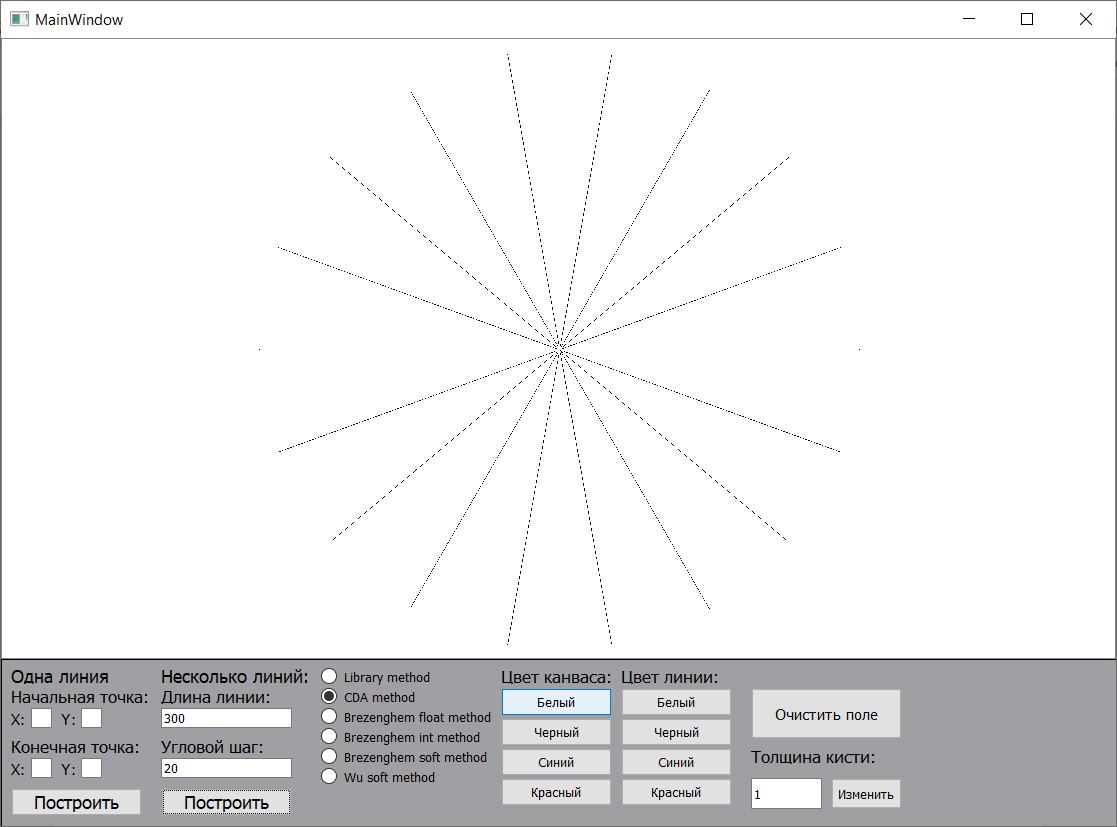
(наложение библиотечного на ЦДА)



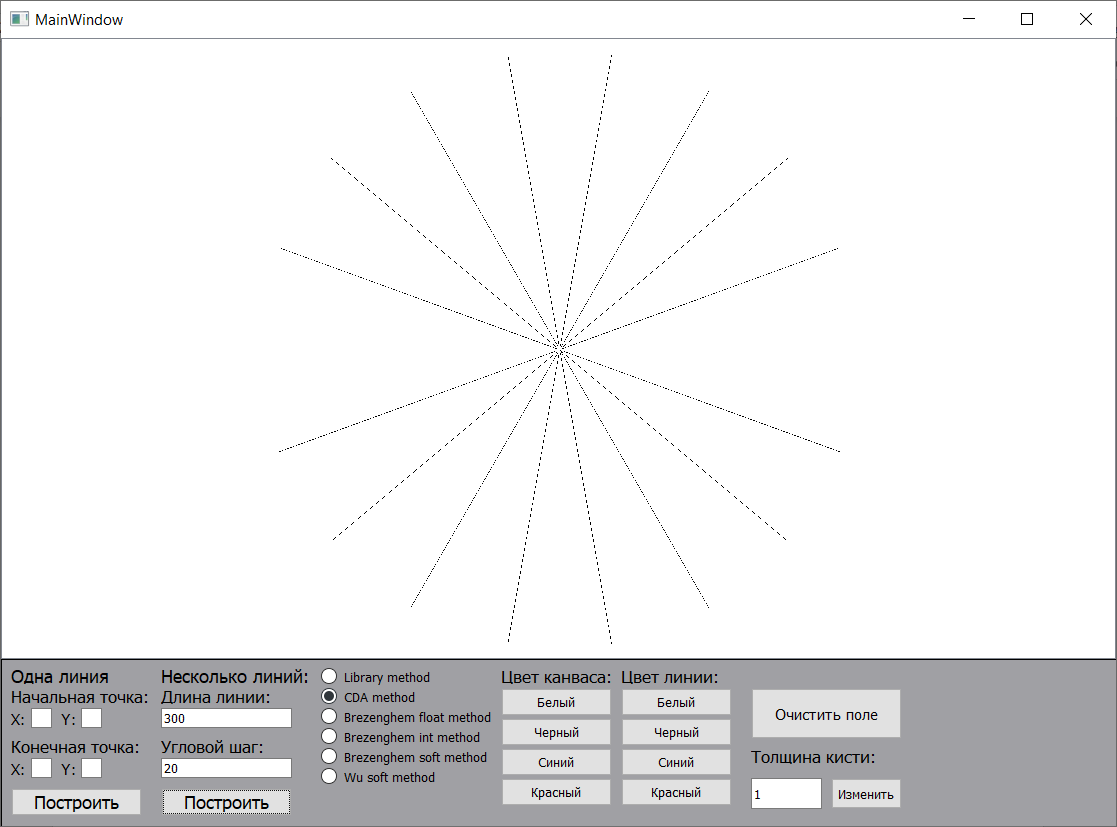
Сравнение ЦДА с методом Ву:



Сравнение метода Брезенхема с действ. Коэффициентами с ЦДА



Сравнение метода Брезенхема с целыми коэффициентами с ЦДА



Сравнение метода Брезенхема с целыми коэффициентами с методом Брезенхема с действ. Коэффициентами

