Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Московский государственный технический университет имени Н. Э. Баумана»

(МГТУ им. Н. Э. Баумана)

Факультет: Информатика и системы управления

Кафедра: Теоретическая информатика и компьютерные технологии

**Лабораторная работа №4**

**«Алгоритмы растеризации»**

**по курсу: «Компьютерная графика»**

Выполнил:

Студент группы ИУ9-42Б

Гасаев Г. К.

**Цель работы:**

* Реализовать алгоритм растровой развертки многоугольника:

построчное сканирования с упорядоченным списком ребер;

* Реализовать алгоритм фильтрации: целочисленный алгоритм Брезенхема с устранением ступенчатости;
* Реализовать необходимые вспомогательные алгоритмы (растеризации отрезка) с модификациями, обеспечивающими корректную работу основного алгоритма;
* Ввод исходных данных каждого из алгоритмов производить интерактивно с помощью клавиатуры и/или мыши. Предусмотреть также возможность очистки области вывода (отмены ввода);
* Растеризацию производить в специально выделенном для этого буфере в памяти с последующим копированием результата в буфер кадра OpenGL. Предусмотреть возможность изменения размеров окна;

**Основная теория:**

**Для доступа к необходимым функциям создания и работы с окном использовалась библиотека GLFW версии 3.**

**Вся логика по работе с буфером пикселей инкапсулирована в классе Field. В качестве поля объект такого класса хранит развернутую в массив матрицу пикселей - буфер, который копируется в буфер кадра OpenGL функцией glDrawPixels. Функции поля позволяют заполнять цветом разной интенсивности одиночные пиксели и рисовать прямые, которые растеризуются целочисленным алгоритмом Брезенхема. Наконец, доступна функция растеризации многоугольника, представленного вспомогательным классом Polygon в виде массива своих точек. Во время растеризации ребер, при которой используется и алгоритм устранения ступенчатости, происходит также их сканирование горизонтальной прямой, а в дальнейшем - заполнение по отсканированному списку.**

**Интерактивный ввод реализован средствами GLFW а именно - с помощью callback функций-обработчиков событий клавиатуры, мыши и изменения окна. Так как эти функции должны иметь доступ к объекту поля и рисуемого многоугольника, используется механизм передачи пользовательских данных через окно GLFW (glfwSetWindowUserPointer), для чего создан вспомогательный класс WindowContext. Функции управления:**

* **клик мыши внутри окна - добавление новой вершины многоугольника;**
* **F - соединений последней вершины с первой;**
* **N - переключения режима заполнения;**
* **C - очищение окна;**
* **P - увеличение окна в горизонтальном направлении;**

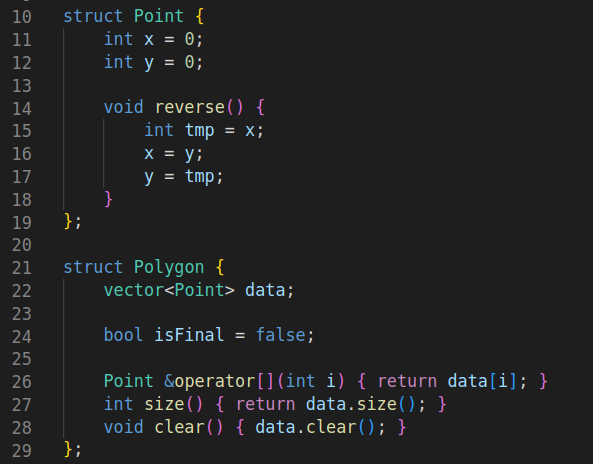
**Источники:**

[**https://www.khronos.org/opengl/**](https://www.khronos.org/opengl/)

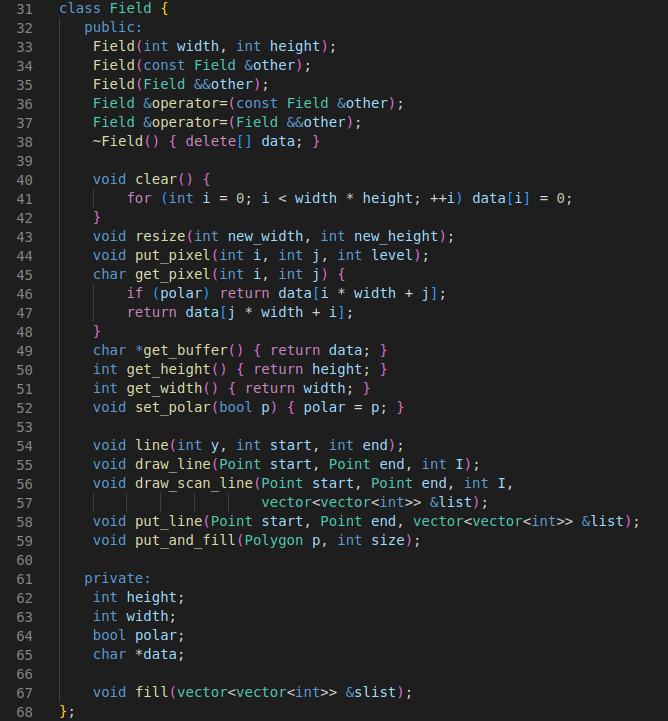
**http://aco.ifmo.ru/el\_books/computer\_visualization/lectures/7.html**

**Практическая реализация:**

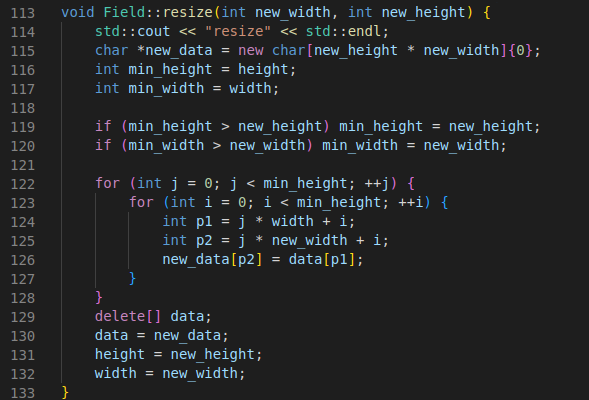
* **Вспомогательные классы:**



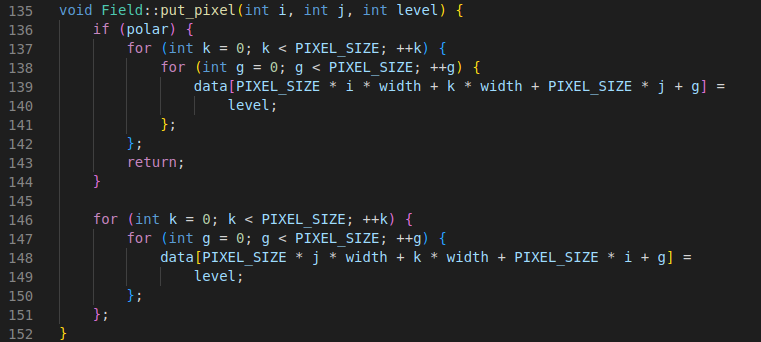
* **Класс Field:**



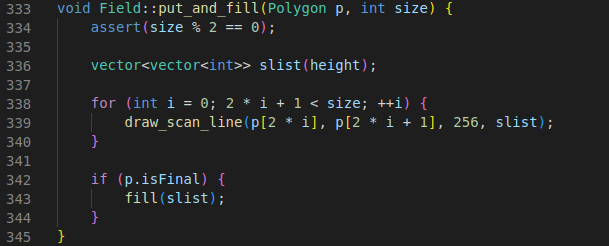
* **Изменение размеров поля:**



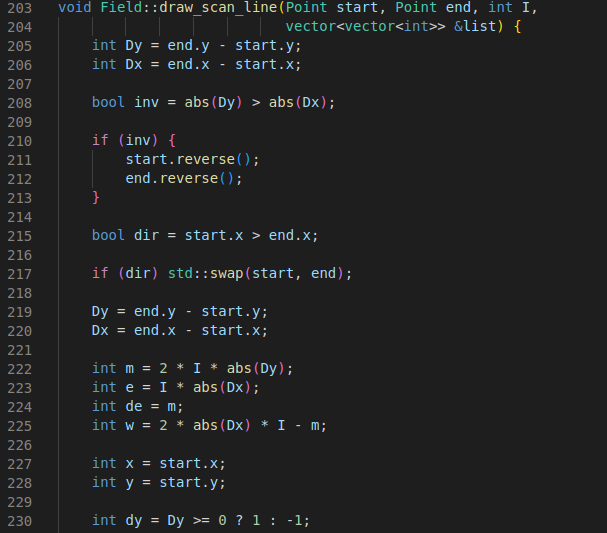
* **Добавление пикселя на поле:**

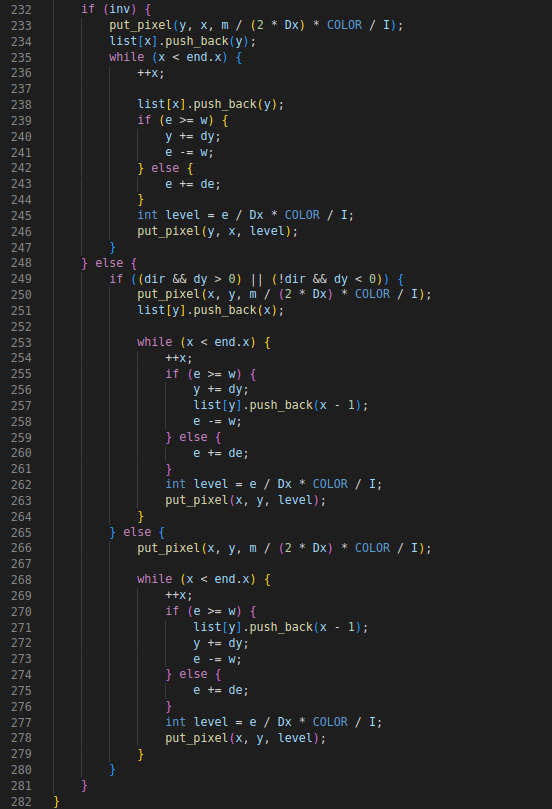


* **Добавление на поле многоугольника с заполнением:**

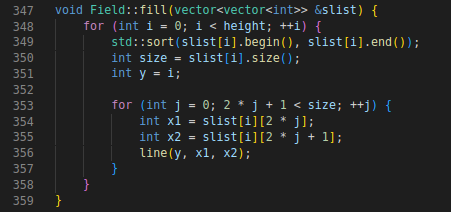


* **Добавление прямой со одновременным сканированием:**

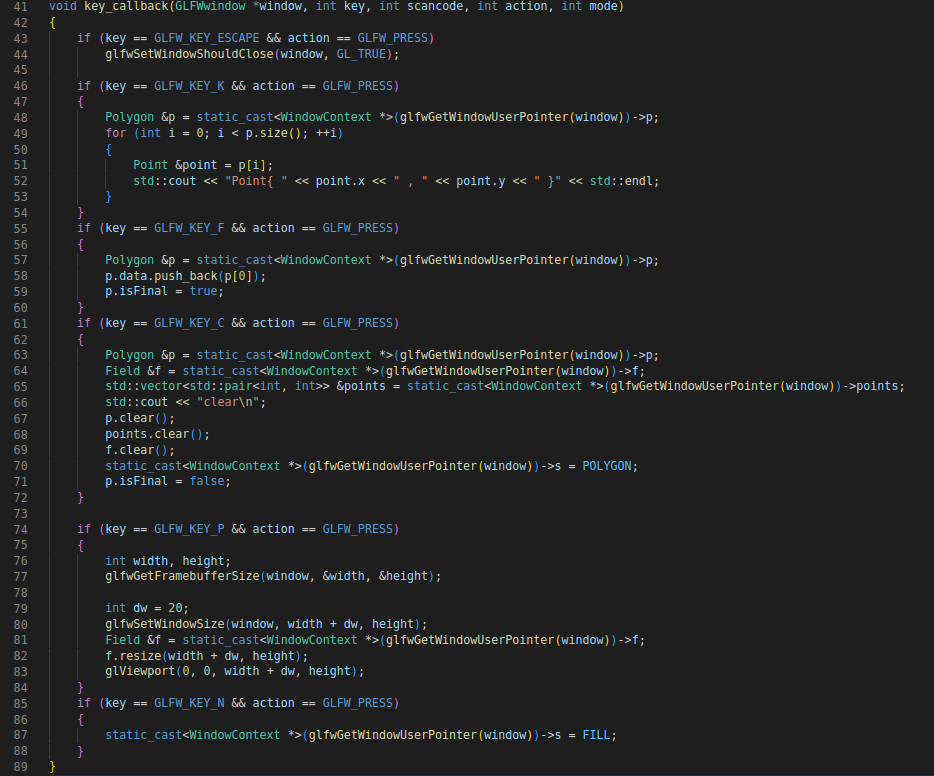




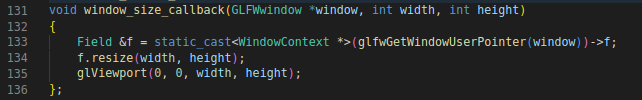
* **Заполнение многоугольника:**



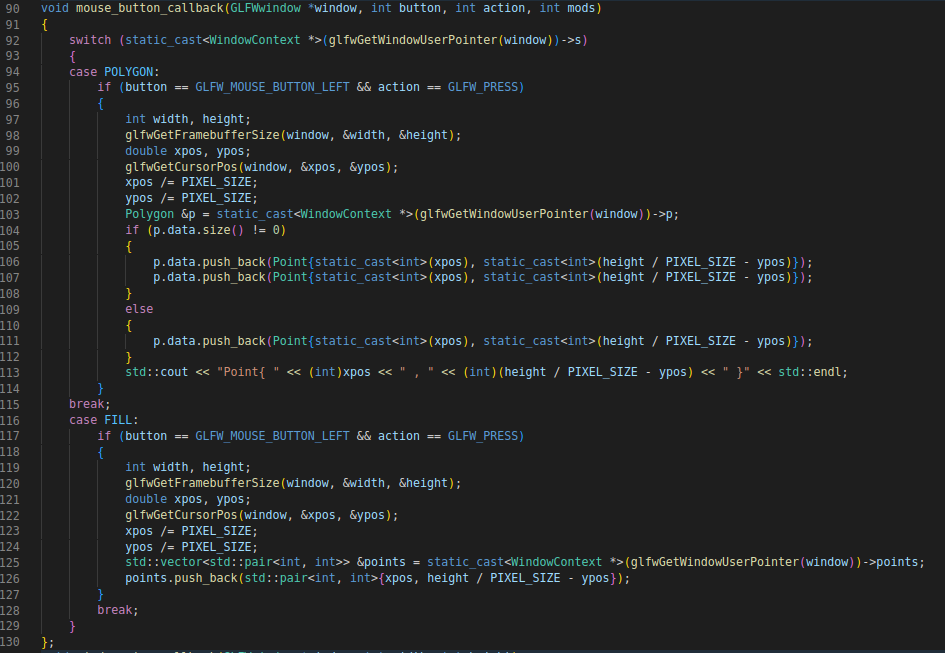
* **Функция-обработчик ввода клавиатуры:**



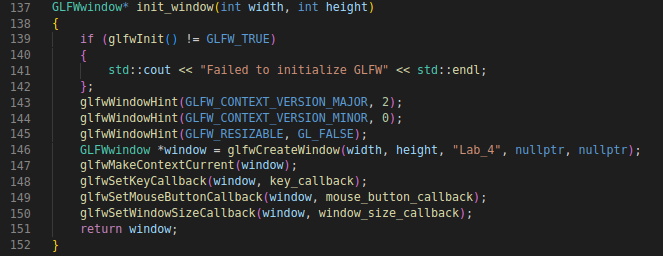
* **Функция-обработчик изменения размера окна:**



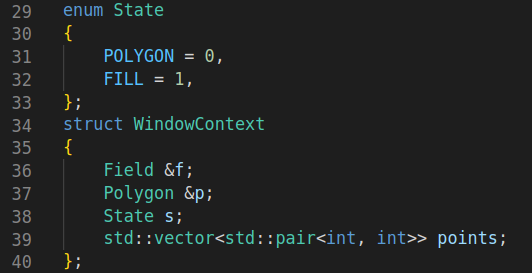
* **Функция-обработчик ввода мыши:**



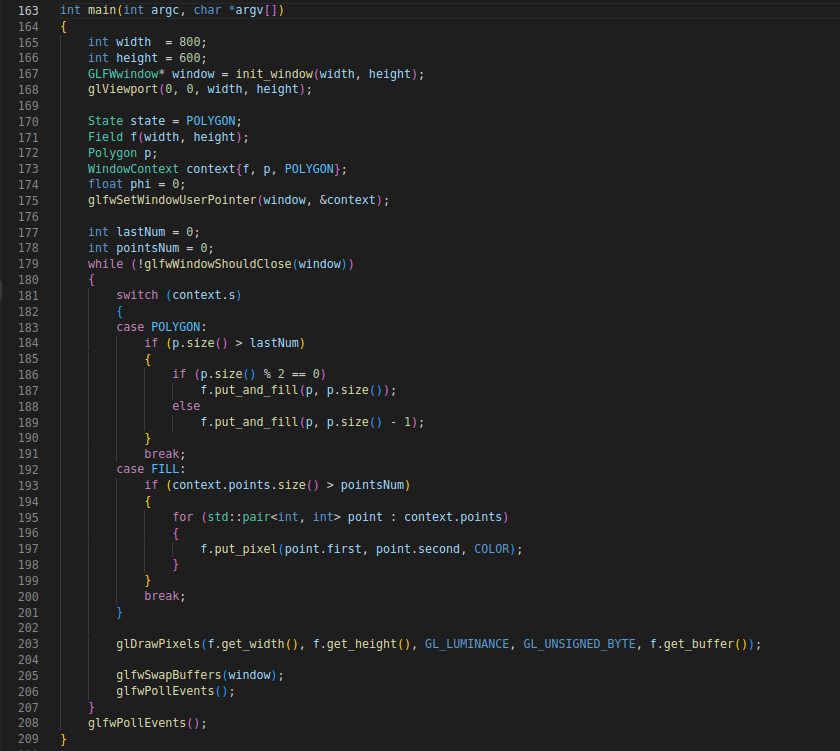
* **Создание окна:**



* **Вспомогательные структуры для обработчиков:**



* **Главный цикл:**



**Заключение:**

**В ходе лабораторной работы:**

1. **Реализован алгоритм растровой развертки многоугольника через построчное сканирования с упорядоченным списком ребер.**
2. **Реализована целочисленный алгоритм Брезенхема устранения ступенчатости ребер многоугольника.**
3. **Реализован вспомогательный алгоритм растеризации отрезка:**

**целочисленный алгоритм Брезенхема.**

1. **Растеризация происходит в специально выделенном буфере, инкапсулированном внутри объекта класса Field.**
2. **Настроен интерактивный ввод данных для растеризации при помощи мыши и клавиатуры.**