|  |  |
| --- | --- |
| **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 № 9**

|  |  |
| --- | --- |
| **Тема РЕАЛИЗАЦИЯ АЛГОРИТМА ОТСЕЧЕНИЯ ПРОИЗВОЛЬНОГО МНОГОУГОЛЬНИКА ВЫПУКЛЫМ ОТСЕКАТЕЛЕМ**  **Студент \_Турсунов Ж. Р.\_\_\_\_\_\_\_\_**  **Группа \_ИУ7-46Б\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Оценка (баллы) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Преподаватель \_Куров А. В.\_\_\_\_** |  |

Москва.

2020 г.

**Цель работы:** Изучение и программная реализация алгоритма Сазерленда-Ходжмена отсечения многоугольников.

**Входные данные:**  Ввод отсекателя — произвольного многоугольника. Высветить его первым цветом. Также необходимо обеспечить ввод выпуклого многоугольника (высветить их вторым цветом). Ввод осуществляется с помощью мыши и нажатия других клавиш.

**Выходные данные:** Пользовательское меню, содержащее поля ввода и холст с конечным изображением. Выполнить отсечение многоугольника, показав результат третьим цветом. Исходный многоугольник не удалять.

**Ошибочные ситуации:** Программа прекращается, если хотя бы один из входных данных не корректен.

**Теоретическая часть**

Для реализации этого алгоритма необходимо хорошо разобрать в внутреннем/внешнем отсечении, определении выпуклости, также в определении видимости/невидимости точки.

**Внутреннее** – определяем части отрезка внутри отсекателя и вычерчиваем. **Внешнее** – определяем

части вне отсекателя и вычерчиваем.

**Определение выпуклости**

Вычислить векторные произведения всех пар смежных сторон. Если все = 0, то многоугольник вырожден. Все разных знаков – многоугольник не выпуклый. Все ≥0 – выпуклый, нормали ориентированы влево от рёбер. ≤0 – выпуклый, нормали вправо.

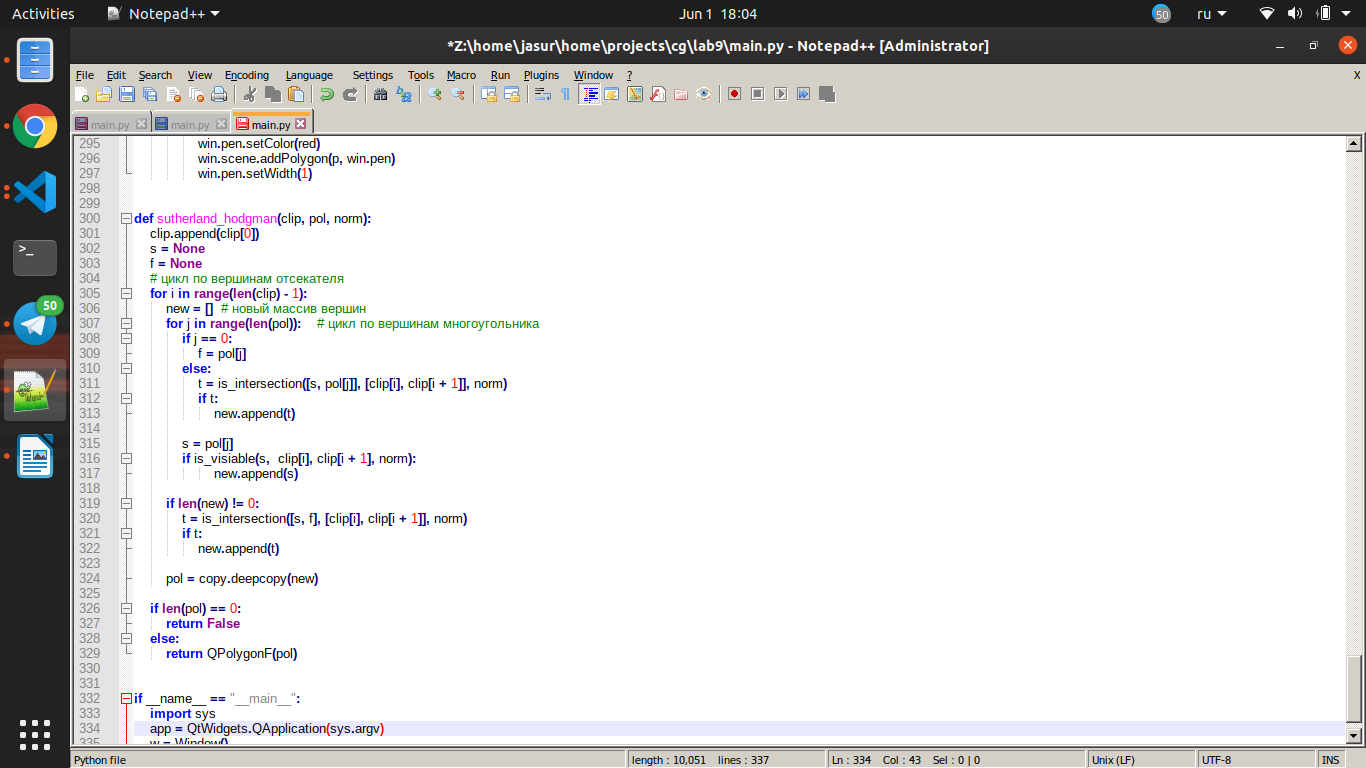
Исходный многоугольник отсекается последовательно каждой стороной.

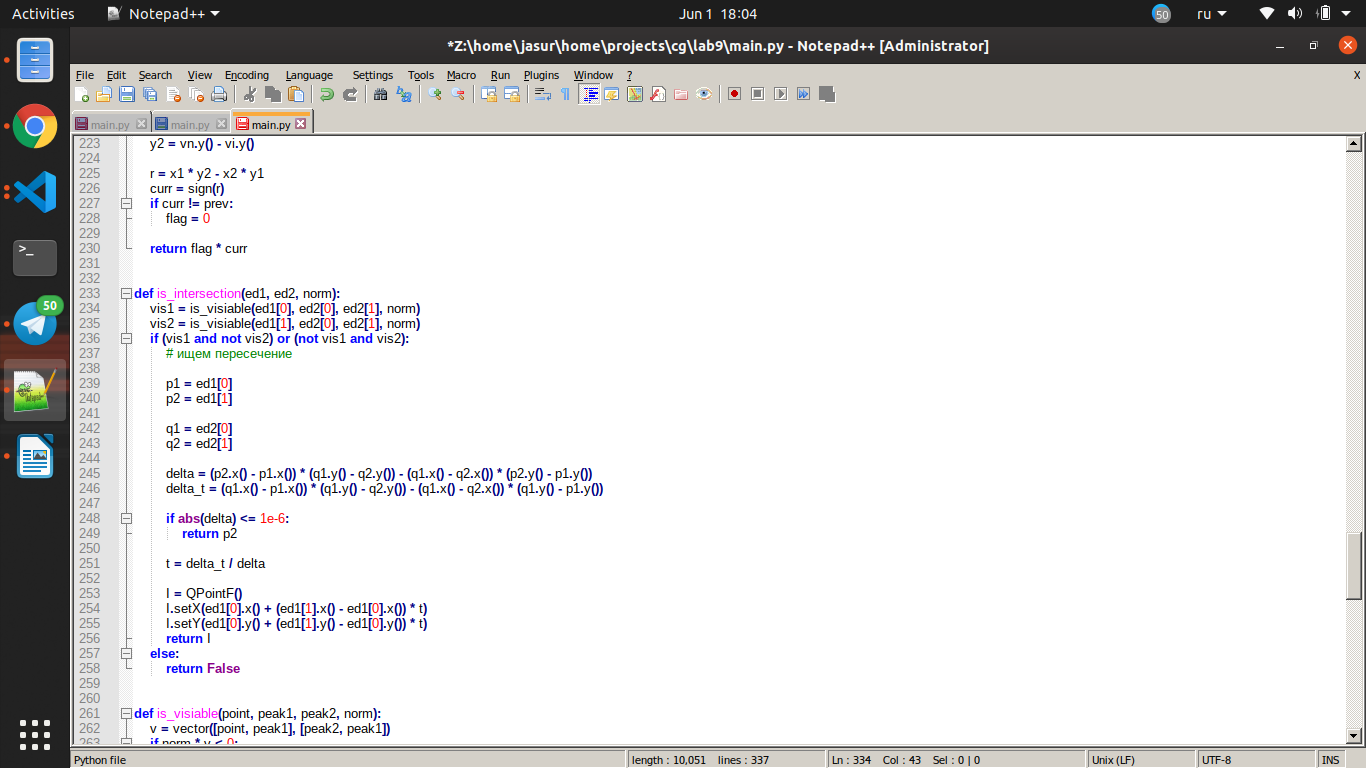
Сам алгоритм можно условно разделить на следующие пункты:

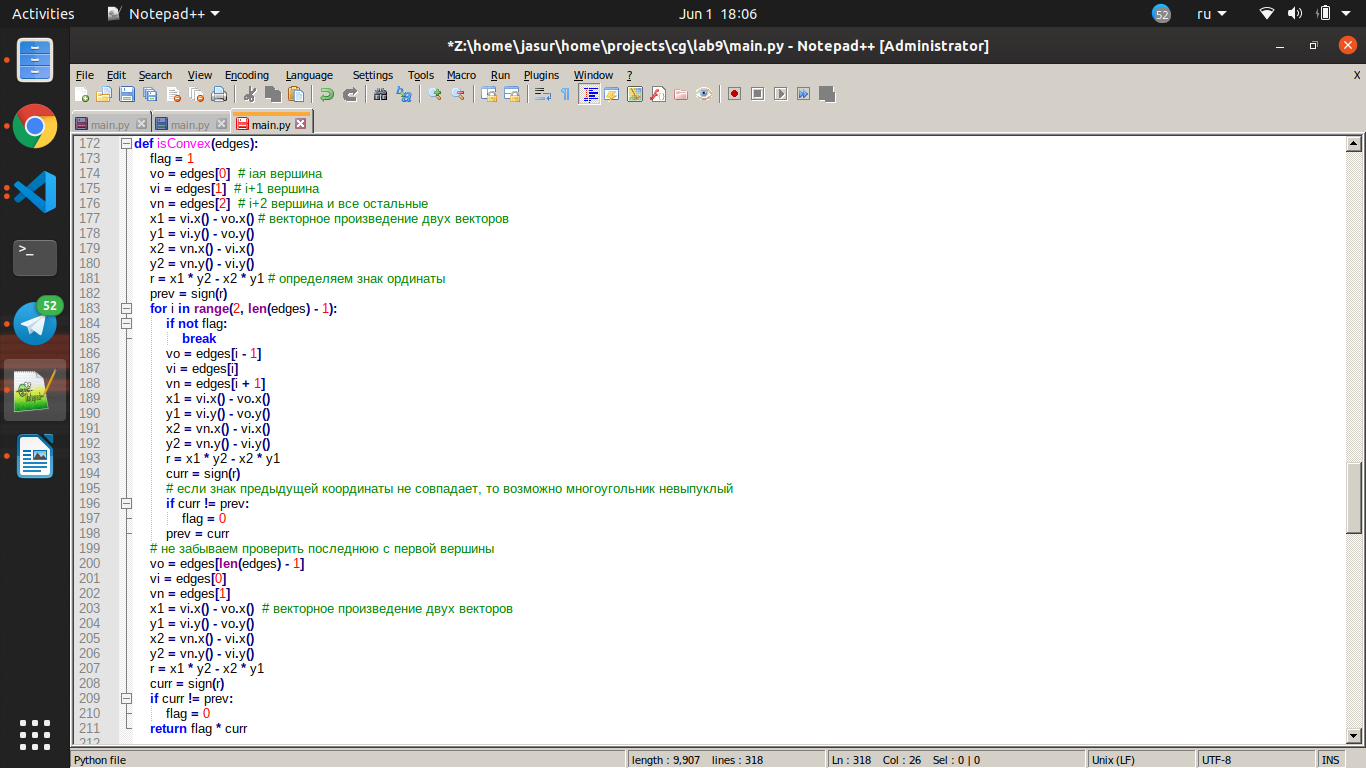
1. Определение видимости. Это можно определить различными способами(скалярное произведение, векторное произведение или использование пробной функции)
2. Нахождение точек пересечения. Первым делом необходимо убедиться в её наличии. Предварительно нужно убедиться в не параллельности прямых – точка пересечения должна существовать. Это определяется с помощью видимости концов ребра многоугольника – если видимость разная, то точка пересечения есть. Начальная вершина очередного ребра является одновременно и конечной вершиной для предыдущего ребра. Эта вершина анализируется (и заносится в результат если видима) на предыдущем шаге.
3. Занесение видимых вершин и найденных точек в результирующий список.

Данный алгоритм имеет недостаток — можно столкнуться с ситуацией построения «ложных ребер». Мы работаем с массивом вершин, вершины обходятся последовательно — ложным будет, то ребро, которое обходится два раза.

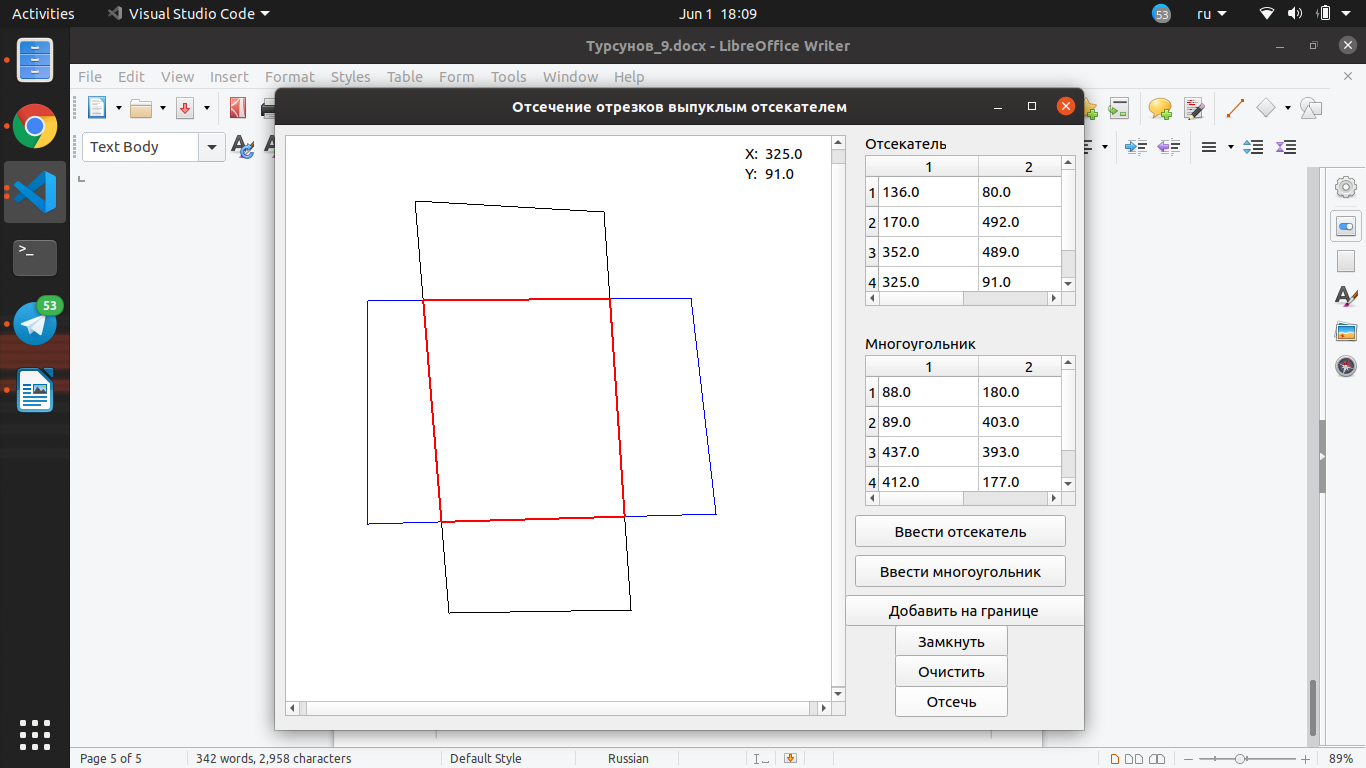
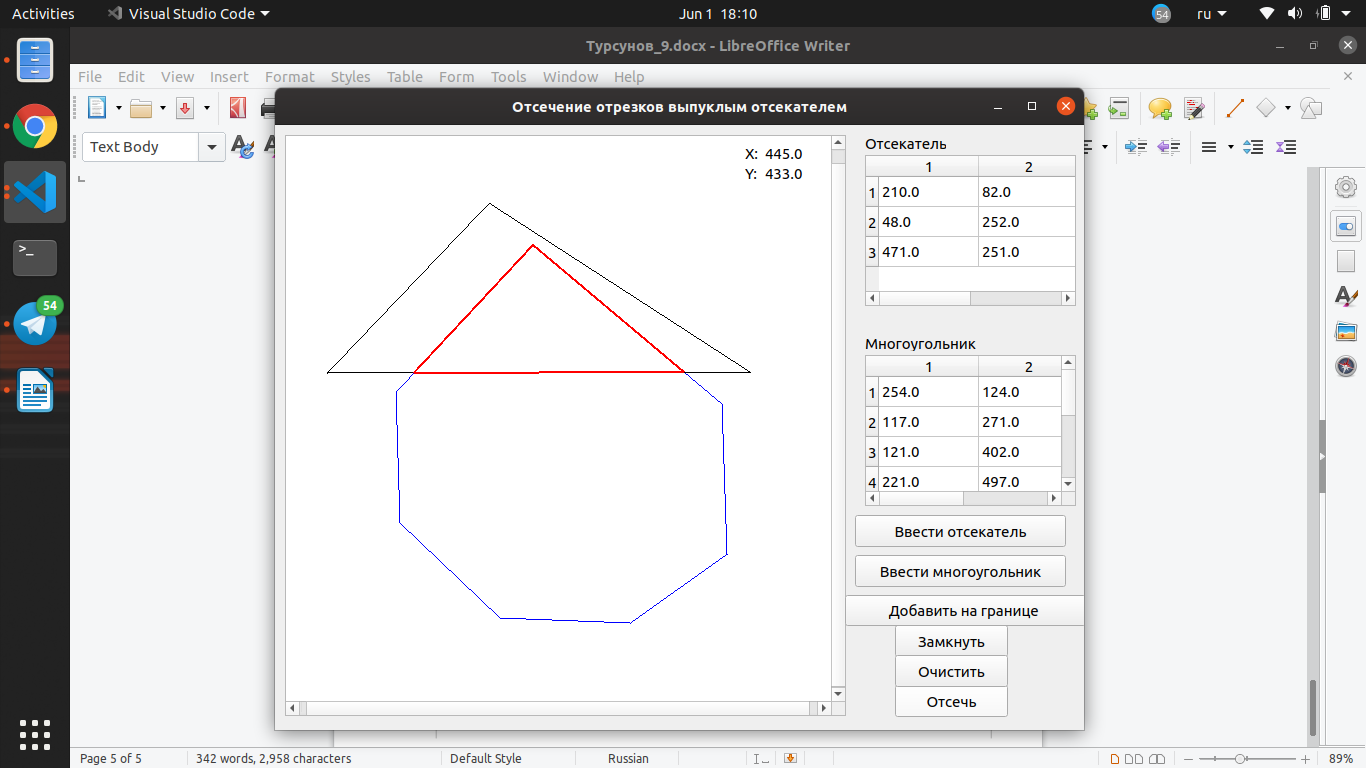
**Код программы**

****

****



**Пример работы**

****

