|  |  |
| --- | --- |
| **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 № 8**

|  |  |
| --- | --- |
| **Тема \_РЕАЛИЗАЦИЯ АЛГОРИТМА ОТСЕЧЕНИЯ ОТРЕЗКА ПРОИЗВОЛЬНЫМ 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 отсечения отрезка.

**Входные данные:**  Ввод отсекателя — произвольного многоугольника. Высветить его первым цветом. Также необходимо обеспечить ввод нескольких (до десяти) различных отрезков (высветить их вторым цветом). Отрезки могут иметь произвольное расположение: горизонтальные, вертикальные, имеющие произвольный наклон. Ввод осуществляется с помощью мыши и нажатия других клавиш.

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

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

**Теоретическая часть**

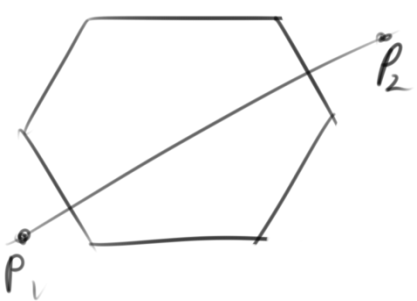
**Описание алгоритма**

**(Алгоритм Кируса-Бека)**

Отсечение (внешнее) – операция удаления части изображения, находящегося за пределами некоторой заданной области (отсекателя). Удаление части области внутри отсекателя – стирание.

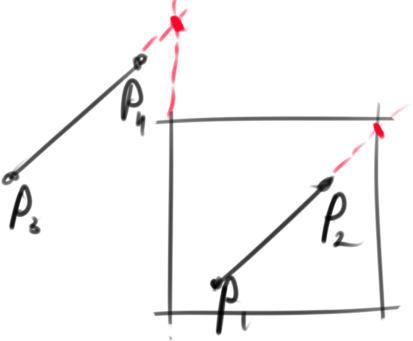
Отсечение может происходить на плоскости и в 3Д-пространстве. Для его выполнения необходимо задать отсекатель – регулярный или нерегулярный. Регулярные: прямоугольник на плоскости, стороны параллельны осям; параллелепипед в пространстве, грани параллельны плоскостям. Нерегулярные: выпуклые и невыпуклые многоугольники; четырёхгранная усеченная пирамида (правильная)

Границы относят к внутренностям области. Точка – лежит либо внутри, либо снаружи. Точка видима, если находится внутри отсекателя и невидима иначе.

*.*

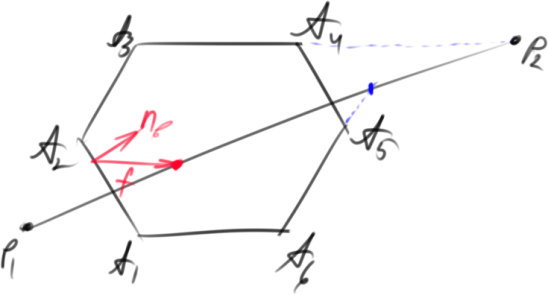
Здесь t – параметр для параметрической формы задания отрезка. Де факто, имеем два уравнения: y(t) = y1 +…, x(t) = x1 + ….

При попытке найти точки пересечения для полностью видимого отрезка, можно получить точку, находящуюся за пределами отсекателя – не проходит проверка на корректность. Раньше мы видимые отрезки отсеивали, а здесь нет. Для полностью невидимого отрезка можно получить аналогичную картину.



Простыми средствами отсеять эти отрезки невозможно, если полная видимость\невидимость и обнаруживается, то только в процессе работы.

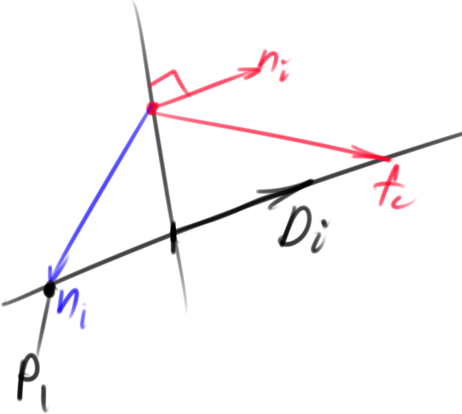
Прямая пересекает выпуклый многоугольник только в двух точках.



Используем скалярное произведение векторов – вектора нормали к стороне, и вектора от точки на ребре к точке на отрезке (красные).

*. //fi –* просматриваем очередную i-ю сторону. В качестве произвольной f удобно задавать вершину отсекателя.

Вектор директрисы, направления отрезка - ;



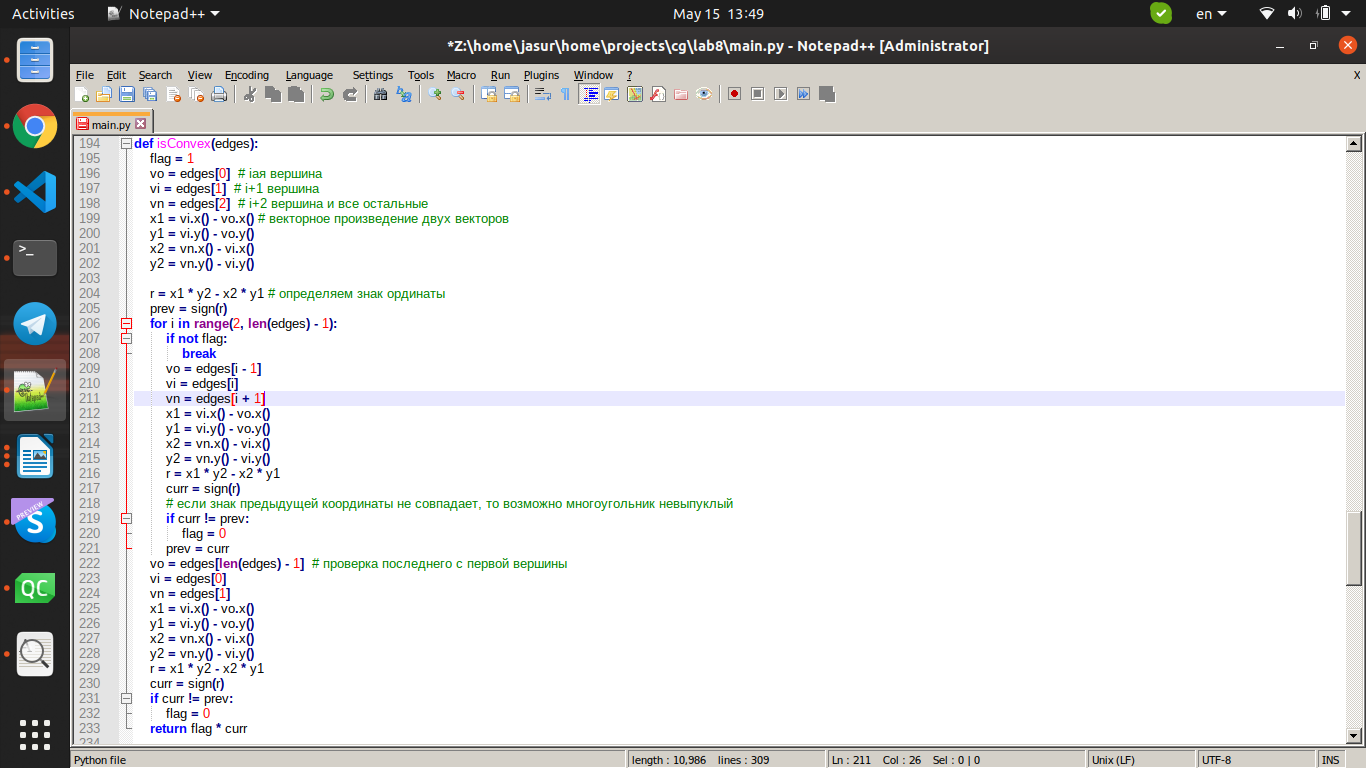
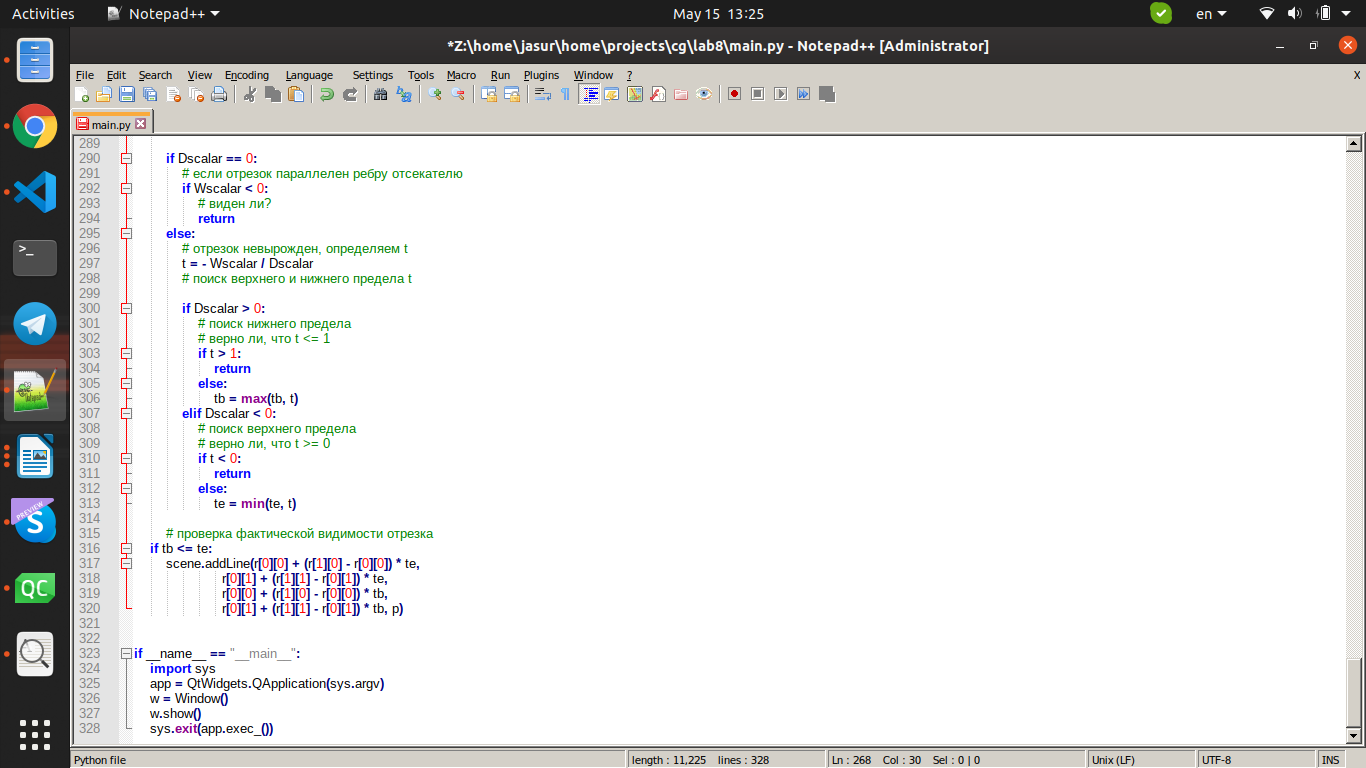
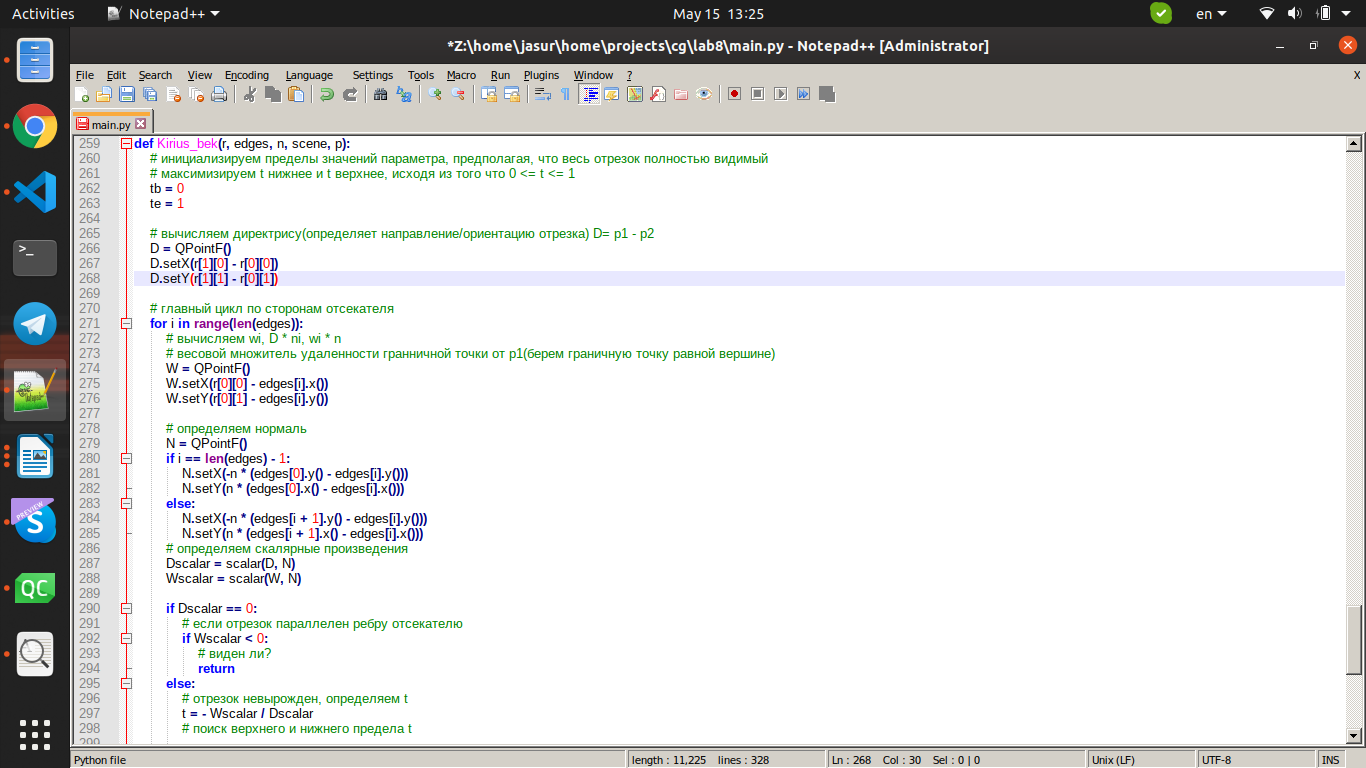
меем сумму скаляров: , отсюда . Анализируем знаменатель на равенство 0: может быть Д=0 – отрезок вырождается в точку, которую анализируем на видимость. Знаменатель = 0, когда нормаль перпендикулярна директрисе, а значит сам отрезок паралеллен стороне отсекателя.

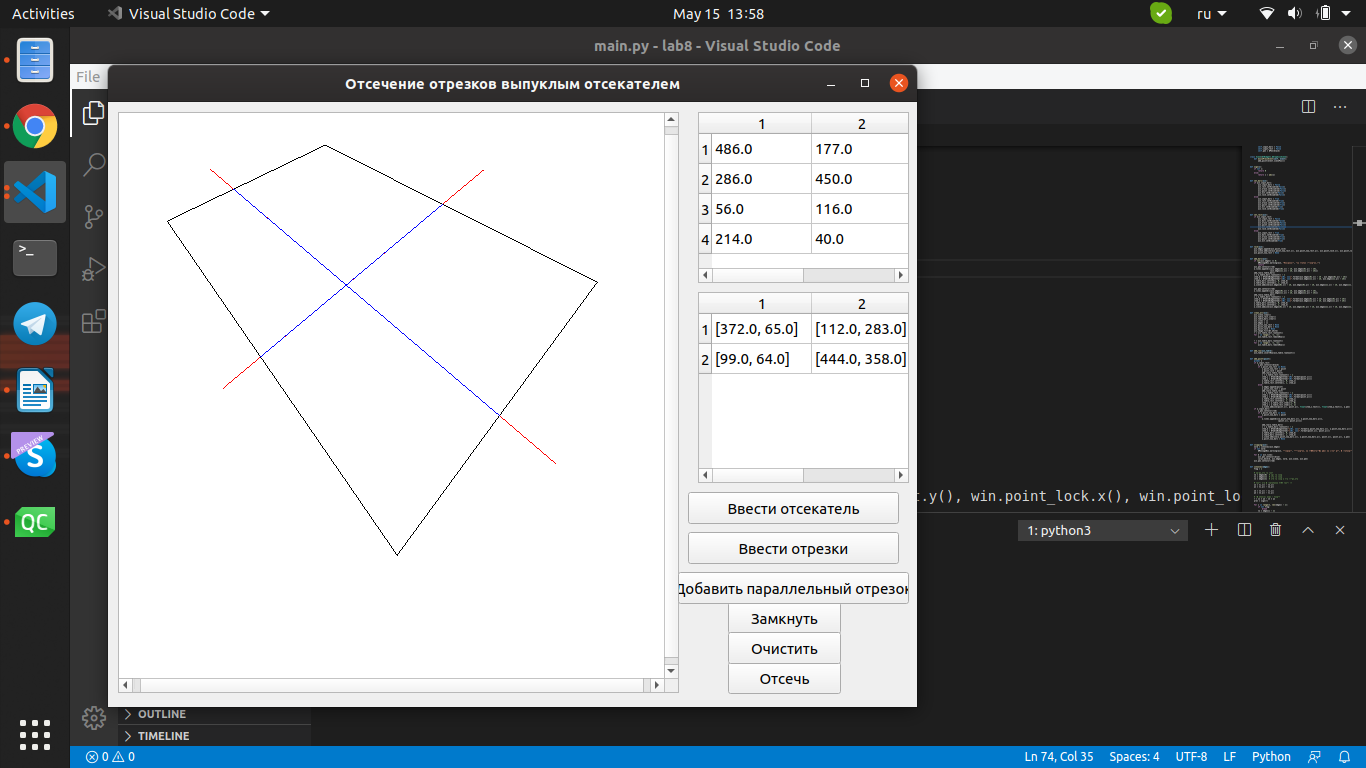
Видимость точки Р1 определяется с помощью скалярного произведения вектора внутренней нормали n на вектор первой вершины W, . <0 – Р1 невидима, >0 – видима, =0 – лежит на границе.

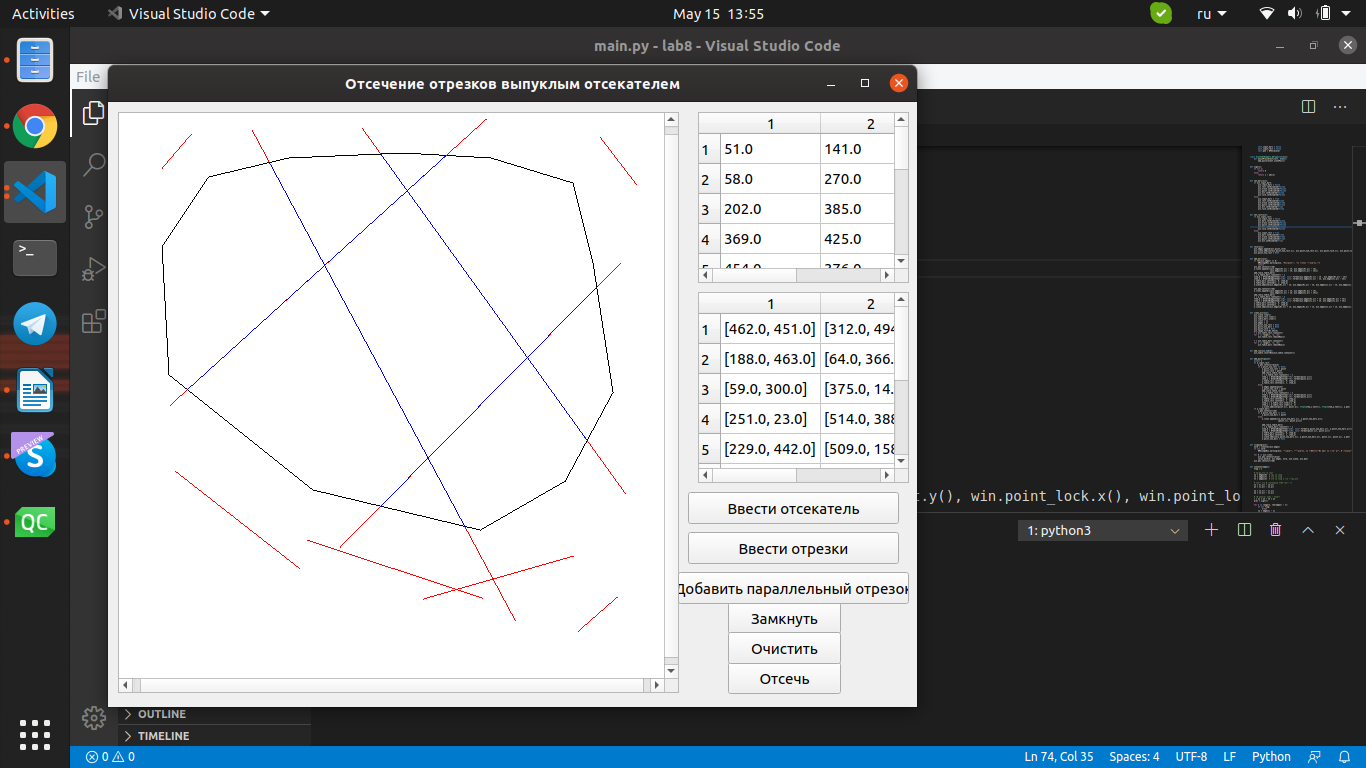
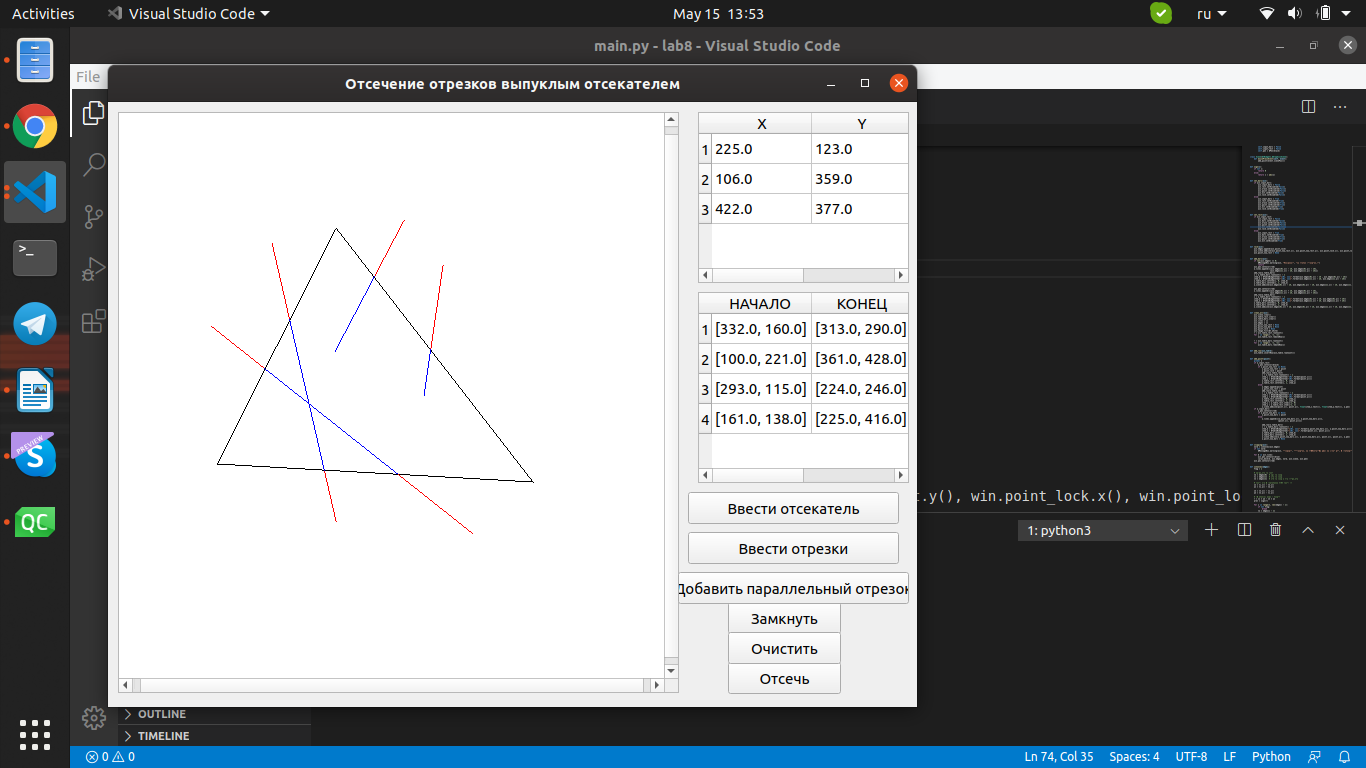
*–* отрезок параллелен стороне отсекателя, нужно проверить его видимость. Отрезок может лежать по видимую или по невидимую сторону. Сделать это можно, проверив видимость любой его точки, например Р1 (с помощью nW). Если относительно текущей границы отрезок видим, то переходим к следующему шагу – ищем точки пересечения отрезка с рёбрами. Если же отрезок невидим относительно текущей границы, то он невидим и относительно остального многоугольника.

Вообще, отрезок может пересекаться с продолжением всех сторон отсекателя, а нам надо выбрать всего две точки.

**Код программы**

****

**Пример работы**

****

