|  |  |
| --- | --- |
| **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 № 10**

|  |  |
| --- | --- |
| **Тема РЕАЛИЗАЦИЯ АЛГОРИТМА ПЛАВАЮЩЕГО ГОРИЗОНТА ПОСТРОЕНИЯ ТРЕХМЕРНЫХ ПОВЕРХНОСТЕЙ**  **Студент \_Турсунов Ж. Р.\_\_\_\_\_\_\_\_**  **Группа \_ИУ7-46Б\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Оценка (баллы) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Преподаватель \_Куров А. В.\_\_\_\_** |  |

Москва.

2020 г.

**Цель работы:** Изучение и программная реализация алгоритма «Плавающего горизонта» построения трехмерных поверхностей.

**Входные данные:**  Список уравнений поверхносте, которые задаются в отдельном модуле.

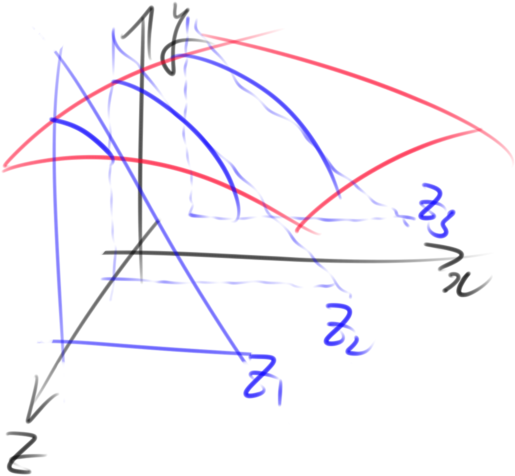
**Выходные данные:** Пользовательское меню, содержащее поля ввода и холст с конечным изображением.

**Результат:**  Должна быть разработана программа, позволяющая осуществлять ввод пределов и шага изменения координат x, z, выбора уравнения поверхности из заранее сформированного списка, построение поверхности. Должен быть обеспечен поворот изображения (поверхности) вокруг каждой из трех координатных осей. Система координат должна быт неподвижной.

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

**Теоретическая часть**

Этот алгоритм предназначен для изображения поверхностей, задаваемых неявным уравнением F(x,y,z)=0. В общем случае считается, что наблюдатель расположен на оси З в бесконечности.

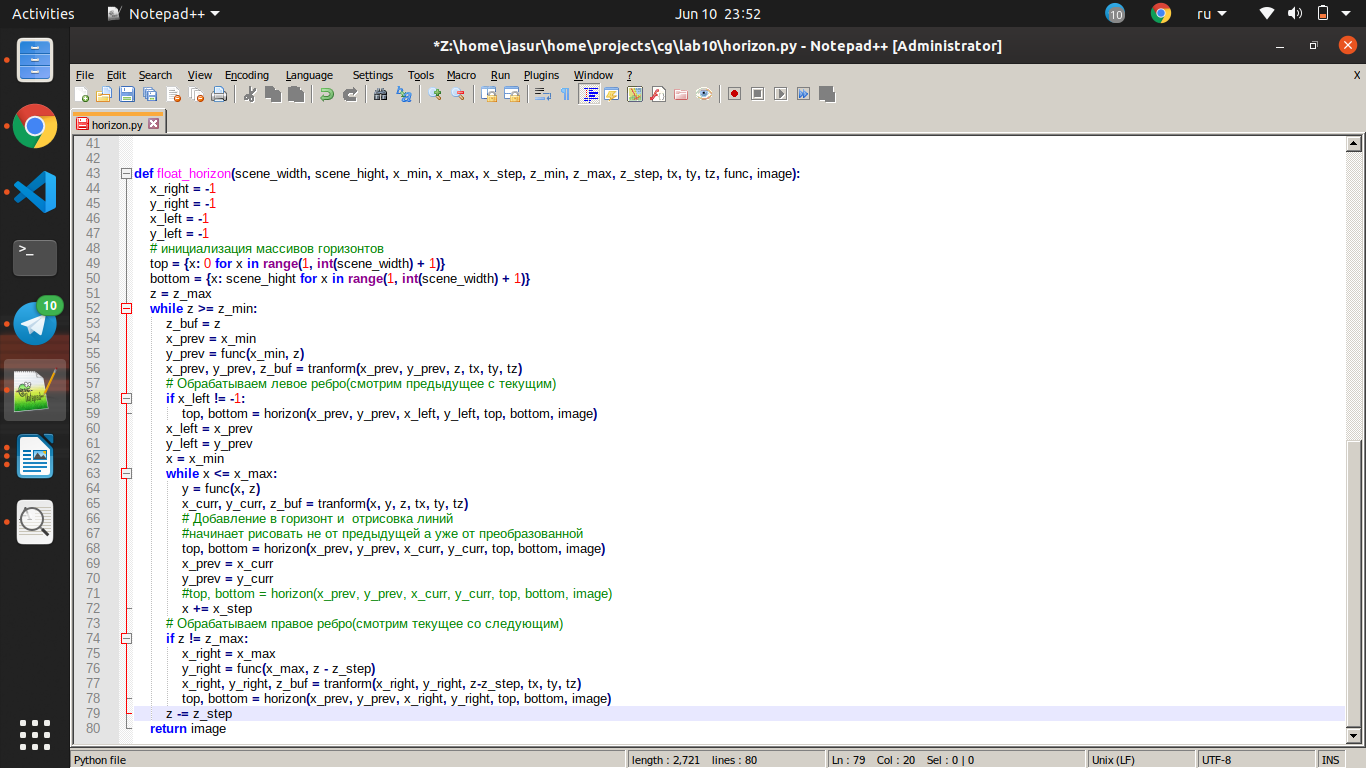


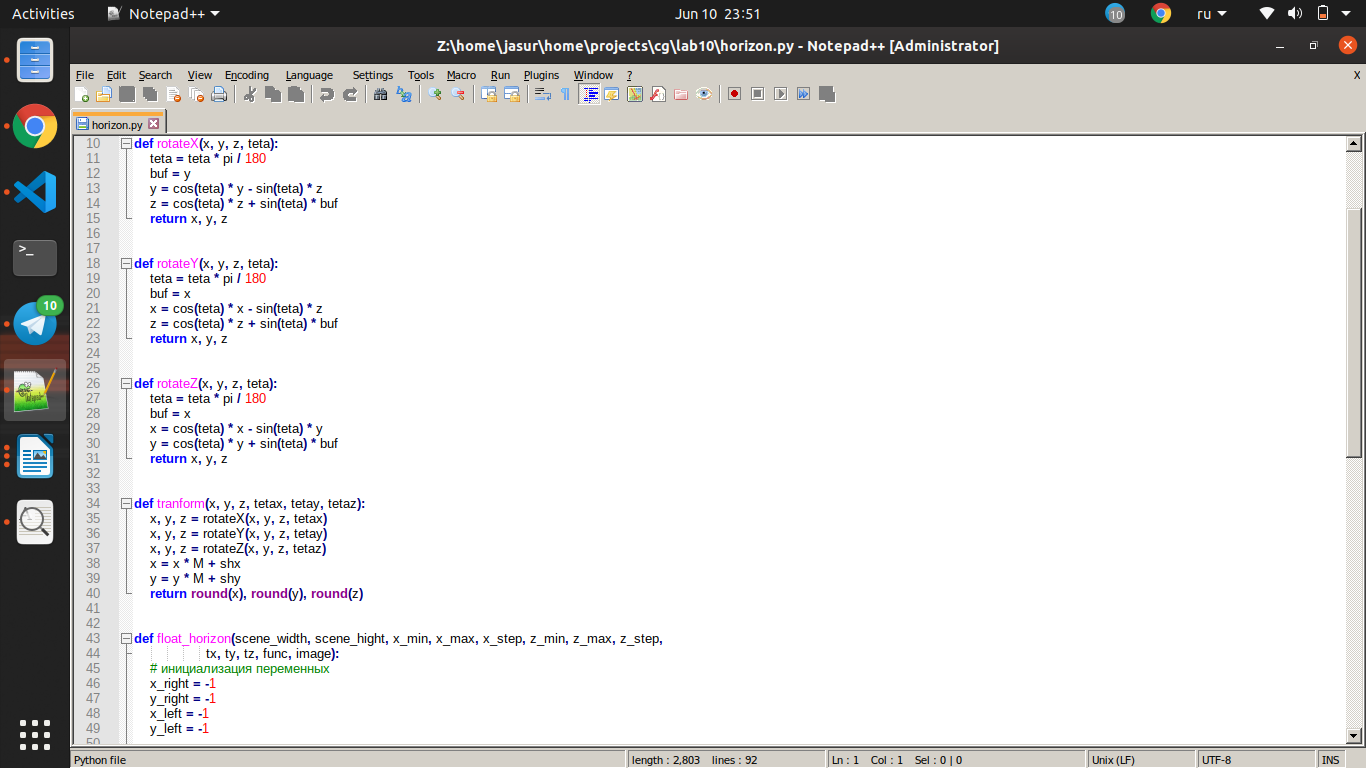
Ближайшая к наблюдателю кривая видима всегда. Далее в каждом сечении имеем уравнение y = f(x, z=const). Для очередной кривой вычисляем Yj(x\_i) – она будет видима, если она расположена выше самой верхней точки кривой j-1, или ниже самой нижней её точки. . Таким образом, нижний горизонт – массив минимальных значений функции, верхний – массив максимальных значений.

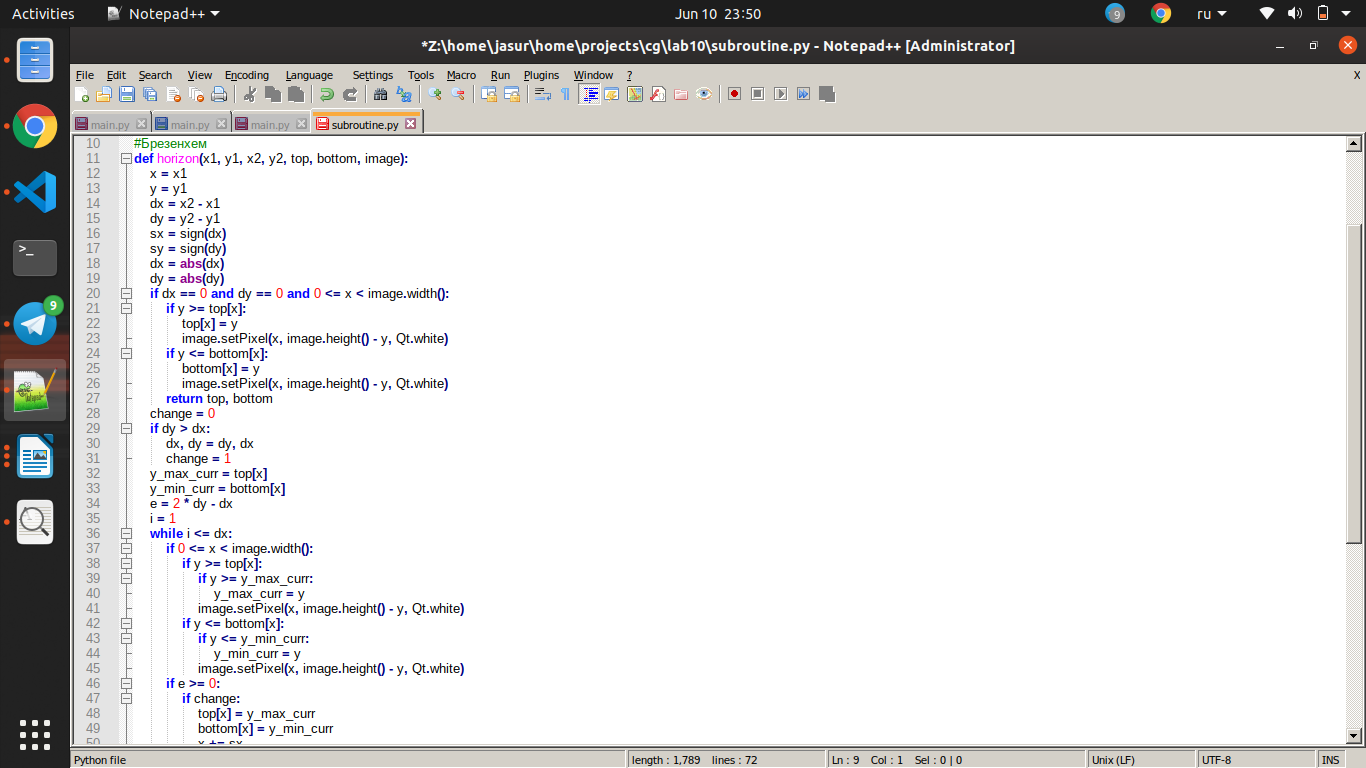
Если мы можем вычислить значение функции в каждой точке, то алгоритм достаточно просто выполняется. Очередная точка текущей кривой является видимой, если её ордината больше текущего максимума (по предыдущим кривым) или меньше текущего минимума.

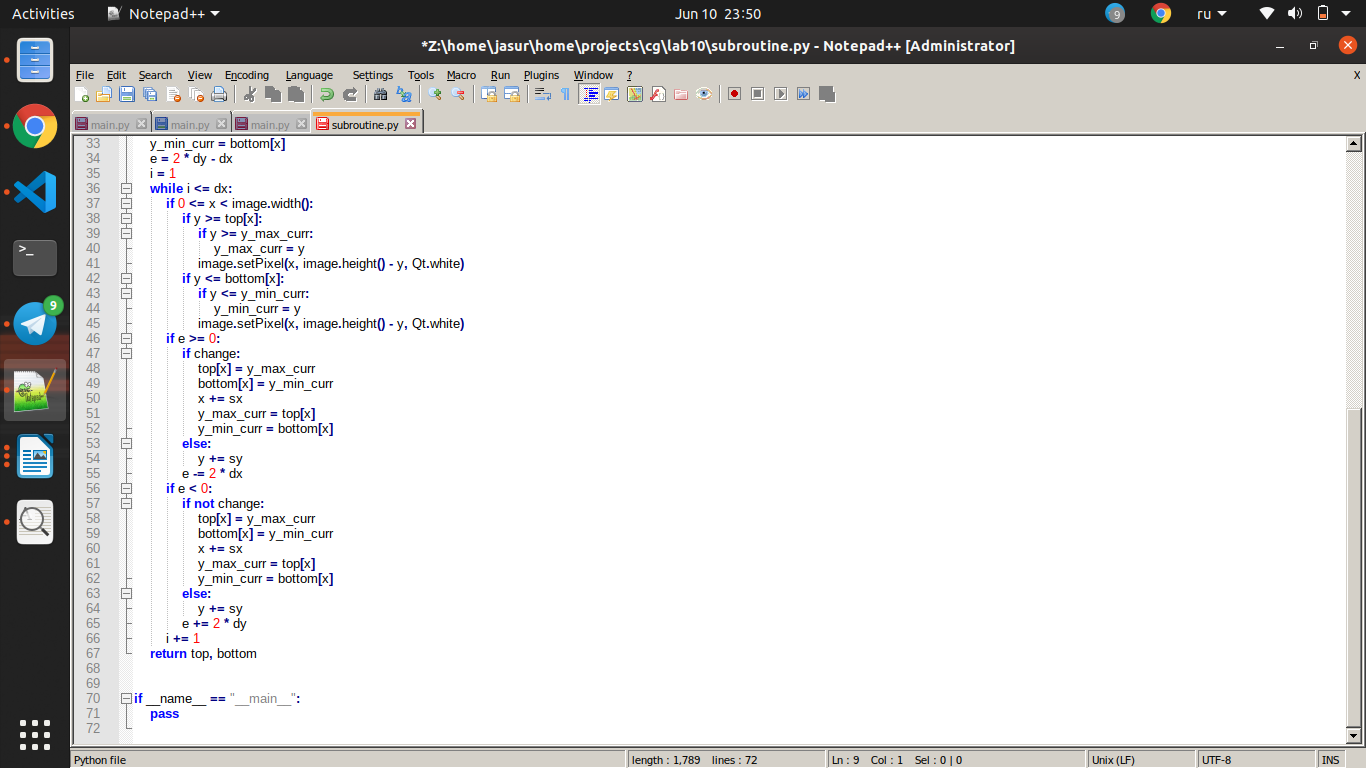
Вычисления могут производиться не с шагом=1 по оХ. Если очередная точка является невидимой, то она не соединяется с предыдущей. Наоборот, если видима, то соединяется. Для упрощения вычислений можно использовать линейную аппроксимации.

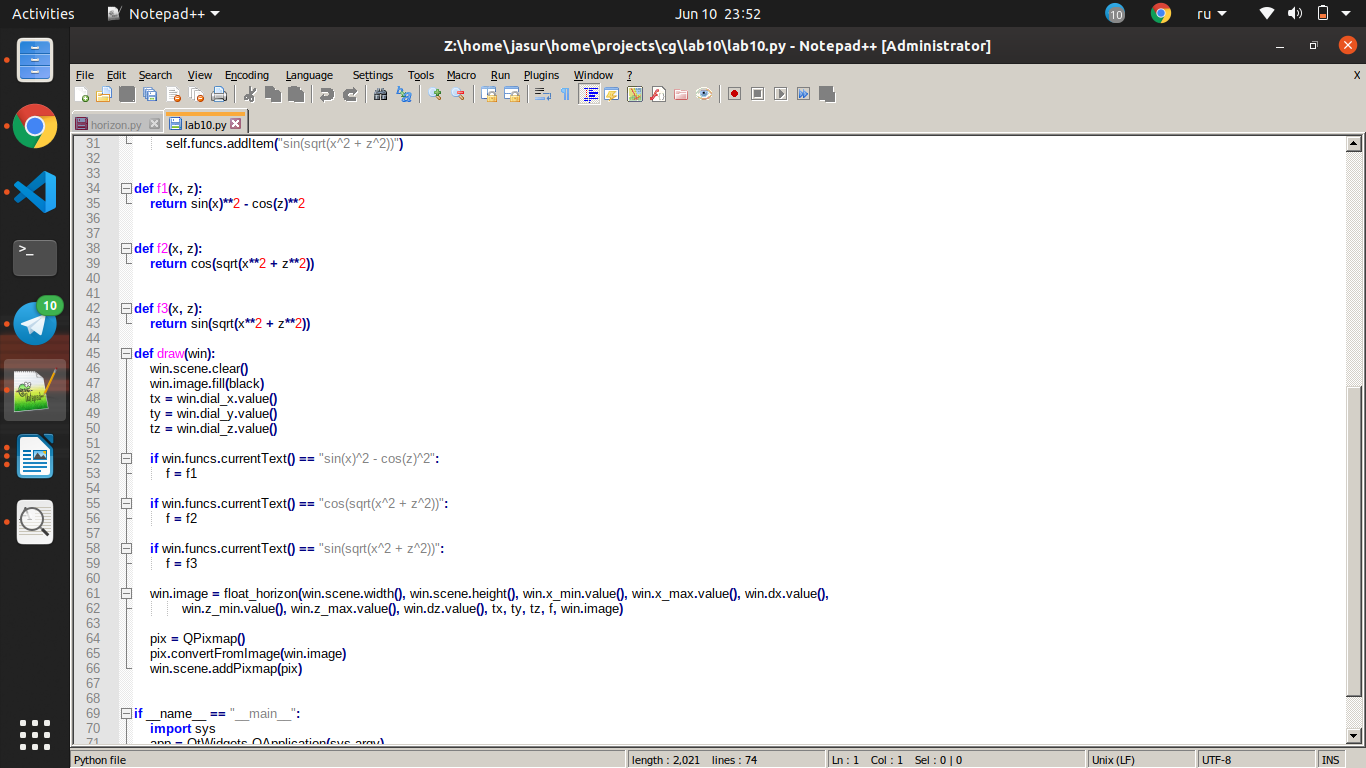
**Код программы**

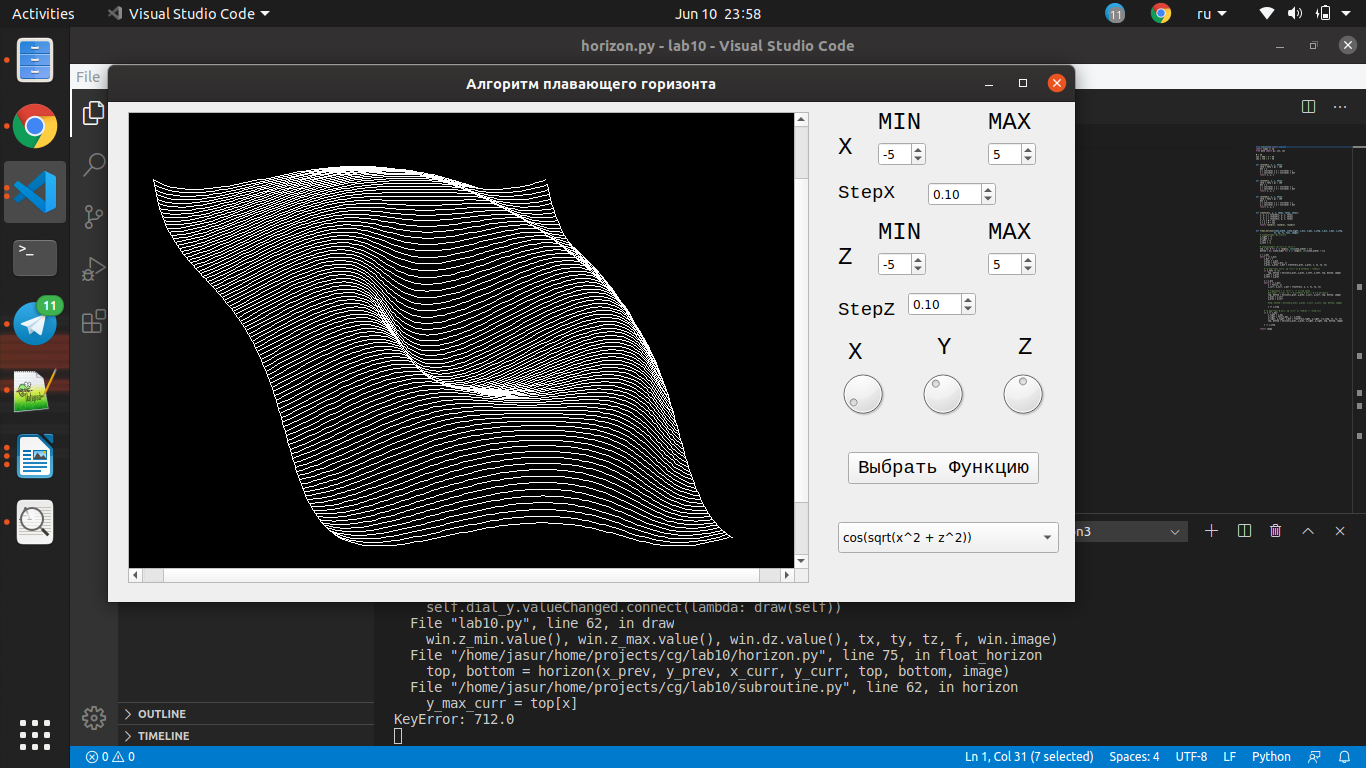
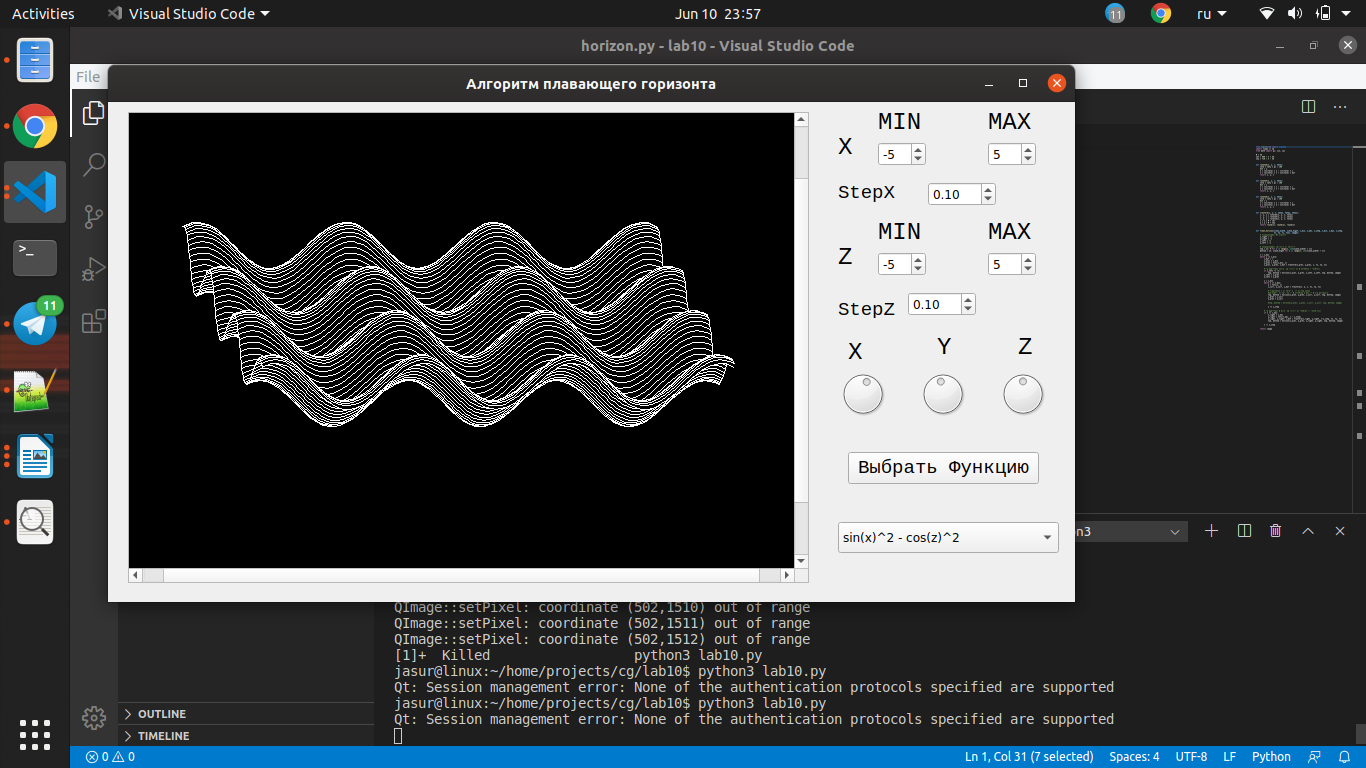
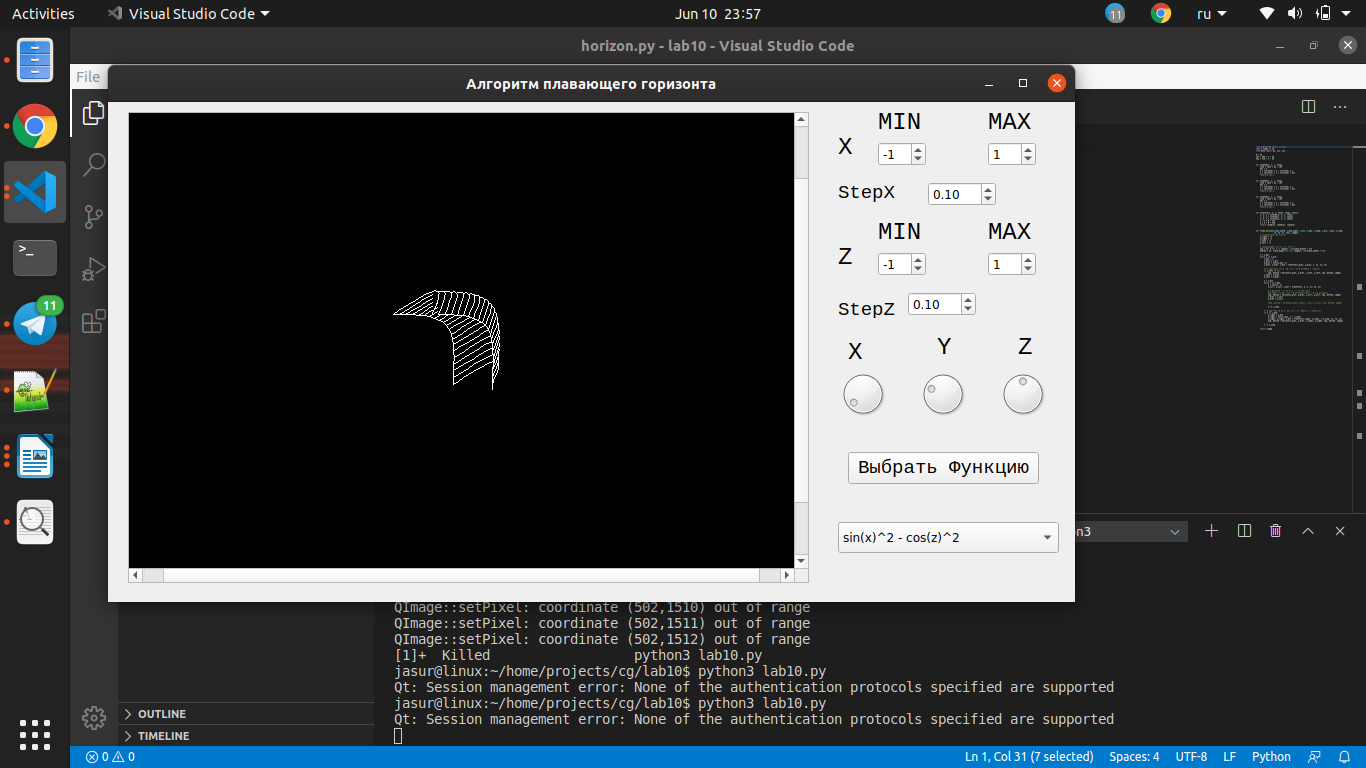
****

****

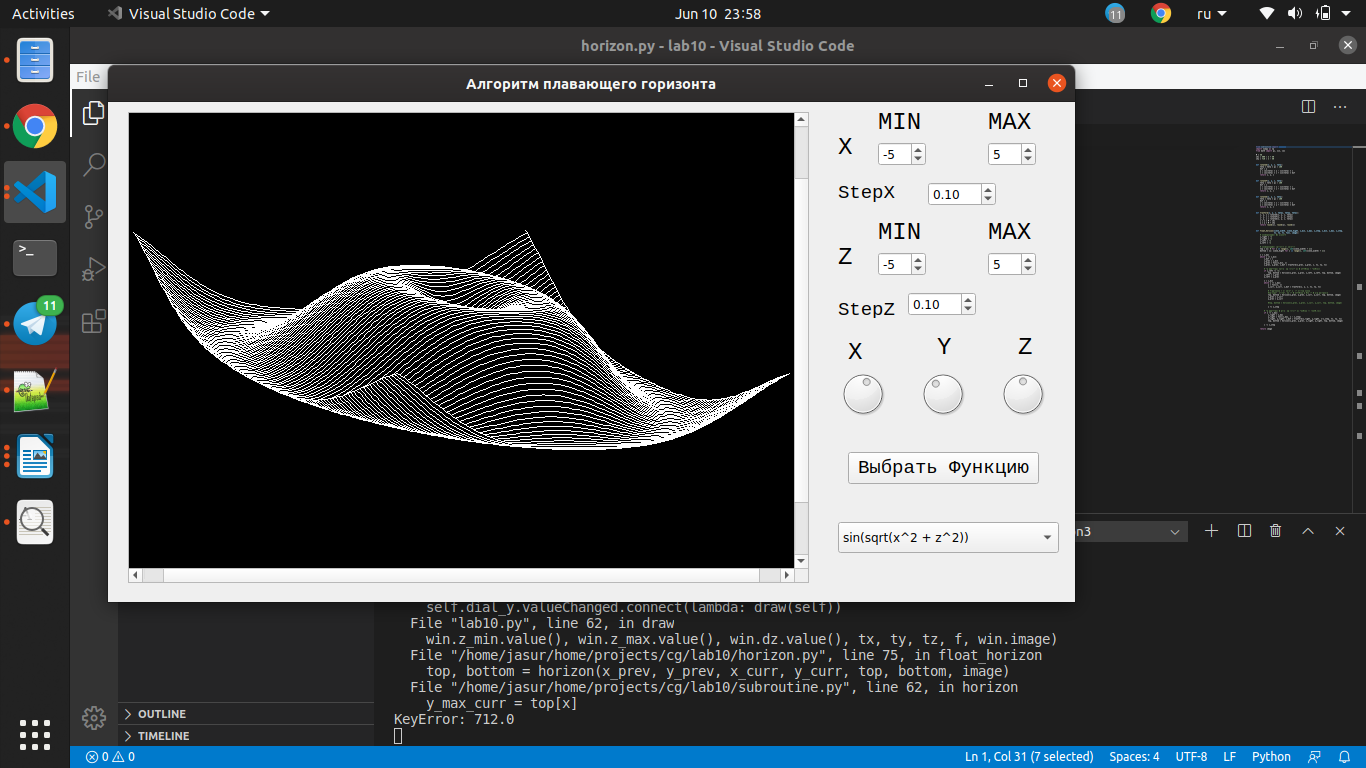
****

****

****

**Пример работы**

**Шаг достаточный маленький**



**Шаг достаточно большой**

