**Министерство науки и высшего образования РФ**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский Авиационный Институт»**

**Национальный Исследовательский Университет**

**Институт** №8 «Информационные технологии и прикладная математика»

**Кафедра** 806 «Вычислительная математика и программирование»

**Лабораторная работа №2**

**по курсу «Компьютерная графика»**

|  |  |
| --- | --- |
| Студент: | Попов И. П. |
| Группа: | М8О-306Б-20 |
| Преподаватель: | Филиппов Г. С. |
| Подпись: |  |
| Оценка: |  |
| Дата: |  |

# Лабораторная работа №2

**Тема:** Каркасная визуализация выпуклого многогранника. Удаление невидимых линий.

**Задание:** Разработать формат представления многогранника и процедуру его каркасной отрисовки в ортографической и изометрической проекциях. Обеспечить удаление невидимых линий и возможность пространственных поворотов и масштабирования многогранника. Обеспечить автоматическое центрирование и изменение размеров изображения при изменении размеров окна. **Вариант:** 16 – гранная прямая правильная пирамида

# Описание

Программа написана на языке программирования Python с использованием библиотек mathplotlib и Poly3DCollection для отрисовки трехмерного графика. Для того, чтобы задать проекцию для отрисовки отображения используются виджеты кнопок Button из mathplotlib.widgets, нажатие на которые поворачивает фигуру нужным образом.

# Исходный код:

''''

Popov Ilya

M80-306Б-20

16 – гранная прямая правильная пирамида

Тема: Каркасная визуализация выпуклого многогранника. Удаление невидимых линий.

Задание: Разработать формат представления многогранника и процедуру его каркасной отрисовки в ортографической и

изометрической проекциях. Обеспечить удаление невидимых линий и возможность пространственных поворотов и

масштабирования многогранника. Обеспечить автоматическое центрирование и изменение размеров изображения при

изменении размеров окна.

'''

import numpy as np

from matplotlib import pyplot as plt

from matplotlib.text import Text

from mpl\_toolkits.mplot3d.art3d import Poly3DCollection

from matplotlib.widgets import Button

fig = plt.figure()

fig.subplots\_adjust(bottom=0.3)

ax = fig.add\_subplot(111, projection='3d')

ax.set\_title(r"16 – гранная прямая правильная пирамида")

N = 15  # количество граней

segm = np.pi \* 2 / N

rotation = 0

radius = 1

#  вершины пирамиды

v = np.array([(np.sin(segm \* i + rotation) \* radius, np.cos(segm \* i + rotation) \* radius, 0)

              for i in range(N)])

top = np.array([0, 0, 0.7])

ax.scatter3D(v[:, 0], v[:, 1], v[:, 2])

ax.set\_zlim(0, 1)

#  грани пирамиды

sides = []

for i in range(1, N):

    sides.append([v[i - 1], v[i], top])

sides.append([v[-1], v[0], top])

# основание пирамиды

sides.append(v)

ax.add\_collection3d(Poly3DCollection(sides, facecolors='blue', linewidths=1, edgecolors='black', alpha=0.5))

#-------------------------

def remove\_func(event):

    ax.add\_collection3d(Poly3DCollection(sides, facecolors='blue', linewidths=1, edgecolors='black', alpha=1))

    plt.draw()

remove\_button\_ax = fig.add\_axes([0.4, 0.15, 0.5, 0.05])

remove\_button = Button(remove\_button\_ax, "Удалить невидимые линии")

remove\_button.on\_clicked(remove\_func)

def show\_func(event):

    ax.add\_collection3d(Poly3DCollection(sides, facecolors='blue', linewidths=1, edgecolors='black', alpha=0.5))

    plt.draw()

show\_button\_ax = fig.add\_axes([0.4, 0.25, 0.5, 0.05])

show\_button = Button(show\_button\_ax, "Показать невидимые линии")

show\_button.on\_clicked(show\_func)

#-------------------------

fig.text(0.1, 0.34, "Проекции:")

def isometric\_func(event):

    ax.view\_init(elev=30)

    plt.draw()

isometric\_button\_ax = fig.add\_axes([0.1, 0.28, 0.23, 0.05])

isometric\_button = Button(isometric\_button\_ax, "Изометрия")

isometric\_button.on\_clicked(isometric\_func)

def top\_func(event):

    ax.view\_init(elev=90)

    plt.draw()

top\_button\_ax = fig.add\_axes([0.1, 0.16, 0.23, 0.05])

top\_button = Button(top\_button\_ax, "Вид сверху")

top\_button.on\_clicked(top\_func)

def front\_func(event):

    ax.view\_init(elev=0)

    plt.draw()

front\_button\_ax = fig.add\_axes([0.1, 0.22, 0.23, 0.05])

front\_button = Button(front\_button\_ax, "Вид спереди")

front\_button.on\_clicked(front\_func)

def bottom\_func(event):

    ax.view\_init(elev=-90)

    plt.draw()

bottom\_button\_ax = fig.add\_axes([0.1, 0.1, 0.23, 0.05])

bottom\_button = Button(bottom\_button\_ax, "Вид снизу")

bottom\_button.on\_clicked(bottom\_func)

ax.grid()

ax.axis()

plt.show()

# Работа программы:

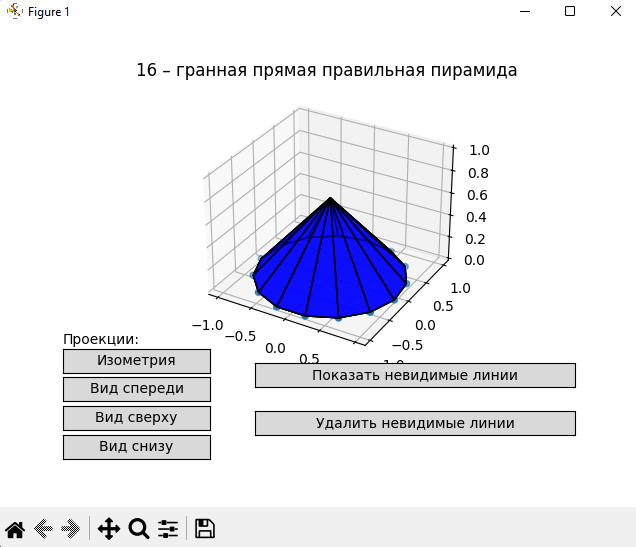
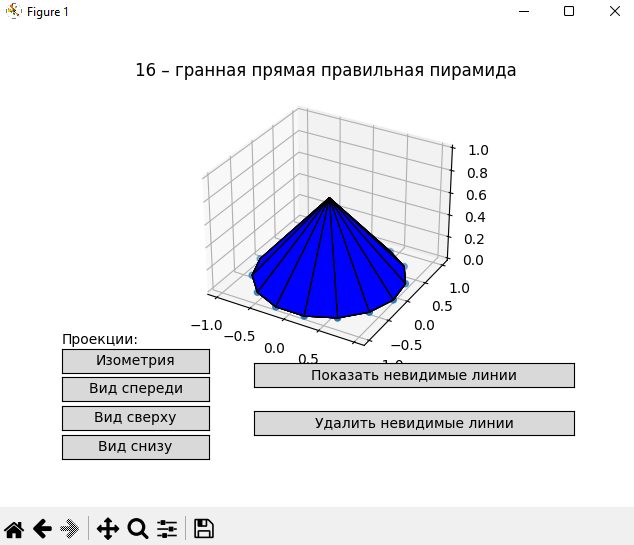
** **

Рис. 1. Невидимые линии отображены Рис. 2. Невидимые линии удалены

1. **Выводы:**

В ходе выполнения данной лабораторной работы была написана программа на языке Python для получения графика каркасной визуализации 16-гранной правильной пирамиды.