**Московский авиационный институт**

**(Национальный исследовательский университет)**

Институт: «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Дисциплина: «Компьютерная графика»

**Лабораторная работа № 2**

Тема: Каркасная визуализация выпуклого многогранника. Удаление невидимых линий.

Студент: Манташев Асадулла Уллубиевич

Группа: 08-305

Преподаватель: Чернышов Л.Н.

Дата:

Оценка:

Москва, 2022

1. Постановка задачи

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.

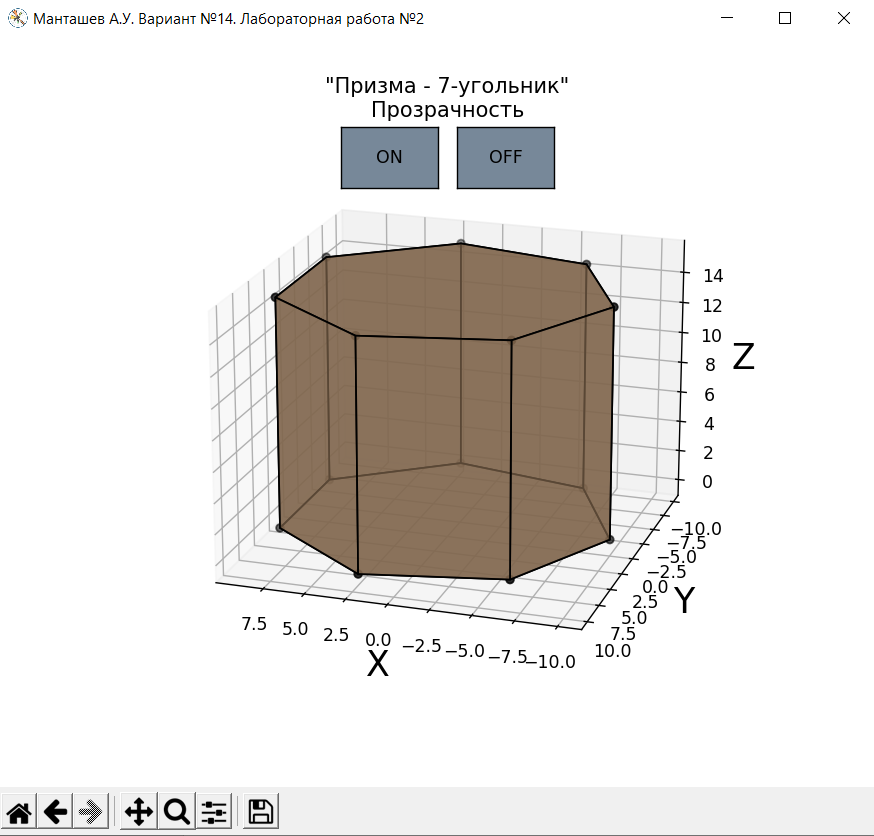
Вариант 14: Прямая призма с основанием - правильный 7-угольник

1. Описание программы

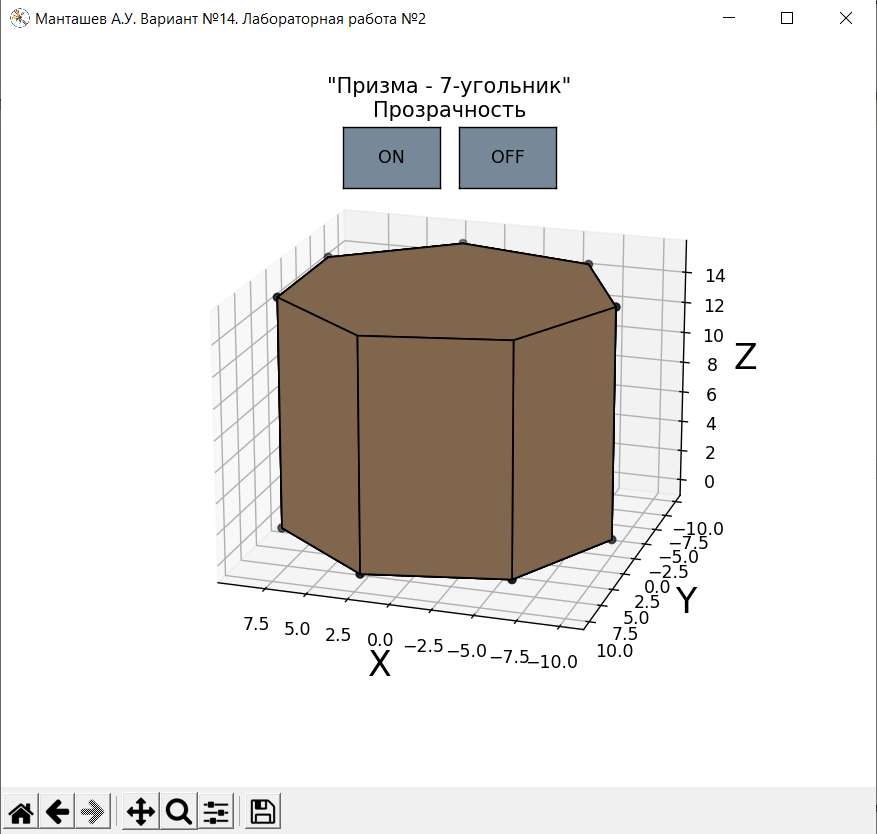
Для реализации программы я использовал python3 с модулями matplotlib с расширением mpl\_toolkits и numpy. Matplotlib необходим для визуализации призмы по точкам в графическом интерфейсе, а numpy необходим для задания функций, по которым строятся точки призмы.

1. Результат работы программы

С мнимыми линиями:



Без мнимых линий:



1. Листинг программы

'''

Манташев А.У. М8О-305Б-20

Лабораторная работа №2.

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 №14:

Прямая призма с основанием - правильный 7-угольник

'''

import numpy as np

import matplotlib.pyplot as plt

from mpl\_toolkits.mplot3d.art3d import Poly3DCollection

from matplotlib.widgets import Button

def TransparentON(event):

axis.add\_collection3d(Poly3DCollection(

frame, facecolors=[0.5, 0.4, 0.3], alpha=0.7, linewidths=1, edgecolors='black'))

def TransparentOFF(event):

axis.add\_collection3d(Poly3DCollection(

frame, facecolors=[0.5, 0.4, 0.3], alpha=1, linewidths=1, edgecolors='black'))

fig = plt.figure('Манташев А.У. Вариант №14. Лабораторная работа №2', figsize=(7, 6))

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

buttonON = fig.add\_subplot(863)

btnON = Button(buttonON, 'ON')

btnON.on\_clicked(TransparentON)

btnON.color = '#F00'

buttonOFF = fig.add\_subplot(864)

btnOFF = Button(buttonOFF, 'OFF')

btnOFF.on\_clicked(TransparentOFF)

btnOFF.color = '#778899'

xC = 0

yC = 0

zC = 0

R = 10

H = 15

N = 7

p = np.array([

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 0)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 0)), zC],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 1)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 1)), zC],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 2)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 2)), zC],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 3)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 3)), zC],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 4)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 4)), zC],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 5)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 5)), zC],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 6)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 6)), zC],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 0)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 0)), zC + H],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 1)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 1)), zC + H],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 2)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 2)), zC + H],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 3)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 3)), zC + H],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 4)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 4)), zC + H],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 5)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 5)), zC + H],

[xC + R \* np.cos(np.pi/N \* (1 + 2 \* 6)), yC + R \* np.sin(np.pi/N \* (1 + 2 \* 6)), zC + H]

])

frame = [

[p[0], p[1], p[2], p[3], p[4], p[5], p[6]],

[p[7], p[8], p[9], p[10], p[11], p[12], p[13]],

[p[0], p[1], p[8], p[7]],

[p[1], p[2], p[9], p[8]],

[p[2], p[3], p[10], p[9]],

[p[3], p[4], p[11], p[10]],

[p[4], p[5], p[12], p[11]],

[p[5], p[6], p[13], p[12]],

[p[6], p[0], p[7], p[13]],

]

axis.scatter3D(p[:, 0], p[:, 1], p[:, 2], color='black')

axis.add\_collection3d(Poly3DCollection(

frame, facecolors=[0.5, 0.4, 0.3], alpha=0.7, linewidths=1, edgecolors='black'))

axis.set\_xlabel('X', fontsize=20, color='black')

axis.set\_ylabel('Y', fontsize=20, color='black')

axis.set\_zlabel('Z', fontsize=20, color='black')

axis.set\_title('\"Призма - 7-угольник\"\nПрозрачность')

plt.show()

ЛИТЕРАТУРА

1. Документация Matplotlib [Электронный ресурс]. URL- <https://matplotlib.org/stable/tutorials/index.html> (дата обращения: 02.11.2022).
2. Руководство по numpy [Электронный ресурс]. URL- <https://numpy.org/doc/stable/reference/generated/numpy.array.html> (дата обращения: 02.11.2022).
3. Призма [Электронный ресурс]. URL- [https://ru.wikipedia.org/wiki/Призма\_(геометрия)](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%B7%D0%BC%D0%B0_(%D0%B3%D0%B5%D0%BE%D0%BC%D0%B5%D1%82%D1%80%D0%B8%D1%8F)) (дата обращения: 02.11.2022).
4. Правильный многоугольник [Электронный ресурс]. URL- [https://ru.wikipedia.org/wiki/Правильный\_многоугольник](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B0%D0%B2%D0%B8%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D0%BC%D0%BD%D0%BE%D0%B3%D0%BE%D1%83%D0%B3%D0%BE%D0%BB%D1%8C%D0%BD%D0%B8%D0%BA) (дата обращения: 02.11.2022).