

**Московский авиационный институт  
(Национальный исследовательский университет)**

Институт: «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Дисциплина: «Компьютерная графика»

**Лабораторная работа № 3**

**Тема: Основы построения фотореалистичных  
изображений**

Студент: Попов Матвей

Группа: 08-308Б

Преподаватель: Чернышов Л.Н.

Дата:

Оценка:

Москва, 2022

## 1. Постановка задачи

Используя результаты Л.Р. №2, аппроксимировать заданное тело выпуклым многогранником. Точность аппроксимации задается пользователем. Обеспечить возможность вращения и масштабирования многогранника и удаление невидимых линий и поверхностей. Реализовать простую модель закраски для случая одного источника света. Параметры освещения и отражающие свойства материала задаются пользователем в диалоговом режиме.

**Вариант 4:** Полушарие

## 2. Описание программы

В программе есть текстовое поле для ввода параметра точности аппроксимации (по умолчанию 40). Этот параметр задаёт количество граней в многограннике, лежащего в основании полушария. Координаты вершин этого многогранника вычисляются каждый раз, когда изменяется точность аппроксимации.

**Установка зависимостей (ввести в консоли):**

```
pip install numpy  
pip install matplotlib
```

**Запуск программы:**

```
python main.py
```

## 3. Результаты выполнения тестов

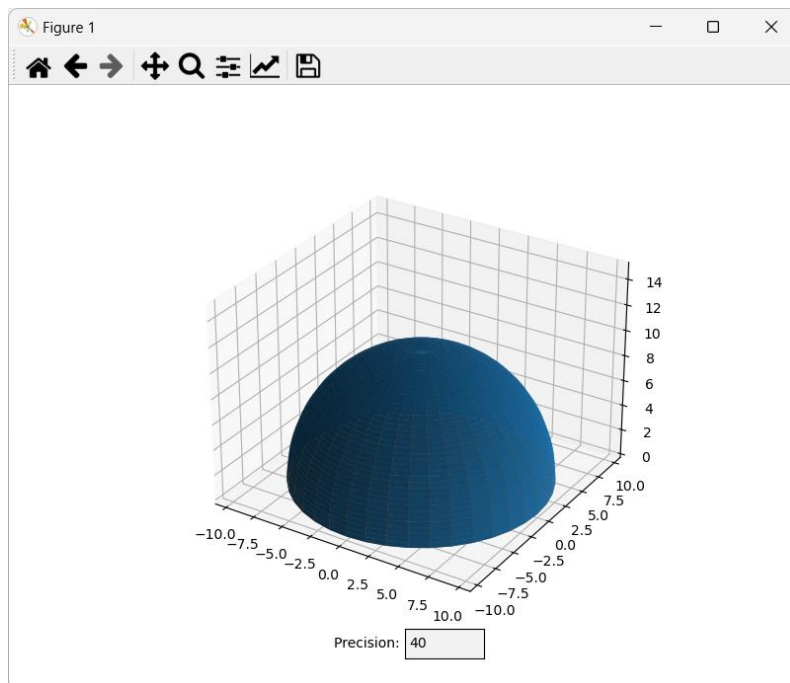


Рис. 1. Точность аппроксимации 40

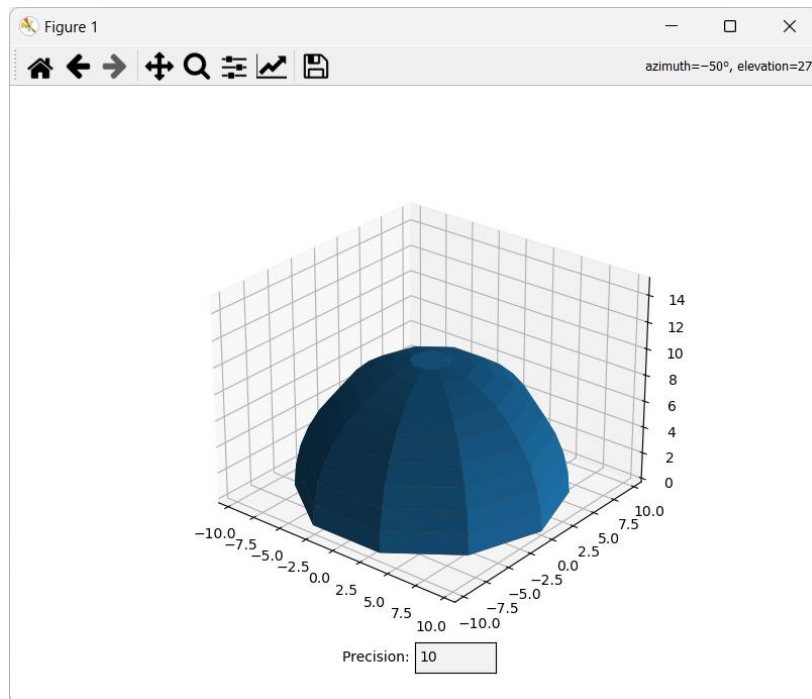


Рис. 2. Точность аппроксимации 10

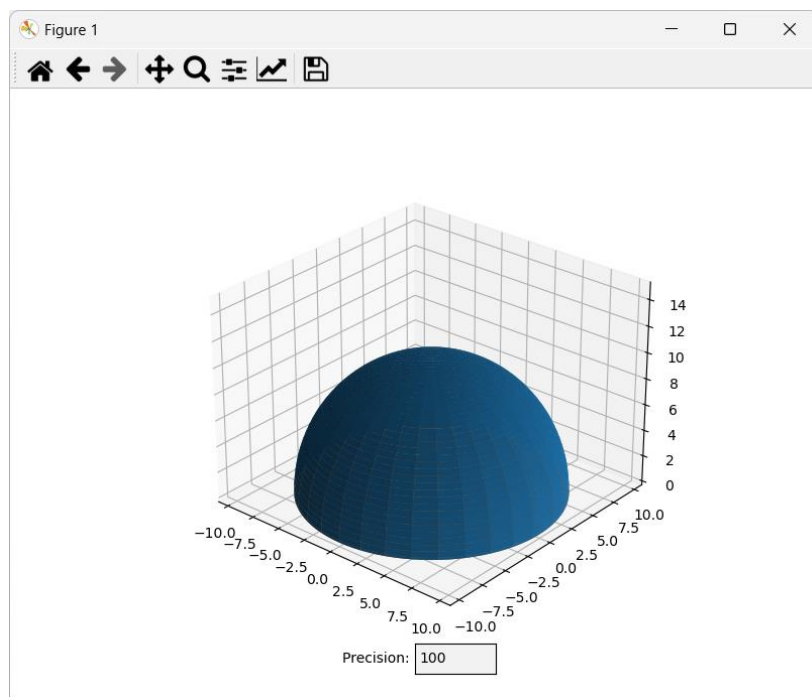


Рис. 3. Точность аппроксимации 100

#### 4. Листинг программы

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.widgets import TextBox

def PlotHalfSphere(radius, precision, transparency):
    precision += 1
    phi = []
    theta = []
```

```

pointTheta = 0
pointPhi = 0
dTheta = (0.5 * pi) / (precision - 1)
dPhi = (2 * pi) / (precision - 1)
tmp = []
for _ in range(precision):
    tmp.append(pointTheta)
theta.append(tmp)
for _ in range(precision - 1):
    pointTheta = pointTheta + dTheta
    tmp = []
    for _ in range(precision):
        tmp.append(pointTheta)
    theta.append(tmp)
tmp = []
tmp.append(pointPhi)
for _ in range(precision - 1):
    pointPhi = pointPhi + dPhi
    tmp.append(pointPhi)
for _ in range(precision):
    phi.append(tmp)
x = radius * np.sin(theta) * np.cos(phi)
y = radius * np.sin(theta) * np.sin(phi)
z = radius * np.cos(theta)
axis.plot_surface(x, y, z, alpha=transparency)
z = np.zeros((precision, precision))
axis.plot_surface(x, y, z, alpha=transparency)
plt.show()

def AxisInstallation():
    axis.set_xlim([-radius - 0.5, radius + 0.5])
    axis.set_ylim([-radius - 0.5, radius + 0.5])
    axis.set_zlim([0, 1.5 * radius])

def ChangePrecision(input):
    axis.clear()
    AxisInstallation()
    PlotHalfSphere(radius, int(input), transparency)

pi = np.pi
radius = 10
transparency = 1

f = plt.figure()
axis = f.add_subplot(111, projection='3d')
AxisInstallation()
precisionField = plt.axes([0.5, 0.05, 0.1, 0.05])
precisionTextBox = TextBox(precisionField, 'Precision: ', '40')
precisionTextBox.on_submit(ChangePrecision)
PlotHalfSphere(radius, 40, transparency)

```

## ЛИТЕРАТУРА

1. Numpy documentation. [Электронный ресурс]  
URL: <https://numpy.org/> (дата обращения: 23.10.2022)
2. Matplotlib tutorials. [Электронный ресурс]  
URL: <https://matplotlib.org/> (дата обращения: 23.10.2022)
3. Аппроксимация экспериментальных данных. [Электронный ресурс]  
URL: [http://lab314.brsu.by/Geo2/labrab/lab\\_10/AprD.htm](http://lab314.brsu.by/Geo2/labrab/lab_10/AprD.htm) (дата обращения 23.10.2022)