

**Міністерство освіти і науки України  
Національний технічний університет України «КПІ» імені Ігоря Сікорського  
Кафедра обчислювальної техніки ФІОТ**

**ЗВІТ  
з лабораторної роботи №3  
з навчальної дисципліни «Computer Vision»**

**Тема:**

**ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ ПОБУДОВИ ТА ПЕРЕТВОРЕННЯ  
КООРДИНАТ ПРОСТОРОВИХ (3D) ОБ'ЄКТІВ**

**Виконала:**

Студентка 3 курсу  
Навчальної групи ІС-12  
Мельникова К.О.

**Перевірив:**

Професор кафедри ОТ ФІОТ  
Писарчук О.О.

**Київ 2024**

## Мета роботи:

Виявити дослідити та узагальнити особливості реалізації алгоритмів формування та обробки векторних цифрових зображень на прикладі застосування алгоритмів інтерполяції, апроксимації та згладжування складних 3D растрових об'єктів та застосування технологій видалення невидимих граней та ребер.

## Варіант:

28	Відображення 3D фігури реалізується з використанням аксонометричної проекції будь-якого типу. Обрати самостійно: бібліотеку, розмір графічного вікна, розмір фігури, динаміку зміни положення фігури, кольорову гамму графічного об'єкту. Всі операції перетворень мають здійснюватися у межах графічного вікна.	Піраміда з чотирикутною основою. Метод інтерполяції: поліном Лагранжа. Метод видалення невидимих ліній та поверхонь: алгоритм плаваючого обрію.
----	---	---

## Завдання I рівня – максимально 7 балів.

Здійснити виконання завдання лабораторної роботи із застосуванням алгоритму інтерполяції для побудови векторного зображення 2D, 3D графічного об'єкту.

## Математична модель

Інтерполяційний поліном Лагранжа має вигляд:

$$L_n(x) = \sum_{i=0}^n c_i(x) f(x_i), \quad f(x_i) = y_i,$$
$$c_i(x) = \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)} = \prod_{\substack{k=0 \\ k \neq i}}^n \frac{(x-x_k)}{(x_i-x_k)},$$
$$L_n(x) = \sum_{i=0}^n y_i \prod_{\substack{k=0 \\ k \neq i}}^n \frac{(x-x_k)}{(x_i-x_k)},$$
$$L_n(x) = y_0 \frac{(x-x_1)(x-x_2)\dots(x-x_n)}{(x_0-x_1)(x_0-x_2)\dots(x_0-x_n)} + y_1 \frac{(x-x_0)(x-x_2)\dots(x-x_n)}{(x_1-x_0)(x_1-x_2)\dots(x_1-x_n)} + \dots$$

Даний алгоритм побудований на використанні дробно-раціональної функції для відтворення кривизни реальної дискретної послідовності значень.

## Програмна реалізація

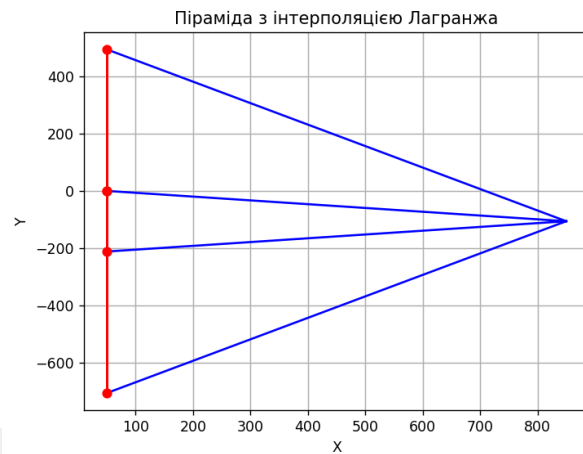
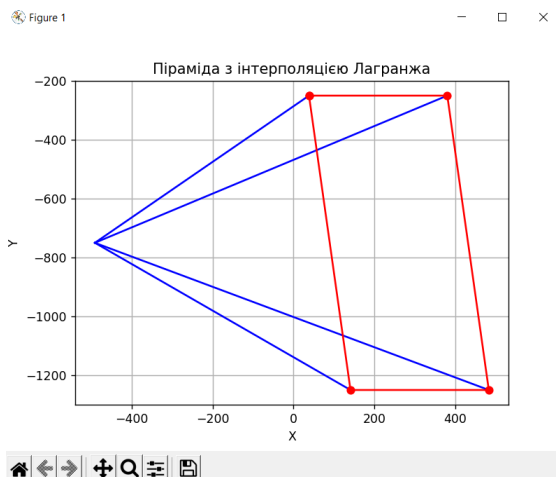
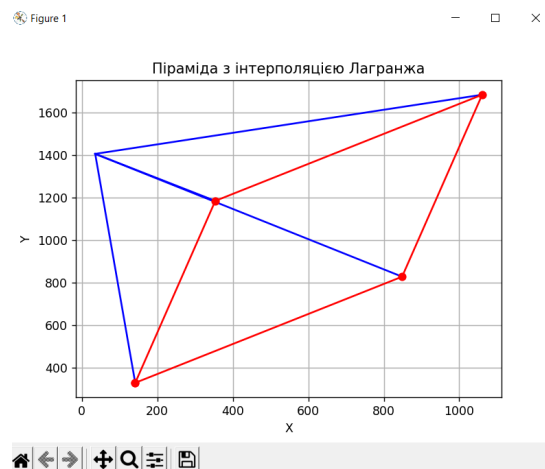
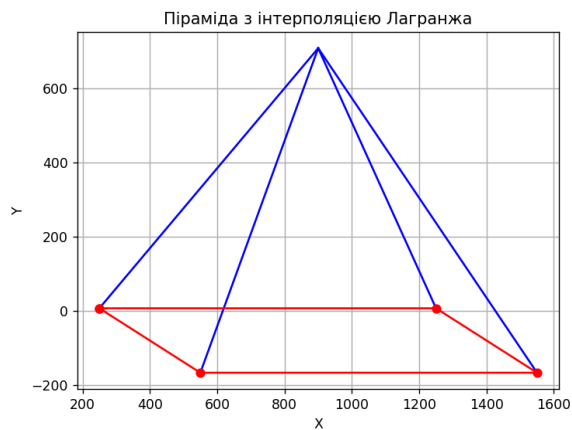
```
lab3.py x
1 import numpy as np
2 import matplotlib.pyplot as plt
3 from scipy.interpolate import lagrange
4
5 # Координати піраміди з чотирикутною основою
6 st, height, offset = 1000, 800, 300
7 Pyramid = np.array([
8     [0, 0, 0, 1],
9     [st, 0, 0, 1],
10    [st + offset, st, 0, 1],
11    [0 + offset, st, 0, 1],
12    [st / 2 + offset / 2, st / 2, height, 1] # Апекс піраміди
13 ])
14
15 # Функція проєкції на xy
16 def ProjectXY(Figure):
17     f = np.array([[1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 0, 0], [0, 0, 0, 1]])
18     return Figure.dot(f.T)
19
20 # Зміщення
21 def ShiftXYZ(Figure, l, m, n):
22     f = np.array([[1, 0, 0, l], [0, 1, 0, m], [0, 0, 1, n], [0, 0, 0, 1]])
23     return Figure.dot(f.T)
24
```

```
lab3.py x
25 # Аксонометрія
26 def dimetri(Figure, TetaG1, TetaG2):
27     TetaR1, TetaR2 = np.radians(TetaG1), np.radians(TetaG2)
28     f1 = np.array([[np.cos(TetaR1), 0, -np.sin(TetaR1), 0], [0, 1, 0, 0], [np.sin(TetaR1), 0, np.cos(TetaR1), 1], [0, 0, 0, 0]])
29     f2 = np.array([[1, 0, 0, 0], [0, np.cos(TetaR2), np.sin(TetaR2), 0], [0, -np.sin(TetaR2), np.cos(TetaR2), 0], [0, 0, 0, 1]])
30     return Figure.dot(f1.T).dot(f2.T)
31
32 # Візуалізація піраміди з інтерполяцією Лагранжа
33 def PyramidWiz(Prxy):
34     base_vertices = Prxy[:4]
35     apex = Prxy[4]
36     vertices = np.vstack((base_vertices, base_vertices[0])) # Замикаємо основу
37
38     # Побудова сторін піраміди з інтерполяцією Лагранжа
39     for i in range(4):
40         x = [base_vertices[i, 0], apex[0]]
41         y = [base_vertices[i, 1], apex[1]]
42         poly = lagrange(x, y)
43         x_new = np.linspace(x[0], x[1], 100)
44         y_new = poly(x_new)
45         plt.plot(x_new, y_new, 'b-')
46
47     # Побудова основи піраміди
48     plt.plot(vertices[:, 0], vertices[:, 1], 'ro-')
49
50     plt.xlabel('X')
51     plt.ylabel('Y')
52     plt.title('Піраміда з інтерполяцією Лагранжа')
53     plt.grid(True)
54     plt.show()
55
```

Активация Windows

```
xw, yw, st, TetaG1, TetaG2 = 600, 600, 50, 0, 100
l, m, n = (xw / 2) - st, (yw / 2) - st, st
Pyramid1 = ShiftXYZ(Pyramid, l, m, n)
Pyramid2 = dimetri(Pyramid1, TetaG1, TetaG2)
Prxy3 = ProjectXY(Pyramid2)
PyramidWiz(Prxy3)
```

## Результат виконання:



## **Висновок**

Під час виконання даної лабораторної роботи було розроблено програмне забезпечення для створення векторних зображень 2D та 3D графічних об'єктів з використанням алгоритму інтерполяції Лагранжа. Застосований підхід дозволяє побудувати зображення піраміди з чотирикутною основою з високою деталізацією та гладкістю за допомогою розробленого програмного забезпечення