Министерство просвещения Российской Федерации федеральное государственное образовательное учреждение высшего учреждения высшего образования « Алтайский государственный педагогический университет»

Институт информационных технологий и физико-математического образования

Кафедра теоретических основ информатики

Отчёт по «Практикум по решению предметных задач»

## Лабораторная работа: **Алгоритмы и методы графических построений**

Выполнил студент 3217д группы

Карамышева Екатерина Сергеевна

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

Руководитель:

Москаленко Е.В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

Дата сдачи

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2022г.

Дата защиты

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2022г.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_

Барнаул 2024

**Вариант 6**

Нарисовать правильный пятиугольник, если радиус описанной окружности около данного пятиугольника равен 50. На каждой из сторон данного пятиугольника построить правильный четырехугольник (при необходимости координаты вершин рассчитываются).

Код:

import numpy as np

import matplotlib.pyplot as plt

# Функция для вычисления координат вершин правильного многоугольника

def polygon\_vertices(sides, radius):

angles = np.linspace(0, 2 \* np.pi, sides, endpoint=False)

return radius \* np.cos(angles), radius \* np.sin(angles)

# Функция для рисования правильного четырехугольника

def draw\_square(ax, center, side\_length):

half\_side = side\_length / 2

square = np.array([

[center[0] - half\_side, center[1] - half\_side],

[center[0] + half\_side, center[1] - half\_side],

[center[0] + half\_side, center[1] + half\_side],

[center[0] - half\_side, center[1] + half\_side],

[center[0] - half\_side, center[1] - half\_side] # Замыкаем квадрат

])

ax.plot(square[:, 0], square[:, 1], 'b')

# Параметры

radius = 50

sides\_polygon = 5

# Вычисляем координаты вершин пятиугольника

x\_polygon, y\_polygon = polygon\_vertices(sides\_polygon, radius)

# Создаем график

fig, ax = plt.subplots()

ax.plot(np.append(x\_polygon, x\_polygon[0]), np.append(y\_polygon, y\_polygon[0]), 'r', label='Пятиугольник') # Замыкаем пятиугольник

# Рисуем четырехугольники на каждой стороне пятиугольника

for i in range(sides\_polygon):

# Находим середину стороны

x\_mid = (x\_polygon[i] + x\_polygon[(i + 1) % sides\_polygon]) / 2

y\_mid = (y\_polygon[i] + y\_polygon[(i + 1) % sides\_polygon]) / 2

# Длина стороны пятиугольника

side\_length = np.sqrt((x\_polygon[i] - x\_polygon[(i + 1) % sides\_polygon])\*\*2 +

(y\_polygon[i] - y\_polygon[(i + 1) % sides\_polygon])\*\*2)

# Рисуем квадрат

draw\_square(ax, (x\_mid, y\_mid), side\_length)

# Настройки графика

ax.set\_aspect('equal')

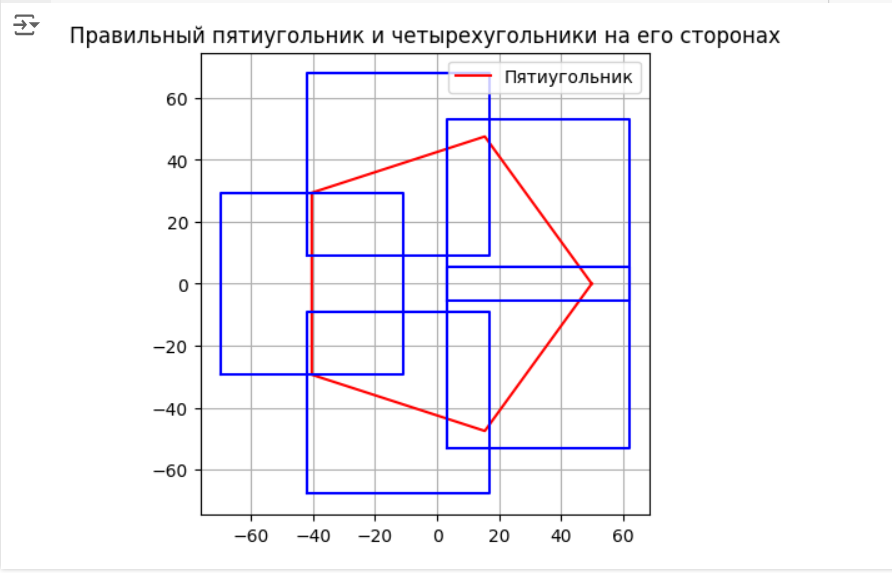
ax.set\_title('Правильный пятиугольник и четырехугольники на его сторонах')

ax.legend()

plt.grid()

plt.show()

Результат тестирования:еР

****

**Ссылка на код:**

**https://colab.research.google.com/drive/1MSbjOPRZiKRkZvhbA4SupGs4aT2HZyOV?usp=sharing**