**Минобрнауки РФ**

**ЮЗГУ**

**ФФиПИ**

Кафедра программной инженерии

**ЛАБОРАТОРНАЯ РАБОТА №7**

**ПРОГРАММА ДЛЯ ФОРМИРОВАНИЯ ИЗОБРАЖЕНИЙ**

Выполнил: ст. гр. ПО-22б

Алешин К.Р.

Проверил: Ефремов В. В.

Курск, 2024 г

**Задание:**  


**Листинг программы:**import numpy as np

import matplotlib.pyplot as plt

from PIL import Image, ImageDraw

def scale\_and\_rotate\_image(image, size, target\_width, target\_height, rotation\_angle=45):

"""Масштабирует, поворачивает и позиционирует изображение."""

width, height = size

cx, cy = width // 2, height // 2 # Центр изображения

# Масштабируем изображение до прямоугольника

resized\_image = image.resize((target\_width, target\_height), resample=Image.Resampling.LANCZOS)

# Поворачиваем изображение на заданный угол

rotated\_image = resized\_image.rotate(rotation\_angle, expand=True)

# Смещение: центрируем верхний край изображения в центре координат

rotated\_width, rotated\_height = rotated\_image.size

position = (cx - rotated\_width // 2, cy - rotated\_height // 4)

# Создаем новый холст с белым фоном и размещаем изображение

new\_image = Image.new("RGBA", size, "white")

new\_image.paste(rotated\_image, position, rotated\_image if rotated\_image.mode == "RGBA" else None)

return new\_image

def draw\_axes(image, color="black"):

"""Рисует оси координат на изображении."""

draw = ImageDraw.Draw(image)

width, height = image.size

cx, cy = width // 2, height // 2

# Горизонтальная ось

draw.line([(0, cy), (width, cy)], fill=color, width=2)

# Вертикальная ось

draw.line([(cx, 0), (cx, height)], fill=color, width=2)

def draw\_cosine\_function(image, color="red"):

"""Рисует график cos(x) на изображении."""

draw = ImageDraw.Draw(image)

width, height = image.size

# Генерируем точки для графика

x\_values = np.linspace(0, width, 1000)

y\_values = height // 2 - (np.cos((x\_values / width) \* 4 \* np.pi) \* (height // 4)) # Масштабирование cos(x)

# Соединяем точки

points = [(x, y) for x, y in zip(x\_values, y\_values)]

draw.line(points, fill=color, width=2)

def main():

# Размер итогового изображения

size = (400, 400)

# Задаём размеры прямоугольника (например, 80x40 пикселей)

target\_width, target\_height = 80, 40

# Загружаем изображение

input\_path = "input.jpg"

try:

original\_image = Image.open(input\_path).convert("RGBA")

except FileNotFoundError:

print("Файл изображения не найден. Пожалуйста, загрузите файл.")

return

# Обрабатываем изображение

final\_image = scale\_and\_rotate\_image(original\_image, size, target\_width, target\_height)

# Рисуем оси координат

draw\_axes(final\_image)

# Рисуем график cos(x)

draw\_cosine\_function(final\_image)

# Отображаем результат

plt.imshow(final\_image)

plt.axis("off")

plt.show()

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Пример работы программы:**

**Изображение выглядит как График, снимок экрана, диаграмма, линия

Автоматически созданное описание**