МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ  
ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

**институт информационных технологий и технологического образования  
кафедра информационных технологий и электронного обучения**

Основная профессиональная образовательная программа  
Направление подготовки 09.03.01 Информатика и вычислительная техника  
Направленность (профиль) «Технологии разработки программного обеспечения»  
форма обучения – очная

**Практическая работа 2**по дисциплине «Математические основы компьютерной графики»

Обучающегося 3 курса.

Собинина Егора Яковлевича

Санкт-Петербург

2022

1.1 Напишите программу, которая по материалам лекции преобразовывает концы отрезка

по заданной матрице преобразования

Затем рассчитайте середину начального отрезка и с помощью матрицы преобразования найдите середину нового отрезка. Отрезки и середины отрезков прорисуйте на экране с помощью библиотеки pygame.

Ответ:

import sys

import numpy as np

import pprint as pprn

import pygame

from pygame.locals import \*

# цвета

COLOR\_RED = (255, 0, 0)

COLOR\_GREEN = (0, 255, 0)

COLOR\_BLUE = (0, 0, 255)

COLOR\_DARKBLUE = (0, 0, 139)

COLOR\_DARKGREEN = (0, 139, 0)

COLOR\_DARKRED = (139, 0, 0)

# инициализация

pygame.init()

pygame.font.init()

window = pygame.display.set\_mode((1500, 1200))

window.fill((255, 255, 255))

my\_game\_font = pygame.font.SysFont('Comic Sans MS', 20)

pygame.display.update()

# смещение

cood\_shift = 50

# координаты

L = np.array([

    [0, 100],

    [200, 300],

])

print(f"Массив координат отрезков L =")

pprn.pprint(L)

# матрица преобразования

T = np.array([

    [1, 2],

    [3, 1],

])

L\_new = L@T

print('Новые координаты точки L\_new =')

pprn.pprint(L\_new)

L\_shifted = np.copy(L) + cood\_shift

# отрисовка начальной фигуры

for i in range(0, 1, 1):

    color = COLOR\_DARKBLUE if i == 0 else COLOR\_BLUE

    line\_width = 6 if i == 0 else 3

    pygame.draw.line(window, color, L\_shifted[0 + i], L\_shifted[1 + i], line\_width)

    pygame.draw.circle(window, color, (L\_shifted[0 + i] + L\_shifted[1 + i]) / 2, line\_width)

    window.blit(

        my\_game\_font.render(f"L\_1({L\_shifted[0 + i, 0]:.3f}, {L\_shifted[0 + i, 1]:.3f})", True, COLOR\_DARKBLUE), (L\_shifted[0 + i, :] - 30)

    )

    window.blit(

        my\_game\_font.render(f"L\_2({L\_shifted[1 + i, 0]:.3f}, {L\_shifted[1 + i, 1]:.3f})", True, COLOR\_DARKBLUE), (L\_shifted[1 + i, :] + 5)

    )

L\_new\_shifted = np.copy(L\_new) + cood\_shift

# отрисовка измененной фигуры

for i in range(0, 1, 1):

    color = COLOR\_DARKRED if i == 0 else COLOR\_RED

    line\_width = 6 if i == 0 else 3

    pygame.draw.line(window, color, L\_new\_shifted[0 + i, :], L\_new\_shifted[1 + i, :], line\_width)

    pygame.draw.circle(window, color, (L\_new\_shifted[0 + i] + L\_new\_shifted[1 + i]) / 2, line\_width)

    window.blit(

        my\_game\_font.render(f"L\_1\_new({L\_new\_shifted[0 + i, 0]:.3f}, {L\_new\_shifted[0 + i, 1]:.3f})", True, COLOR\_DARKRED), (L\_new\_shifted[0 + i, :] - 30)

    )

    window.blit(

        my\_game\_font.render(f"L\_2\_new({L\_new\_shifted[1 + i, 0]:.3f}, {L\_new\_shifted[1 + i, 1]:.3f})", True, COLOR\_DARKRED), (L\_new\_shifted[1 + i, :] + 5)

    )

pygame.display.update()

# запуск отрисовки

FPS = 30

clock = pygame.time.Clock()

while True:

    for event in pygame.event.get():

        if event.type == QUIT:

            pygame.quit()

            sys.exit()

    clock.tick(FPS)

    pygame.display.update()

1.2 Напишите программу, которая по материалам лекции преобразовывает параллельные отрезки заданные матрицей (первые две строки - координаты первого отрезка, вторые две координаты второго)

по заданной матрице преобразования

Затем рассчитайте начальный наклон отрезков и конечный наклон отрезков с дополнительной проверке по формулам из лекции. Прорисуйте всё на экране с помощью библиотеки pygame.

Ответ:

import sys

import numpy as np

import pprint as pprn

import pygame

from pygame.locals import \*

# цвета

COLOR\_RED = (255, 0, 0)

COLOR\_GREEN = (0, 255, 0)

COLOR\_BLUE = (0, 0, 255)

COLOR\_DARKBLUE = (0, 0, 139)

COLOR\_DARKGREEN = (0, 139, 0)

COLOR\_DARKRED = (139, 0, 0)

# инициализация

pygame.init()

pygame.font.init()

window = pygame.display.set\_mode((1500, 1200))

window.fill((255, 255, 255))

my\_game\_font = pygame.font.SysFont('Comic Sans MS', 20)

pygame.display.update()

# смещение

cood\_shift = 50

# координаты

L = np.array([

    [50, 100],

    [250, 200],

    [50, 200],

    [250, 300]

])

print(f"Массив координат отрезков L =")

pprn.pprint(L)

# матрица преобразований

T = np.array([

    [1, 2],

    [3, 1],

])

L\_shifted = np.copy(L) + cood\_shift

print(f'Координаты точек со смещением относительно левой грани')

pprn.pprint(L\_shifted)

L\_new = L@T

print('Новые координаты точки L\_new =')

pprn.pprint(L\_new)

L\_new\_shifted = np.copy(L\_new) + cood\_shift

print(f'Координаты новых точек со смещением относительно левой грани')

pprn.pprint(L\_new\_shifted)

# отрисовка первой фигуры

for i in range(0, 3, 2):

    color = COLOR\_DARKBLUE if i == 0 else COLOR\_BLUE

    line\_width = 6 if i == 0 else 3

    pygame.draw.line(window, color, L\_shifted[0 + i], L\_shifted[1 + i], line\_width)

    window.blit(

        my\_game\_font.render(f"L\_1({L\_shifted[0 + i, 0]:.3f}, {L\_shifted[0 + i, 1]:.3f})", True, COLOR\_DARKBLUE), (L\_shifted[0 + i, :] - 30)

    )

    window.blit(

        my\_game\_font.render(f"L\_2({L\_shifted[1 + i, 0]:.3f}, {L\_shifted[1 + i, 1]:.3f})", True, COLOR\_DARKBLUE), (L\_shifted[1 + i, :] + 5)

    )

# отрисовка второй фигуры

for i in range(0, 3, 2):

    color = COLOR\_DARKBLUE if i == 0 else COLOR\_BLUE

    line\_width = 6 if i == 0 else 3

    pygame.draw.line(window, color, L\_new\_shifted[0 + i], L\_new\_shifted[1 + i], line\_width)

    window.blit(

        my\_game\_font.render(f"L\_1({L\_new\_shifted[0 + i, 0]:.3f}, {L\_new\_shifted[0 + i, 1]:.3f})", True, COLOR\_DARKBLUE), (L\_new\_shifted[0 + i, :] - 30)

    )

    window.blit(

        my\_game\_font.render(f"L\_2({L\_new\_shifted[1 + i, 0]:.3f}, {L\_new\_shifted[1 + i, 1]:.3f})", True, COLOR\_DARKBLUE), (L\_new\_shifted[1 + i, :] + 5)

    )

for i in range(0, 1, 1):

    scale1 = (L[i+1][i+1] - L[i][i+1])/(L[i+1][0]-L[i][i])

    arc1 = np.arctan(scale1)

    scale2 = (L\_new[i+1][i+1] - L\_new[i][i+1])/(L\_new[i+1][0]-L\_new[i][i])

    arc2 = np.arctan(scale2)

print(f'Угло наклона начального отрезка к оси Х (меньший) ≈ {round(arc1\*180/np.pi)}°')

print(f'Угло наклона измененного отрезка к оси Х (меньший) ≈ {round(arc2\*180/np.pi)}°')

pygame.display.update()

# запуск отрисовки

FPS = 30

clock = pygame.time.Clock()

while True:

    for event in pygame.event.get():

        if event.type == QUIT:

            pygame.quit()

            sys.exit()

    clock.tick(FPS)

    pygame.display.update()

1.3 Напишите программу, которая по материалам лекции преобразовывает пересекающиеся отрезки заданные матрицей (первые две строки - координаты первого отрезка, вторые две координаты второго)

по заданной матрице преобразования

Предварительно умножьте L на 100 и перед каждой отрисовкой на экране сместите отрезки на 100-200 пикселей в положительных направлениях по осям OX OY на экране компьютера для лучшего отображения. Прорисуйте всё на экране с помощью библиотеки pygame.

Ответ:

import sys

import numpy as np

import pprint as pprn

import pygame

from pygame.locals import \*

# цвета

COLOR\_RED = (255, 0, 0)

COLOR\_GREEN = (0, 255, 0)

COLOR\_BLUE = (0, 0, 255)

COLOR\_DARKBLUE = (0, 0, 139)

COLOR\_DARKGREEN = (0, 139, 0)

COLOR\_DARKRED = (139, 0, 0)

# инициализация

pygame.init()

pygame.font.init()

window = pygame.display.set\_mode((1000, 1000))

window.fill((255, 255, 255))

my\_game\_font = pygame.font.SysFont('Comic Sans MS', 20)

pygame.display.update()

# смещение

cood\_shift = 300

scale = 50

# координаты

L = np.array([

    [-1/2, 3/2],

    [3, -2],

    [-1, -1],

    [3, 5/3]

])

L = L \* scale

print(f"Массив координат отрезков L =")

pprn.pprint(L)

# матрица преобразования

T = np.array([

    [1, 2],

    [1, -3],

])

L\_shifted = np.copy(L) + cood\_shift

print(f'Координаты точек со смещением относительно левой грани')

pprn.pprint(L\_shifted)

L\_new = L@T

print('Новые координаты точки L\_new =')

pprn.pprint(L\_new)

L\_new\_shifted = np.copy(L\_new) + cood\_shift

print(f'Координаты новых точек со смещением относительно левой грани')

pprn.pprint(L\_new\_shifted)

# отрисовка первой фигуры

for i in range(0, 3, 2):

    color = COLOR\_DARKBLUE if i == 0 else COLOR\_BLUE

    line\_width = 6 if i == 0 else 3

    pygame.draw.line(window, color, L\_shifted[0 + i], L\_shifted[1 + i], line\_width)

    print(L\_shifted[0 + i], L\_shifted[1 + i])

    window.blit(

        my\_game\_font.render(f"L\_1({L\_shifted[0 + i, 0]:.3f}, {L\_shifted[0 + i, 1]:.3f})", True, COLOR\_DARKBLUE), (L\_shifted[0 + i, :] - 30)

    )

    window.blit(

        my\_game\_font.render(f"L\_2({L\_shifted[1 + i, 0]:.3f}, {L\_shifted[1 + i, 1]:.3f})", True, COLOR\_DARKBLUE), (L\_shifted[1 + i, :] + 5)

    )

# отрисовка второй фигуры

for i in range(0, 3, 2):

    color = COLOR\_DARKBLUE if i == 0 else COLOR\_BLUE

    line\_width = 6 if i == 0 else 3

    pygame.draw.line(window, color, L\_new\_shifted[0 + i], L\_new\_shifted[1 + i], line\_width)

    window.blit(

        my\_game\_font.render(f"L\_1({L\_new\_shifted[0 + i, 0]:.3f}, {L\_new\_shifted[0 + i, 1]:.3f})", True, COLOR\_DARKBLUE), (L\_new\_shifted[0 + i, :] - 30)

    )

    window.blit(

        my\_game\_font.render(f"L\_2({L\_new\_shifted[1 + i, 0]:.3f}, {L\_new\_shifted[1 + i, 1]:.3f})", True, COLOR\_DARKBLUE), (L\_new\_shifted[1 + i, :] + 5)

    )

pygame.display.update()

# запуск отрисовки

FPS = 30

clock = pygame.time.Clock()

while True:

    for event in pygame.event.get():

        if event.type == QUIT:

            pygame.quit()

            sys.exit()

    clock.tick(FPS)

    pygame.display.update()

1.4 Напишите программу, которая по материалам лекции вращает фигуру треугольник на 90 градусов против часовой стрелки относительно начала координат

по заданной матрице преобразования

Предварительно умножьте L на 100 и перед каждой отрисовкой на экране сместите все координаты на 100-200 пикселей в положительных направлениях по осям OX OY на экране компьютера для лучшего отображения. Прорисуйте всё на экране с помощью библиотеки pygame.

Ответ:

import sys

import numpy as np

import pprint as pprn

import pygame

from pygame.locals import \*

# цвета

COLOR\_RED = (255, 0, 0)

COLOR\_GREEN = (0, 255, 0)

COLOR\_BLUE = (0, 0, 255)

COLOR\_DARKBLUE = (0, 0, 139)

COLOR\_DARKGREEN = (0, 139, 0)

COLOR\_DARKRED = (139, 0, 0)

# инициализация

pygame.init()

pygame.font.init()

window = pygame.display.set\_mode((1000, 1000))

window.fill((255, 255, 255))

my\_game\_font = pygame.font.SysFont('Comic Sans MS', 20)

pygame.display.update()

# Смещение

cood\_shift = 200

# координаты фигуры

L = np.array([

    [3, -1],

    [4, 1],

    [2, 1]

]) \* 100

print(f"Массив координат отрезков L =")

pprn.pprint(L)

L\_shifted = np.copy(L) + cood\_shift

# Отрисовываем начальную фигуру

pygame.draw.lines(window, COLOR\_DARKBLUE, True, L\_shifted, 6)

pygame.display.update()

# матрица преобразования

T = np.array([

    [0, 1],

    [-1, 0]

])

print('Матрица T =')

pprn.pprint(T)

# Преобразование координат отрезков L

L\_new = L@T

print('Новые координаты точки L\_new =')

pprn.pprint(L\_new)

L\_new\_shifted = np.copy(L\_new) + cood\_shift

# Отрисовываем новую фигуру

pygame.draw.lines(window, COLOR\_DARKRED, True, L\_new\_shifted, 6)

pygame.display.update()

# отрисовка

FPS = 30

clock = pygame.time.Clock()

while True:

    for event in pygame.event.get():

        if event.type == QUIT:

            pygame.quit()

            sys.exit()

    clock.tick(FPS)

    pygame.display.update()

1.5 Напишите программу, которая по материалам лекции отражает фигуру треугольник относительно прямой y = x

по заданной матрице преобразования

Предварительно умножьте L на 100 и перед каждой отрисовкой на экране сместите все координаты на 100-200 пикселей в положительных направлениях по осям OX OY на экране компьютера для лучшего отображения. Прорисуйте всё на экране с помощью библиотеки pygame.

Ответ:

import sys

import numpy as np

import pprint as pprn

import pygame

from pygame.locals import \*

# цвет

COLOR\_RED = (255, 0, 0)

COLOR\_GREEN = (0, 255, 0)

COLOR\_BLUE = (0, 0, 255)

COLOR\_DARKBLUE = (0, 0, 139)

COLOR\_DARKGREEN = (0, 139, 0)

COLOR\_DARKRED = (139, 0, 0)

# инициализация

pygame.init()

pygame.font.init()

window = pygame.display.set\_mode((1400, 1200))

window.fill((255, 255, 255))

my\_game\_font = pygame.font.SysFont('Comic Sans MS', 20)

pygame.display.update()

# Смещение

cood\_shift = 200

# координаты

L = np.array([

    [8, 1],

    [7, 3],

    [6, 2]

]) \* 100

print(f"Массив координат отрезков L =")

pprn.pprint(L)

L\_shifted = np.copy(L) + cood\_shift

# Отрисовываем начальную фигуру

pygame.draw.lines(window, COLOR\_DARKBLUE, True, L\_shifted, 6)

pygame.display.update()

# y = x

T = np.array([

    [0, 1],

    [1, 0]

])

print('Матрица T =')

pprn.pprint(T)

# Преобразование координат отрезков L

L\_new = L@T

print('Новые координаты точки L\_new =')

pprn.pprint(L\_new)

L\_new\_shifted = np.copy(L\_new) + cood\_shift

# Отрисовываем новую фигуру

pygame.draw.lines(window, COLOR\_DARKRED, True, L\_new\_shifted, 6)

pygame.display.update()

# запуск отрисовки

FPS = 30

clock = pygame.time.Clock()

while True:

    for event in pygame.event.get():

        if event.type == QUIT:

            pygame.quit()

            sys.exit()

    clock.tick(FPS)

    pygame.display.update()

1.6 Напишите программу, которая по материалам лекции масштабирует со смещением фигуру треугольник, заданный матрицей

по заданной матрице преобразования

Предварительно умножьте L на 100 и перед каждой отрисовкой на экране сместите все координаты на 100-200 пикселей в положительных направлениях по осям OX OY на экране компьютера для лучшего отображения. Прорисуйте всё на экране с помощью библиотеки pygame.

Ответ:

import sys

import numpy as np

import pprint as pprn

import pygame

from pygame.locals import \*

# цвета

COLOR\_RED = (255, 0, 0)

COLOR\_GREEN = (0, 255, 0)

COLOR\_BLUE = (0, 0, 255)

COLOR\_DARKBLUE = (0, 0, 139)

COLOR\_DARKGREEN = (0, 139, 0)

COLOR\_DARKRED = (139, 0, 0)

# инициализация

pygame.init()

pygame.font.init()

window = pygame.display.set\_mode((1400, 1200))

window.fill((255, 255, 255))

my\_game\_font = pygame.font.SysFont('Comic Sans MS', 20)

pygame.display.update()

# Смещение

cood\_shift = 200

# координаты

L = np.array([

    [5, 1],

    [5, 2],

    [3, 2]

]) \* 100

print(f"Массив координат отрезков L =")

pprn.pprint(L)

L\_shifted = np.copy(L) + cood\_shift

# Отрисовываем начальную фигуру

pygame.draw.lines(window, COLOR\_DARKBLUE, True, L\_shifted, 6)

pygame.display.update()

# матрица преобразования для масштабирования

T = np.array([

    [2, 0],

    [0, 2]

])

print('Матрица T =')

pprn.pprint(T)

# Преобразование координат отрезков L

L\_new = L@T

print('Новые координаты точки L\_new =')

pprn.pprint(L\_new)

L\_new\_shifted = np.copy(L\_new) + cood\_shift

# Отрисовываем новую фигуру

pygame.draw.lines(window, COLOR\_DARKRED, True, L\_new\_shifted, 6)

pygame.display.update()

# запуск отрисовки

FPS = 30

clock = pygame.time.Clock()

while True:

    for event in pygame.event.get():

        if event.type == QUIT:

            pygame.quit()

            sys.exit()

    clock.tick(FPS)

    pygame.display.update()