



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ 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

ОТЧЁТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №1

По дисциплине: Математические основы компьютерной
графики

Выполнил студент 3 курса группы 1.1

Щербинин А. В.

Санкт-Петербург
2022

Постановка задачи:

1. Комплект 1:

- 1.1. Установите Python последней версии и менеджер пакетов PIP одним из предложенных в Интернете способов.
- 1.2. Установите среду разработки PyCharm или среду VSCode с соответствующими плагинами для Python.
- 1.3. С помощью менеджера пакетов PIP установите библиотеки numpy, pygame и pprint.

2. Комплект 2:

- 2.1. Напишите программу, которая позволяет ввести с клавиатуры координаты точки на плоскости, а затем подвергает эти координаты матричному преобразованию с помощью матрицы 2x2. Результатом являются координаты новой точки на плоскости. Распечатать введенные координаты, матрицу преобразования и новые координаты. Матрица преобразования координат дана в виде:

$$T = \begin{pmatrix} 1 & 3 \\ 4 & 1 \end{pmatrix}$$

- 2.2. Изучить прорисовку графических примитивов из библиотеки pygame - окружности, линии и текста. Нарисовать эти примитивы в окне программы разными цветами.
- 2.3. Напишите программу, которая позволяет ввести с клавиатуры координаты точки на плоскости, а затем подвергает эти координаты матричному преобразованию с помощью матрицы 2x2. Результатом являются координаты новой точки на плоскости. Распечатать введенные координаты, матрицу преобразования и новые координаты. Матрица преобразования координат дана в виде:

$$T = \begin{pmatrix} 1 & 3 \\ 4 & 1 \end{pmatrix}$$

Прорисовать изначальную точку и точку с новыми координатами в виде двух небольших окружностей разными цветами. Рядом с точками напечатать соответствующими цветами значения координат точек в игровом окне программы.

- 2.4. Напишите программу, которая позволяет ввести с клавиатуры координаты двух точек на плоскости, записывает их как вектор-строки в единую матрицу отрезка (линии) на плоскости, а затем подвергает эти координаты линии матричному преобразованию с помощью матрицы 2x2. Результатом являются координаты новой линии на плоскости. Распечатать введенные координаты в виде матрицы координат двух точек, матрицу преобразования и матрицу координат новой линии. Матрица преобразования координат дана в виде:

$$T = \begin{pmatrix} 1 & 3 \\ 4 & 1 \end{pmatrix}$$

Прорисовать изначальную линию и линию с новыми координатами разными цветами с помощью pygame. Рядом с концами линий напечатать соответствующими цветами значения координат точек в игровом окне программы.

3. Комплект 3:

- 3.1.** Подберите 3–5 ссылок на Интернет-материалы, посвященные установке языка программирования Python, менеджера пакетов PIP и библиотеки pygame на ваш компьютер. Составьте к каждой ссылке краткую аннотацию о содержании каждого прочитанного материала.
- 3.2.** Подберите 3 ссылки на Интернет-материалы, посвященные основам использования библиотеки программирования игр pygame. Составьте к каждой ссылке краткую аннотацию о содержании каждого прочитанного материала.

Комплект 1: Установка среды программирования и разработки

Были успешно установлены: Python; менеджер пакетов PIP; среда разработки VSCode с соответствующими плагинами для Python; библиотеки numpy, pygame и pprint.

Комплект 2: Задачи для самостоятельной работы

Код программы

```
import numpy as np
import pygame as pg
from pprint import pprint as pp
import sys

def main():
    T = np.array([[1, 3], [4, 1]])
    lab1_1(T)
    lab1_2()
    lab1_3(T)
    lab1_4(T)

# Самостоятельная работа 1
def lab1_1(tran):
    print("Enter A:")
    A = mat_in(2)
    print("A")
    mat_out(A)
    print("T")
    mat_out(tran)
    A1 = A@tran
    print("A1")
    mat_out(A1)
```

```
# Самостоятельная работа 2
```

```
def lab1_2():  
    sc = pg.display.set_mode((600, 400))  
    pg.font.init()  
    print("Line/Circle/Text\nClose the pygame's window to continue\n")  
    pg.draw.line(sc, (255, 255, 255), [10, 30], [290, 15], 3)  
    pg.draw.circle(sc, (230, 50, 230), (100, 100), 50, 5)  
    f = pg.font.SysFont('serif', 48)  
    t = f.render("Hello world!", False, (190, 60, 30))  
    sc.blit(t, (10, 170))  
    pg.display.update()  
    pg_window()
```

```
# Самостоятельная работа 3
```

```
def lab1_3(tran):  
    print("Enter B:")  
    B = mat_in(2)  
    print("B")  
    mat_out(B)  
    print("T")  
    mat_out(tran)  
    B1 = B@tran  
    print("B1")  
    mat_out(B1)  
    sc = pg.display.set_mode((600, 400))  
    pg.font.init()  
    pg.draw.circle(sc, (10, 50, 230), (B[0], B[1]), 50, 10)  
    pg.draw.circle(sc, (10, 150, 10), (B1[0], B1[1]), 50, 10)  
    f1 = pg.font.SysFont('serif', 24)  
    t1 = f1.render(f"{B}", False, (10, 50, 230))  
    f2 = pg.font.SysFont('serif', 24)  
    t2 = f2.render(f"{B1}", False, (10, 150, 10))  
    sc.blit(t1, (B[0], B[1] + 75))  
    sc.blit(t2, (B1[0], B1[1] + 75))  
    pg.display.update()  
    print("Close the pygame's window to continue\n")  
    pg_window()
```

```

# Самостоятельная работа 4
def lab1_4(tran):
    print("Enter C:")
    C = mat_in(2)
    print("C")
    mat_out(C)
    print("T")
    mat_out(tran)
    C1 = C@tran
    print("C1")
    mat_out(C1)
    print("Enter D:")
    D = mat_in(2)
    print("D")
    mat_out(D)
    print("T")
    mat_out(tran)
    D1 = D@tran
    print("D1")
    mat_out(D1)
    sc = pg.display.set_mode((600, 400))
    pg.font.init()
    print("Close the pygame's window to continue\n")
    pg.draw.line(sc, (255, 0, 0), C, D, 3)
    pg.draw.line(sc, (0, 0, 255), C1, D1, 3)
    f = pg.font.SysFont('serif', 24)
    t1_1 = f.render(f"{C}", False, (255, 0, 0))
    t1_2 = f.render(f"{D}", False, (255, 0, 0))
    t2_1 = f.render(f"{C1}", False, (0, 0, 255))
    t2_2 = f.render(f"{D1}", False, (0, 0, 255))
    sc.blit(t1_1, (C[0], C[1] + 5))
    sc.blit(t1_2, (D[0], D[1] + 5))
    sc.blit(t2_1, (C1[0], C1[1] + 5))
    sc.blit(t2_2, (D1[0], D1[1] + 5))
    pg.display.update()
    pg_window()

# Функция ввода матрицы
def mat_in(n):
    mat = []
    for i in range(n):
        mat.append(int(input()))
    return np.array(mat)

```

```

# Функция вывода матрицы
def mat_out(n):
    print("-----")
    pp(n)
    print("-----\n")

# Функция открытого окна pygame
def pg_window():
    running = True
    while running:
        for i in pg.event.get():
            if i.type == pg.QUIT:
                running = False
                pg.quit()

main()

```

Результаты выполнения программы

2.1 Матричное преобразование координат.

```

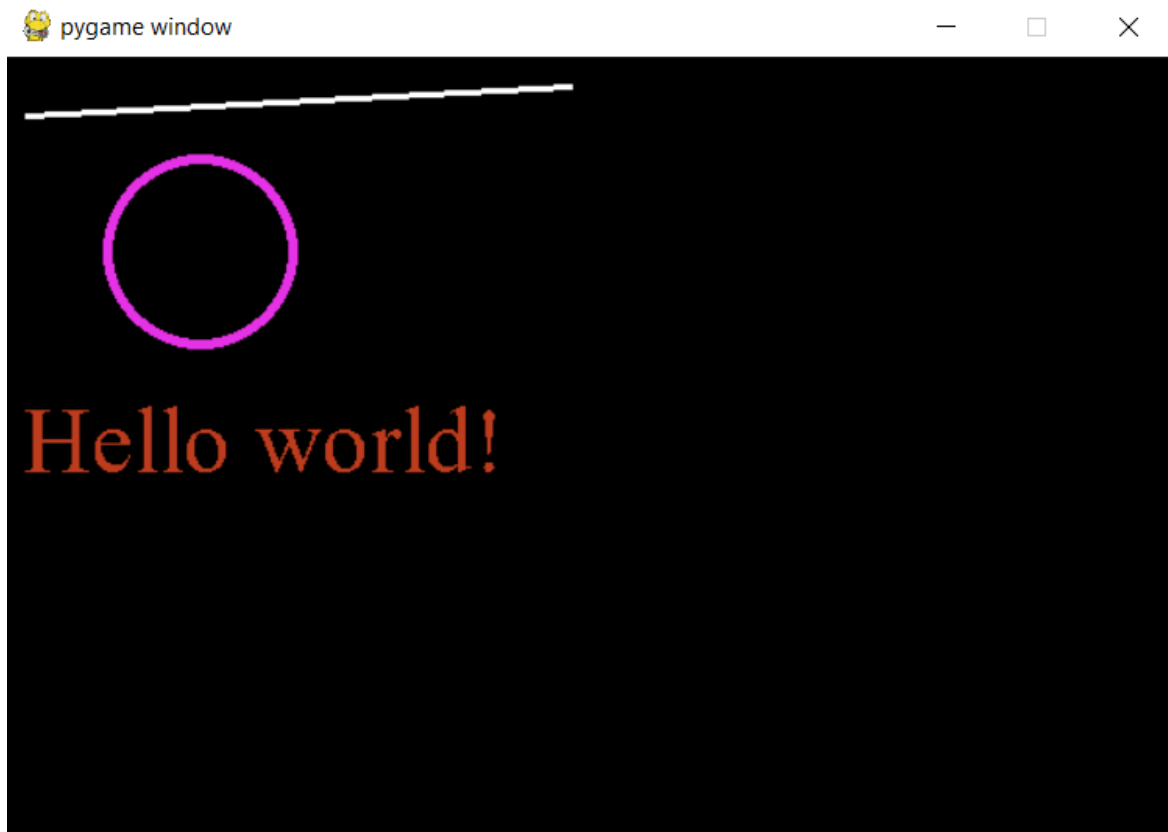
Enter A:
76
68
A
-----
array([76, 68])
-----

T
-----
array([[1, 3],
       [4, 1]])
-----

A1
-----
array([348, 296])
-----

```

2.2 Прорисовка графических примитивов:

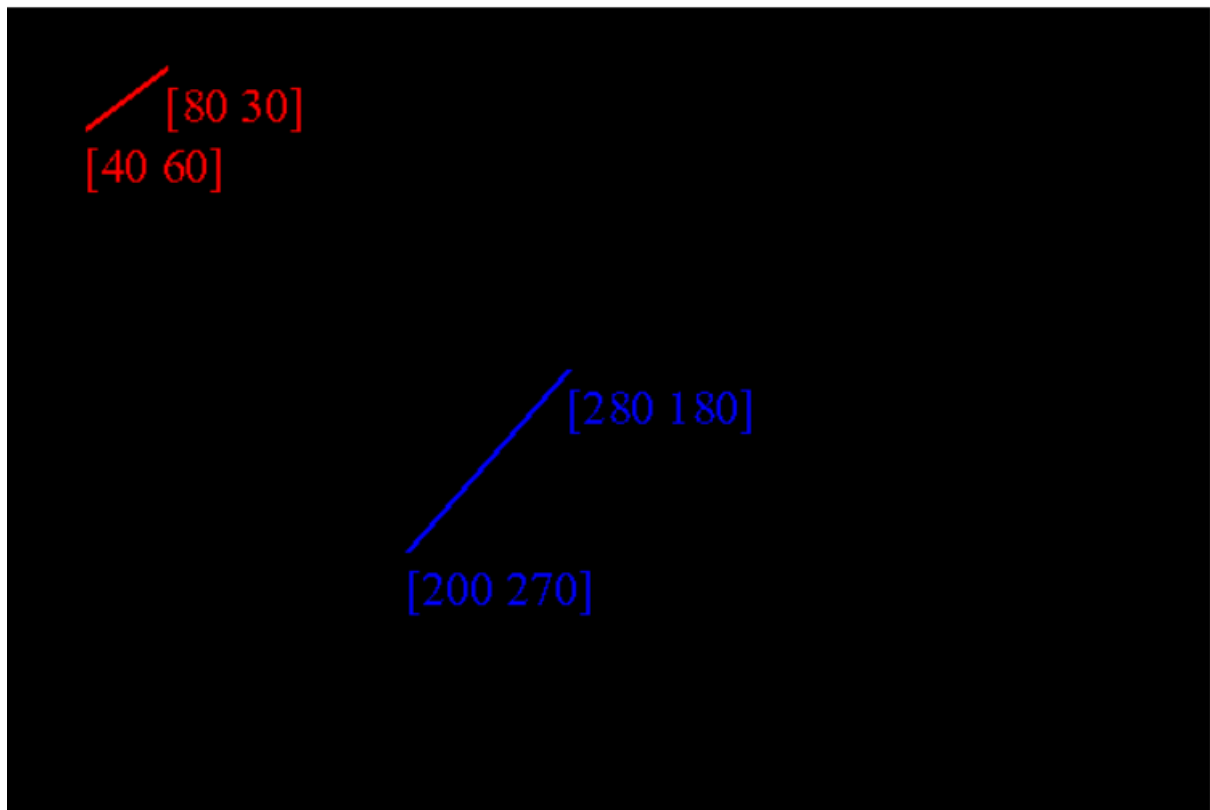


2.3 Прорисовка небольших окружностей с матричным преобразованием координат:



2.4 Прорисовка линий с матричным преобразованием координат:

pygame window



Комплект 3: Задачи для самостоятельно изучения

3.1 Интернет-материалы, посвященные установке языка программирования Python, менеджера пакетов PIP и библиотеки pygame на ваш компьютер.

№	Название	Ссылка	Аннотация
1	Как установить Python на компьютер и начать на нём писать	https://thecode.media/py-install/	Небольшая инструкция по установке Python, а также по последующему подключению VS Code. Вся инструкция сопровождается наглядными скриншотами.
2	Установка PIP на Python 3. Базовые команды	https://otus.ru/journal/ustanovka-pip-na-python-3-bazovye-komandy/	В данной статье рассказывается о том, что представляет из себя PIP, как его устанавливать на разные ОС (Mac, Windows, Linux).
3	Установка pygame и создание шаблона для разработки игр в Python 3	https://www.8host.com/blog/ustanovka-pygame-i-sozдание-shablona-dlya-razrabotki-igr-v-python-3/	В этой инструкции рассматриваются условия установки pygame и её процесс. Также в дополнение идут указания к началу работы с данной библиотекой.

3.2 Интернет-материалы, посвященные основам использования библиотеки программирования игр pygame.

№	Название	Ссылка	Аннотация
1	Pygame documentation	https://devdocs.io/pygame/	На английском описывается огромное количество функций данной библиотеки.
2	Pygame и разработка игр	https://younglinux.info/pygame/pygame	Коллекция из 11 уроков для получения базовых навыков в разработке игр с помощью pygame.
3	Пишем Тетрис на Python с помощью библиотеки Pygame	https://proglib.io/p/pishem-tetris-na-python-s-pomoshchyu-biblioteki-pygame-2022-05-30	Статья, с помощью которой можно попробовать себя в роли разработчика тетриса, получив много полезных приемов и решений в копилку знаний.