

Практический линал. Лабораторная работа № 2

Троицкая Тамара Андреевна, 368924

22 октября 2023 г.

Если Вам совсем лень читать отчёт, зайдите на стр [23](#) и [24](#), пожалуйста.

В прошлый раз я написала код на C, но не заморочилась с оформлением. На этот раз я написала на C++ код для умножения матриц и, как видите, оформила отчёт в латехе. Основные функции из кода я вставлю в конце отчёта, чтобы не смущать людей так сразу. Для визуализации использовала python, библиотеку matplotlib.pyplot. При желании все исходники можно найти здесь:

https://github.com/cgsg-tt6ITMO/s3_practlinal_lab2

В первом задании я рисовала графики в desmos, чтобы показать, откуда берутся точки. Хорошая визуализация представлена в третьем задании, хотя иногда она почти дублируется.

Вывод. Эта лаба заняла у меня 30 часов. Я освоила латех, построение графиков с помощью пайтона, выделение памяти в c++, наглядно увидела, что собственные вектора преобразуются в коллинеарные им самим. Несколько раз писала автоматизацию муторных процессов (нахождение матрицы по двум точкам и их образам, запись матрицы в формате латеха, нахождение образа многоугольника и его изображение в пайтоне). Я стала ценить оформление, оно создаёт иное впечатление от работы. И вообще латех позволяет более понятно объяснить все ходы. Также я стала меньше брезговать пайтоном. И вообще эта лаба помогла мне отвлечься от плохих мыслей и послушать много музыки.

Enjoy!

NB $a = 2$, $b = 3$, $c = 6$, $d = 9$.

Содержание

1	Задание 1. Придумайте.	3
1.1	1 Отображение (симметрия) плоскости относительно $y = ax$ ($a = 2$)	3
1.2	2 Отображение всей плоскости в прямую $y = bx$ ($b = 3$)	3
1.3	3 Поворот плоскости на $10c$ (60) градусов против часовой стрелки.	4
1.4	4 Центральную симметрию плоскости относительно начала координат.	4
1.5	5 Отображение, которое можно описать так: сначала отражение относительно прямой $y = ax$ ($y = 2x$), потом поворот на $10d$ (90) градусов по часовой стрелке.	5
1.6	6 Отображение, которое переводит прямую $y = 0$ в $y = ax$ ($y = 2x$) и прямую $x = 0$ в $y = bx$ ($y = 3x$).	6
1.7	7 Отображение, которое переводит прямую $y = ax$ ($y = 2x$) в $y = 0$ и прямую $y = bx$ ($y = 3x$) в $x = 0$.	7
1.8	8 Отображение, которое меняет местами прямые $y = ax$ и $y = bx$ ($y = 2x$ и $y = 3x$).	7
1.9	9 Отображение, которое переводит круг единичной площади с центром в начале координат в круг площади c ($c = 6$).	8
1.10	10 Отображение, которое переводит круг единичной площади с центром в начале координат в некруг площади d ($d = 9$).	9

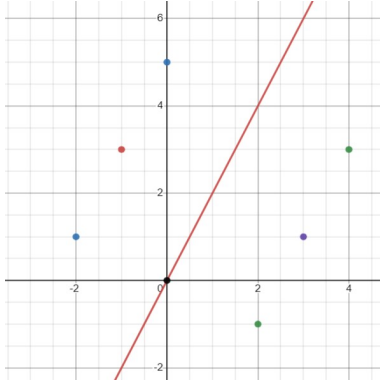
1.11	11	Отображение, у которого собственные вектора перпендикулярны, и ни один из них не лежит на прямой $y = 0$ или $y = x$.	9
1.12	12	Отображение, у которого нет двух неколлинеарных собственных векторов.	9
1.13	13	Отображение, у которого нет ни одного вещественного собственного вектора (но при этом само отображение задаётся вещественной матрицей).	10
1.14	14	Отображение, для которого любой ненулевой вектор является собственным.	10
1.15	15	Пару отображений, последовательное применение которых даёт различные результаты в зависимости от порядка: $AB \neq BA$.	10
1.16	16	Пару отображений, последовательное применение которых даёт одинаковый результат независимо от порядка: $AB = BA$. Постарайтесь, чтобы матрицы A и B были максимально непохожими друг на друга.	11
2		Задание 2. Проанализируйте.	12
2.1	•	Найдите образ и ядро придуманных вами отображений из пунктов 1, 2, 13, 14.	12
2.2	•	Найдите собственные числа и собственные вектора придуманных вами отображений из пунктов 1, 2, 3, 4, 8, 11, 12, 13, 14, 15, 16.	13
2.3	•	Найдите определитель матриц из пунктов 1, 2, 3, 4, 5, 9, 10.	15
2.4	•	В каких пунктах матрица обязательно получается симметричной?	15
3		Задание 3. Визуализируйте.	16
3.1	1)	!	16
3.2	2)		17
3.3	3)		17
3.4	4)		18
3.5	5)		18
3.6	6)		19
3.7	7)		19
3.8	8)		20
3.9	9)		20
3.10	10)		20
3.11	11)	!	21
3.12	12)	!	21
3.13	13)		22
3.14	14)	!	22
3.15	15)	!	23
3.16	16)	!	24
4		Приложение (код)	25
4.1		Вспомогательные функции для подсчёта определителя (с++)	25
4.2		Функции для работы с матрицами для первого задания (с++)	26
4.3		Функции для автоматического подсчёта образа многоугольника (python)	28

1 Задание 1. Придумайте.

Придумайте матрицы 2×2 , которые задают:

1.1 1 Отображение (симметрия) плоскости относительно $y = ax$ ($a = 2$)

В начало: [1](#) Визуализация: [16](#)



Примеры преобразования точек:

$$\begin{pmatrix} 0 & 5 \end{pmatrix} \rightarrow \begin{pmatrix} 4 & 3 \end{pmatrix}$$

$$\begin{pmatrix} 3 & 1 \end{pmatrix} \rightarrow \begin{pmatrix} -1 & 3 \end{pmatrix}$$

$$A \cdot B = C$$

где матрица A - искомая матрица преобразования, столбцы матрицы B - это исходные векторы, а столбцы матрицы C - это полученные в результате преобразования векторы.

$$A = C \cdot B^{-1}$$

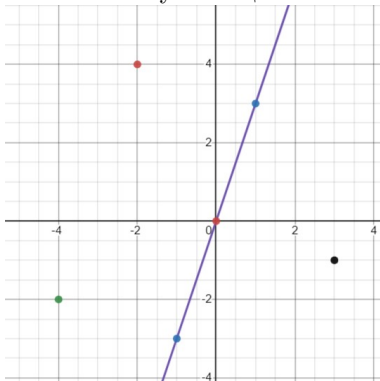
$$\begin{pmatrix} 4 & -1 \\ 3 & 3 \end{pmatrix} \cdot \begin{pmatrix} 0 & 3 \\ 5 & 1 \end{pmatrix}^{-1} = \boxed{\begin{pmatrix} -0.6 & 0.8 \\ 0.8 & 0.6 \end{pmatrix}}$$

Проверка:

$$\begin{pmatrix} -0.6 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \cdot \begin{pmatrix} -2 \\ 1 \end{pmatrix} = \begin{pmatrix} (-0.6) \cdot (-2) + 0.8 \cdot 1 \\ 0.8 \cdot (-2) + 0.6 \cdot 1 \end{pmatrix} = \begin{pmatrix} 2 \\ -1 \end{pmatrix}$$

1.2 2 Отображение всей плоскости в прямую $y = bx$ ($b = 3$)

В начало: [1](#) Визуализация: [17](#)



Примеры преобразования точек:

$$\begin{pmatrix} -2 & 4 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 3 \end{pmatrix}$$

$$\begin{pmatrix} -4 & -2 \end{pmatrix} \rightarrow \begin{pmatrix} -1 & -3 \end{pmatrix}$$

$$C \cdot B^{-1} = A$$

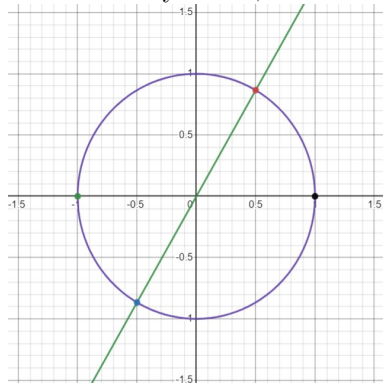
$$\begin{pmatrix} 1 & -1 \\ 3 & -3 \end{pmatrix} \cdot \begin{pmatrix} -2 & -4 \\ 4 & -2 \end{pmatrix}^{-1} = \boxed{\begin{pmatrix} 0.1 & 0.3 \\ 0.3 & 0.9 \end{pmatrix}}$$

Проверка:

$$\begin{pmatrix} 0.1 & 0.3 \\ 0.3 & 0.9 \end{pmatrix} \cdot \begin{pmatrix} 3 \\ -1 \end{pmatrix} = \begin{pmatrix} 0.1 \cdot 3 + 0.3 \cdot (-1) \\ 0.3 \cdot 3 + 0.9 \cdot (-1) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

1.3 3 Поворот плоскости на 10с (60) градусов против часовой стрелки.

В начало: [1](#) Визуализация: [17](#)



Примеры преобразования точек:

$$\begin{pmatrix} 1 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 1/2 & \sqrt{3}/2 \end{pmatrix}$$

$$\begin{pmatrix} -1 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} -1/2 & -\sqrt{3}/2 \end{pmatrix}$$

$$C \cdot B^{-1} = A$$

$$\begin{pmatrix} 1/2 & -1/2 \\ \sqrt{3}/2 & -\sqrt{3}/2 \end{pmatrix} \cdot \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}^{-1} = \boxed{\begin{pmatrix} 1/2 & -\sqrt{3}/2 \\ \sqrt{3}/2 & 1/2 \end{pmatrix}}$$

Вспомним, что матрица поворота на α :

$$\begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

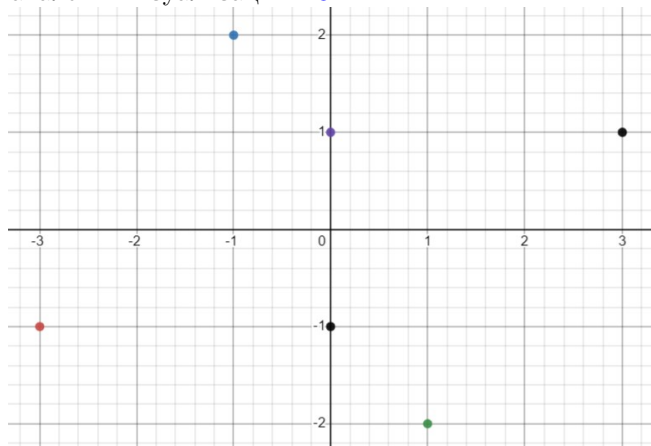
Полученный нами результат согласуется с теорией.

Проверка:

$$\begin{pmatrix} \frac{1}{2} & \frac{-\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} \frac{\sqrt{3}}{2} \\ \frac{1}{2} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

1.4 4 Центральную симметрию плоскости относительно начала координат.

В начало: [1](#) Визуализация: [18](#)



Примеры преобразования точек:

$$\begin{pmatrix} 3 & 1 \end{pmatrix} \rightarrow \begin{pmatrix} -3 & -1 \end{pmatrix}$$

$$\begin{pmatrix} -1 & 2 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & -2 \end{pmatrix}$$

$$C \cdot B^{-1} = A$$

$$\begin{pmatrix} -3 & 1 \\ -1 & -2 \end{pmatrix} \cdot \begin{pmatrix} 3 & -1 \\ 1 & 2 \end{pmatrix}^{-1} = \boxed{\begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}}$$

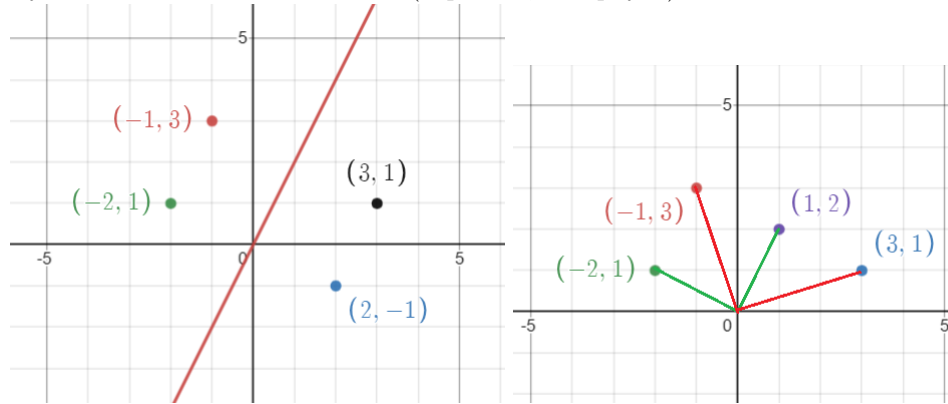
Проверка:

$$\begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

1.5 5 Отображение, которое можно описать так: сначала отражение относительно прямой $y = ax$ ($y = 2x$), потом поворот на $10d$ (90) градусов по часовой стрелке.

В начало: [1](#) Визуализация: [18](#)

Будем искать точки в два этапа (отразить, повернуть):



Запишем преобразование векторов в два этапа:

$$\begin{pmatrix} 2 & -1 \end{pmatrix} \rightarrow \begin{pmatrix} -2 & 1 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 2 \end{pmatrix}$$

$$\begin{pmatrix} 3 & 1 \end{pmatrix} \rightarrow \begin{pmatrix} -1 & 3 \end{pmatrix} \rightarrow \begin{pmatrix} 3 & 1 \end{pmatrix}$$

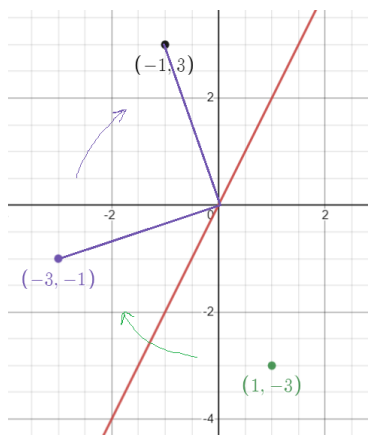
$$C \cdot B^{-1} = A = \begin{pmatrix} 1 & 3 \\ 2 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 & 3 \\ -1 & 1 \end{pmatrix}^{-1} = \boxed{\begin{pmatrix} 0.8 & 0.6 \\ 0.6 & -0.8 \end{pmatrix}}$$

Заметим, что

$$\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} -0.6 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} = \begin{pmatrix} 0.8 & 0.6 \\ 0.6 & -0.8 \end{pmatrix}$$

То есть эту матрицу можно получить последовательным умножением матрицы поворота (см. п. 3) и матрицы отражения (см. п. 1). Порядок именно такой, потому что при умножении этой конструкции на вектор сначала на него умножится матрица отражения, находящаяся правее, а потом уже матрица поворота. Можно сказать, что действие произойдёт справа налево.

Проверка:

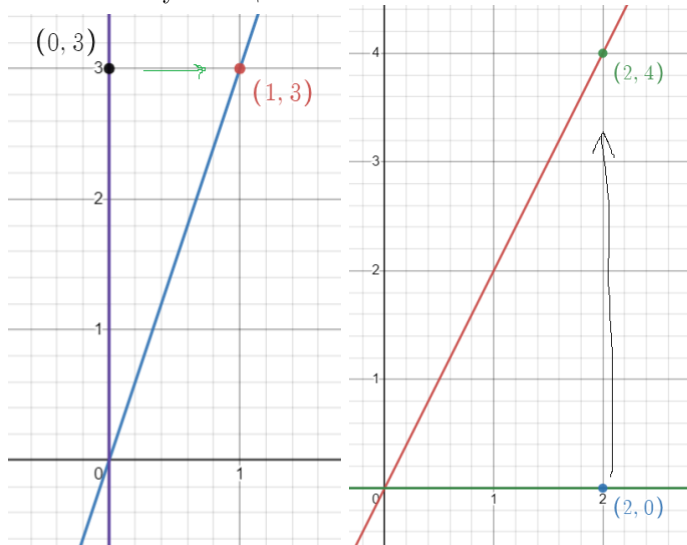


$$\begin{pmatrix} 1 & -3 \end{pmatrix} \rightarrow \begin{pmatrix} -3 & -1 \end{pmatrix} \rightarrow \begin{pmatrix} -1 & 3 \end{pmatrix}$$

$$\begin{pmatrix} 0.8 & 0.6 \\ 0.6 & -0.8 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ -3 \end{pmatrix} = \begin{pmatrix} -1 \\ 3 \end{pmatrix}$$

1.6 6 Отображение, которое переводит прямую $y = 0$ в $y = ax$ ($y = 2x$) и прямую $x = 0$ в $y = bx$ ($y = 3x$).

В начало: [1](#) Визуализация: [19](#)



Примеры преобразования точек:

$$\begin{pmatrix} 0 & 3 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 3 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 2 & 4 \end{pmatrix}$$

$$C \cdot B^{-1} = A$$

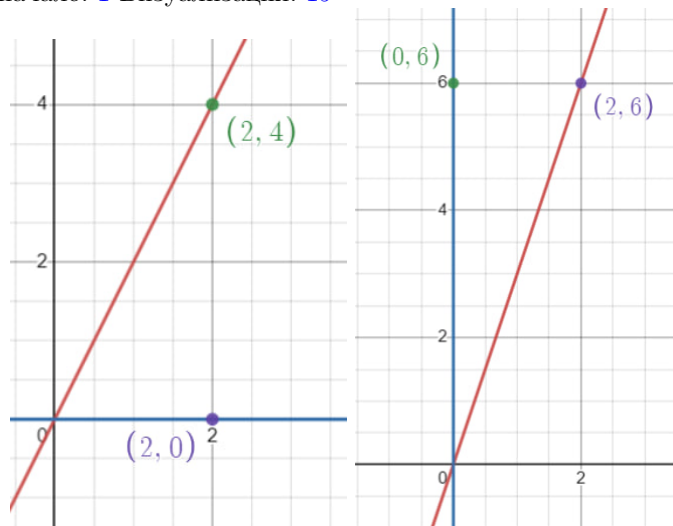
$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \cdot \begin{pmatrix} 0 & 2 \\ 3 & 0 \end{pmatrix}^{-1} = \boxed{\begin{pmatrix} 1 & 1/3 \\ 2 & 1 \end{pmatrix}}$$

Проверка:

$$\begin{pmatrix} 1 & 1/3 \\ 2 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 6 \end{pmatrix} = \begin{pmatrix} 2 \\ 6 \end{pmatrix}$$

1.7 7 Отображение, которое переводит прямую $y = ax$ ($y = 2x$) в $y = 0$ и прямую $y = bx$ ($y = 3x$) в $x = 0$.

В начало: [1](#) Визуализация: [19](#)



Примеры преобразования точек:

$$\begin{pmatrix} 2 & 4 \end{pmatrix} \rightarrow \begin{pmatrix} 2 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 6 \end{pmatrix} \rightarrow \begin{pmatrix} 0 & 6 \end{pmatrix}$$

$$C \cdot B^{-1} = A$$

$$\begin{pmatrix} 2 & 0 \\ 0 & 6 \end{pmatrix} \cdot \begin{pmatrix} 2 & 2 \\ 4 & 6 \end{pmatrix}^{-1} = \begin{pmatrix} 3 & -1 \\ -6 & 3 \end{pmatrix}$$

Можно заметить, что

$$\begin{pmatrix} 1 & 1/3 \\ 2 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} 3 & -1 \\ -6 & 3 \end{pmatrix}$$

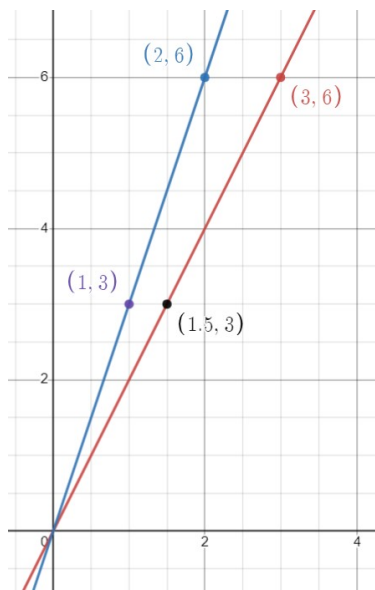
(Получившаяся матрица обратна к матрице из прошлого задания)

Проверка:

$$\begin{pmatrix} 3 & -1 \\ -6 & 3 \end{pmatrix} \cdot \begin{pmatrix} -1 \\ -2 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$$

1.8 8 Отображение, которое меняет местами прямые $y = ax$ и $y = bx$ ($y = 2x$ и $y = 3x$).

В начало: [1](#) Визуализация: [20](#)



Примеры преобразования точек:

$$\begin{pmatrix} 1 & 3 \end{pmatrix} \rightarrow \begin{pmatrix} 1.5 & 3 \end{pmatrix}$$

$$\begin{pmatrix} 3 & 6 \end{pmatrix} \rightarrow \begin{pmatrix} 2 & 6 \end{pmatrix}$$

$$C \cdot B^{-1} = A$$

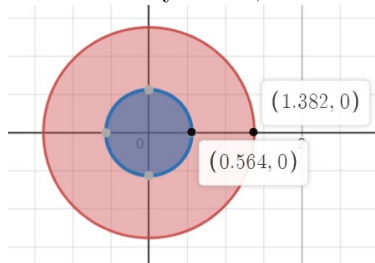
$$\begin{pmatrix} 1 & 2 \\ 3 & 6 \end{pmatrix} \cdot \begin{pmatrix} 1 & 3 \\ 3 & 6 \end{pmatrix}^{-1} = \boxed{\begin{pmatrix} -1 & 5/6 \\ 0 & 1 \end{pmatrix}}$$

Проверка:

$$\begin{pmatrix} -1 & 5/6 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} -1 \\ -3 \end{pmatrix} = \begin{pmatrix} -1.5 \\ -3 \end{pmatrix}$$

1.9 [9] Отображение, которое переводит круг единичной площади с центром в начале координат в круг площади с ($c = 6$).

В начало: [1](#) Визуализация: [20](#)



$$\pi r^2 = 1$$

$$r = 1/\sqrt{\pi}$$

$$\pi R^2 = c$$

$$R = \sqrt{c/\pi} = \sqrt{6/\pi}$$

Примеры преобразования точек:

$$\begin{pmatrix} 0 & r \end{pmatrix} \rightarrow \begin{pmatrix} 0 & R \end{pmatrix}$$

$$\begin{pmatrix} r & 0 \end{pmatrix} \rightarrow \begin{pmatrix} R & 0 \end{pmatrix}$$

$$B = \begin{pmatrix} 0 & 1/\sqrt{\pi} \\ 1/\sqrt{\pi} & 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & \sqrt{6/\pi} \\ \sqrt{6/\pi} & 0 \end{pmatrix}$$

$$A = \begin{pmatrix} \sqrt{6} & 0 \\ 0 & \sqrt{6} \end{pmatrix}$$

1.10 10 Отображение, которое переводит круг единичной площади с центром в начале координат в некруг площади d ($d = 9$).

В начало: [1](#) Визуализация: [20](#)

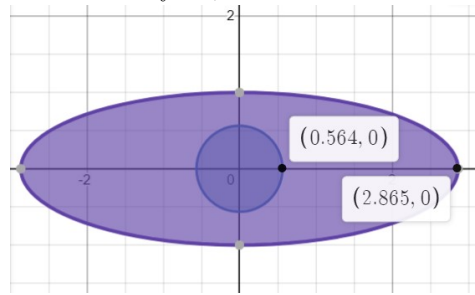
$$\pi r^2 = 1$$

$$r = 1/\sqrt{\pi}$$

Допустим, некруг это эллипс. Площадь эллипса вычисляется как $S = ab\pi$. Например, $a = 9/\pi, b = 1$. Тогда по оси ox фигура растягивается в $1 : (1/\sqrt{\pi}) = \sqrt{\pi}$ раз, по оси oy в $\frac{9}{\pi} : \frac{1}{\sqrt{\pi}} = 9/\sqrt{\pi}$ раз.

То есть собственные векторы: $v_1 = (1, 0)$ и $v_2 = (0, 1)$,

а соответствующие им собственные числа: $\lambda_1 = \sqrt{\pi}, \lambda_2 = 9/\sqrt{\pi}$



$$\begin{pmatrix} \sqrt{\pi} & 0 \\ 0 & 9/\sqrt{\pi} \end{pmatrix}$$

1.11 11 Отображение, у которого собственные вектора перпендикулярны, и ни один из них не лежит на прямой $y = 0$ или $y = x$.

В начало: [1](#) Визуализация: [21](#)

$$\text{Пусть } v_1 = \begin{pmatrix} 2 & 1 \end{pmatrix} \lambda_1 = -1,$$

$$v_2 = \begin{pmatrix} -1 & 2 \end{pmatrix} \lambda_2 = 3/2$$

$$A \begin{pmatrix} 2 \\ 1 \end{pmatrix} = -1 \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

$$A \begin{pmatrix} -1 \\ 2 \end{pmatrix} = 3/2 \begin{pmatrix} -1 \\ 2 \end{pmatrix}$$

$$A \begin{pmatrix} 2 & -1 \\ 1 & 2 \end{pmatrix} = \begin{pmatrix} -2 & -1.5 \\ -1 & 3 \end{pmatrix}$$

$$A = C \cdot B^{-1} = \begin{pmatrix} -2 & -1.5 \\ -1 & 3 \end{pmatrix} \begin{pmatrix} 2 & -1 \\ 1 & 2 \end{pmatrix}^{-1} = \begin{pmatrix} -0.5 & -1 \\ -1 & 1 \end{pmatrix}$$

1.12 12 Отображение, у которого нет двух неколлинеарных собственных векторов.

В начало: [1](#) Визуализация: [21](#)

$$\begin{pmatrix} 2 & 1 \\ 0 & 2 \end{pmatrix}$$

Это Жорданова клетка для $\lambda = 2$ с геометрической кратностью 1. Все собственные векторы пропорциональны $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$, то есть коллинеарны. (Если говорить безграмотно, у неё только один собственный вектор)

1.13 13 Отображение, у которого нет ни одного вещественного собственного вектора (но при этом само отображение задаётся вещественной матрицей).

В начало: [1](#) Визуализация: [22](#)

$$\begin{pmatrix} -2 & -1 \\ 9 & -2 \end{pmatrix}$$

$$\det \begin{pmatrix} -2 - \lambda & -1 \\ 9 & -2 - \lambda \end{pmatrix} = (\lambda + 2)^2 + 9 = 0$$

$$\lambda_1 = -2 - 3i \quad v_1 = \begin{pmatrix} -i/3 \\ 1 \end{pmatrix}$$

$$\lambda_2 = -2 + 3i \quad v_2 = \begin{pmatrix} i/3 \\ 1 \end{pmatrix}$$

1.14 14 Отображение, для которого любой ненулевой вектор является собственным.

В начало: [1](#) Визуализация: [22](#)

$$\begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix}$$

$$\lambda = 4$$

$$\begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 4x \\ 4y \end{pmatrix}$$

$$\forall x, y$$

1.15 15 Пару отображений, последовательное применение которых даёт различные результаты в зависимости от порядка: $AB \neq BA$.

В начало: [1](#) Визуализация: [23](#)

Отразим относительно прямой $y = x$ и повернём на 90° против часовой стрелки.

Матрица поворота на 90° против часовой стрелки:

$$A = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

Матрица отражения относительно прямой $y = x$:

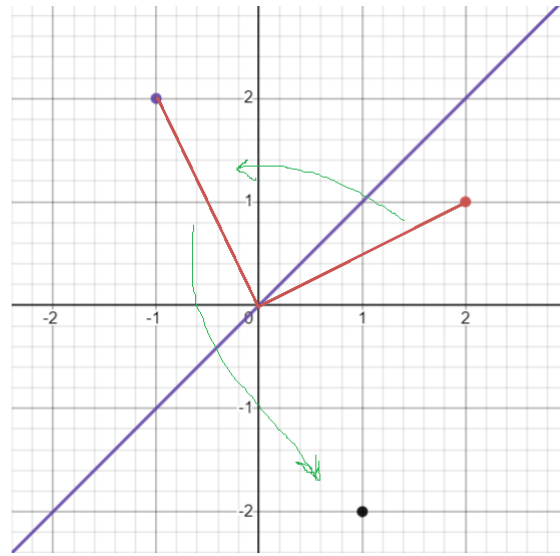
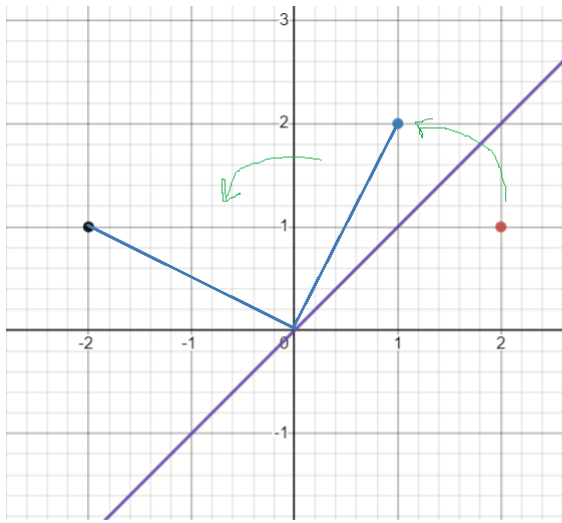
$$B = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Отразить и потом повернуть:

$$A \cdot B = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$$

Повернуть и потом отразить:

$$B \cdot A = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$



Слева отразили и повернули, справа повернули и отразили. Результаты не совпали.

1.16 16 Пару отображений, последовательное применение которых даёт одинаковый результат независимо от порядка: $AB = BA$. Постарайтесь, чтобы матрицы A и B были максимально непохожими друг на друга.

В начало: [1](#) Визуализация: [24](#)

Сделать два поворота. Сначала на 60° , потом на 30° . И в обратной последовательности.

Матрица поворота на 60° против часовой стрелки:

$$A = \begin{pmatrix} \cos(60^\circ) & -\sin(60^\circ) \\ \sin(60^\circ) & \cos(60^\circ) \end{pmatrix} = \begin{pmatrix} 1/2 & -\sqrt{3}/2 \\ \sqrt{3}/2 & 1/2 \end{pmatrix}$$

Матрица поворота на 30° против часовой стрелки:

$$B = \begin{pmatrix} \cos(30^\circ) & -\sin(30^\circ) \\ \sin(30^\circ) & \cos(30^\circ) \end{pmatrix} = \begin{pmatrix} \sqrt{3}/2 & -1/2 \\ 1/2 & \sqrt{3}/2 \end{pmatrix}$$

$$A \cdot B = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

$$B \cdot A = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

Нетрудно заметить, что в обоих случаях получилась матрица поворота на 90° против часовой оси. Это согласуется с теорией (и здравым смыслом) – порядок выполнения двух поворотов неважен, повороты "складываются".

2 Задание 2. Проанализируйте.

2.1 • Найдите образ и ядро придуманных вами отображений из пунктов 1, 2, 13, 14.

В начало: [1](#)

$$1) \begin{pmatrix} -0.6 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \sim \begin{pmatrix} 0 & 1.25 \\ 0.8 & 0.6 \end{pmatrix}$$
$$\begin{pmatrix} -0.6 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -0.6 \\ 0.8 \end{pmatrix}, \begin{pmatrix} -0.6 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0.8 \\ 0.6 \end{pmatrix}$$

Получаем образ:

$$\text{im}(A) = \text{Span}\left[\begin{pmatrix} -0.6 \\ 0.8 \end{pmatrix}, \begin{pmatrix} 0.8 \\ 0.6 \end{pmatrix}\right] = \text{Span}\left[\begin{pmatrix} -3 \\ 4 \end{pmatrix}, \begin{pmatrix} 4 \\ 3 \end{pmatrix}\right]$$

$$\text{rank}(A) = 2$$

$$\text{nullity}(A) = 0$$

$$\ker f(A) = \{0\}$$

$$2) \begin{pmatrix} 0.1 & 0.3 \\ 0.3 & 0.9 \end{pmatrix} \sim \begin{pmatrix} 0.1 & 0.3 \\ 0 & 0 \end{pmatrix}$$

$$\text{rank}(A) = 1$$

$$\text{nullity}(A) = 1$$

$$\text{im}(A) = \text{Span}\left[\begin{pmatrix} 1 \\ 3 \end{pmatrix}\right]$$

$$\text{Ядро: } 0.1x + 0.3y = 0$$

$$\ker f(A) = \text{Span}\left[\begin{pmatrix} -3 \\ 1 \end{pmatrix}\right]$$

$$13) \begin{pmatrix} -2 & -1 \\ 9 & -2 \end{pmatrix} \sim \begin{pmatrix} -2 & -1 \\ 0 & -6.5 \end{pmatrix}$$

$$\text{rank}(A) = 2$$

$$\text{im}(A) = \text{Span}\left[\begin{pmatrix} -2 \\ 9 \end{pmatrix}, \begin{pmatrix} -1 \\ -2 \end{pmatrix}\right]$$

$$\text{nullity}(A) = 0$$

$$\ker f(A) = \{0\}$$

$$14) \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix}$$

$$\text{rank}(A) = 2$$

$$\text{nullity}(A) = 0$$

Образ – любой вектор длины 2.

$$\text{Im}(A) = \text{Span}\left[\begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}\right]$$

$$\ker f(A) = \{0\}$$

2.2 • Найдите собственные числа и собственные вектора придуманных вами отображений из пунктов 1, 2, 3, 4, 8, 11, 12, 13, 14, 15, 16.

В начало: [1](#)

$$1) \begin{pmatrix} -0.6 & 0.8 \\ 0.8 & 0.6 \end{pmatrix}$$

$$\det \begin{pmatrix} -0.6 - \lambda & 0.8 \\ 0.8 & 0.6 - \lambda \end{pmatrix} = (\lambda + 0.6) \cdot (\lambda - 0.6) - 0.64 = \lambda^2 - 0.36 - 0.64 = \lambda^2 - 1 = 0$$

$$\lambda_1 = 1, \lambda_2 = -1$$

$$v_1 = \begin{pmatrix} 0.5 \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} -2 \\ 1 \end{pmatrix}$$

$$2) \begin{pmatrix} 0.1 & 0.3 \\ 0.3 & 0.9 \end{pmatrix}$$

$$\det \begin{pmatrix} 0.1 - \lambda & 0.3 \\ 0.3 & 0.9 - \lambda \end{pmatrix} = (0.1 - \lambda) \cdot (0.9 - \lambda) - 0.9 = \lambda^2 - \lambda + 0.9 - 0.9 = 0$$

$$\lambda_1 = 0, \lambda_2 = 1$$

$$v_1 = \begin{pmatrix} -3 \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} 1/3 \\ 1 \end{pmatrix}$$

$$3) \begin{pmatrix} 1/2 & -\sqrt{3}/2 \\ \sqrt{3}/2 & 1/2 \end{pmatrix}$$

$$\lambda_1 = \frac{-\sqrt{3}i+1}{2}, \lambda_2 = \frac{\sqrt{3}i+1}{2}$$

$$v_1 = \begin{pmatrix} -i \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} i \\ 1 \end{pmatrix}$$

$$4) \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$\lambda_1 = -1$$

$$v_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, v_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$8) \begin{pmatrix} -1 & 5/6 \\ 0 & 1 \end{pmatrix}$$

$$\lambda_1 = 1, \lambda_2 = -1$$

$$v_1 = \begin{pmatrix} 5/12 \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$11) \begin{pmatrix} -0.5 & -1 \\ -1 & 1 \end{pmatrix}$$

$$\lambda_1 = -1, \lambda_2 = 1.5$$

$$v_1 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} -0.5 \\ 1 \end{pmatrix}$$

$$12) \begin{pmatrix} 2 & 1 \\ 0 & 2 \end{pmatrix}$$

$$\lambda_1 = 2$$

$$v_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$

$$13) \begin{pmatrix} -2 & -1 \\ 9 & -2 \end{pmatrix}$$

$$\lambda_1 = -2 - 3i, \lambda_2 = -2 + 3i$$

$$v_1 = \begin{pmatrix} -i/3 \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} i/3 \\ 1 \end{pmatrix}$$

$$14) \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix}$$

$$\lambda_1 = 4$$

$$v_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, v_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$15)$$

$$A = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

$$\lambda_1 = i, \lambda_2 = -i$$

$$v_1 = \begin{pmatrix} i \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} -i \\ 1 \end{pmatrix}$$

$$B = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$\lambda_1 = 1, \lambda_2 = -1$$

$$v_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

$$A \cdot B = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\lambda_1 = 1, \lambda_2 = -1$$

$$v_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$B \cdot A = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$\lambda_1 = 1, \lambda_2 = -1$$

$$v_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, v_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$16)$$

$$A = \begin{pmatrix} 1/2 & -\sqrt{3}/2 \\ \sqrt{3}/2 & 1/2 \end{pmatrix}$$

$$\lambda_1 = \frac{-\sqrt{3}i+1}{2}, \lambda_2 = \frac{\sqrt{3}i+1}{2}$$

$$v_1 = \begin{pmatrix} -i \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} i \\ 1 \end{pmatrix}$$

$$B = \begin{pmatrix} \sqrt{3}/2 & -1/2 \\ 1/2 & \sqrt{3}/2 \end{pmatrix}$$

$$\lambda_1 = \frac{\sqrt{3}-i}{2}, \lambda_2 = \frac{\sqrt{3}+i}{2}$$

$$v_1 = \begin{pmatrix} -i \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} i \\ 1 \end{pmatrix}$$

$$A \cdot B = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

$$\lambda_1 = i, \lambda_2 = -i$$

$$v_1 = \begin{pmatrix} i \\ 1 \end{pmatrix}, v_2 = \begin{pmatrix} -i \\ 1 \end{pmatrix}$$

$$B \cdot A = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

2.3 • Найдите определитель матриц из пунктов 1, 2, 3, 4, 5, 9, 10.

В начало: [1](#)

$$1) \det \begin{pmatrix} -0.6 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} = -0.36 - 0.64 = -1$$

$$2) \det \begin{pmatrix} 0.1 & 0.3 \\ 0.3 & 0.9 \end{pmatrix} = 0$$

$$3) \det \begin{pmatrix} 1/2 & -\sqrt{3}/2 \\ \sqrt{3}/2 & 1/2 \end{pmatrix} = 1/4 + 3/4 = 1$$

$$4) \det \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} = 1$$

$$5) \det \begin{pmatrix} 0.8 & 0.6 \\ 0.6 & -0.8 \end{pmatrix} = -0.64 - 0.36 = -1$$

$$9) \det \begin{pmatrix} \sqrt{6} & 0 \\ 0 & \sqrt{6} \end{pmatrix} = 6$$

$$10) \det \begin{pmatrix} \sqrt{\pi} & 0 \\ 0 & 9/\sqrt{\pi} \end{pmatrix} = 9/\pi$$

2.4 • В каких пунктах матрица обязательно получается симметричной?

В начало: [1](#)

Точно да: 1, 2, 4, 5, 9, 10 (если делать эллипс), 11, 14.

(точно будет 2 вещественных собственных вектора и вещественные собственные числа, собственные векторы ортогональные)

Точно не: 3, 6, 7, 8, 12, 13, 15, 16.

3 Задание 3. Визуализируйте.

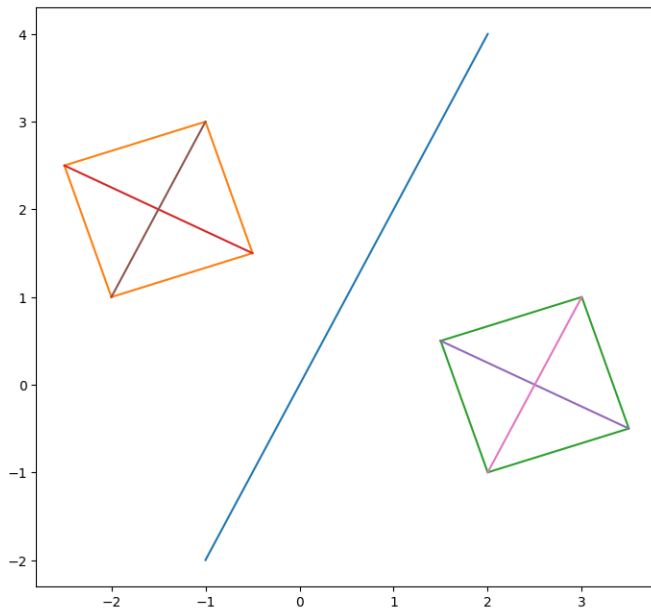
Используя MATLAB или Python, выполните визуализацию полученных линейных преобразований. Для этого:

- Задайте произвольную фигуру как многоугольник с вершинами в выбранных вами точках. Постройте её графическое изображение. Это – оригинал.
- Найдите образ каждой вершины многоугольника при линейном отображении рассматриваемой матрицей. Постройте графическое изображение многоугольника на полученных (отображённых) вершинах. Это – результат преобразования, образ.
- Выполните указанную визуализацию для всех отображений из первого задания.
- При работе с пунктами 15 и 16 сделайте визуализацию всех рассматриваемых отображений, а именно: А, В, АВ и ВА.
- Для пунктов 1, 11, 12, 14, 15, 16 добавьте на картинку прямые, совпадающие с направлениями собственных векторов.

В отображениях, где поворот или отражение производится относительно центра, я добавила оси. Восклицательным знаком пометила те задачи, в которых нужно также изобразить собственные векторы.

3.1 1) !

В начало: [1](#) К описанию матрицы: [3](#) Собственные векторы: [13](#)

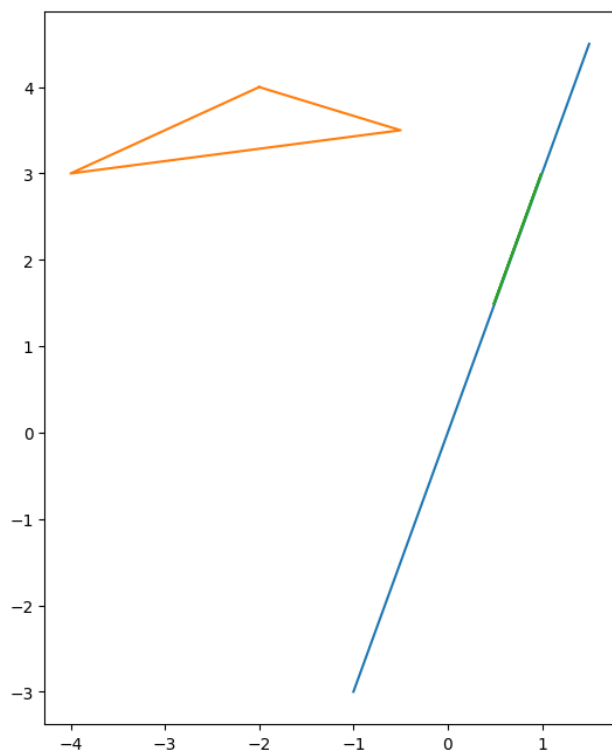


Зелёный – оригинал, рыжий – его образ. Но вообще они переходят друг в друга. Синяя прямая – ось симметрии.

Фиолетовый и красный, розовый и коричневый соответственно коллинеарны собственным векторам матрицы этого отображения, поэтому они не изменили направления.

3.2 2)

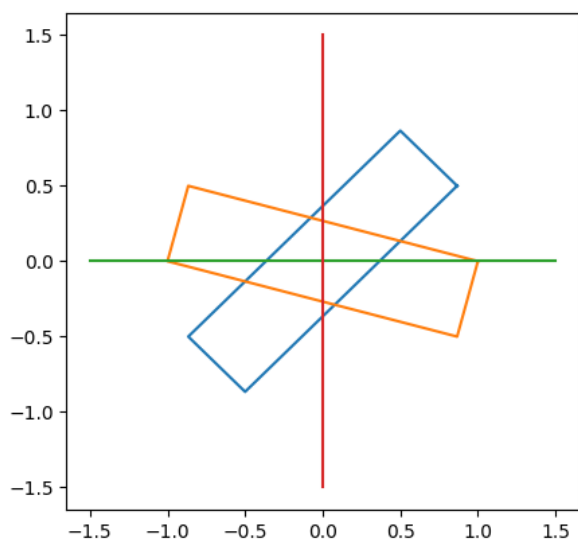
В начало: [1](#) К описанию матрицы: [3](#)



Рыжий – оригинал, он перешёл в зелёный отрезок.

3.3 3)

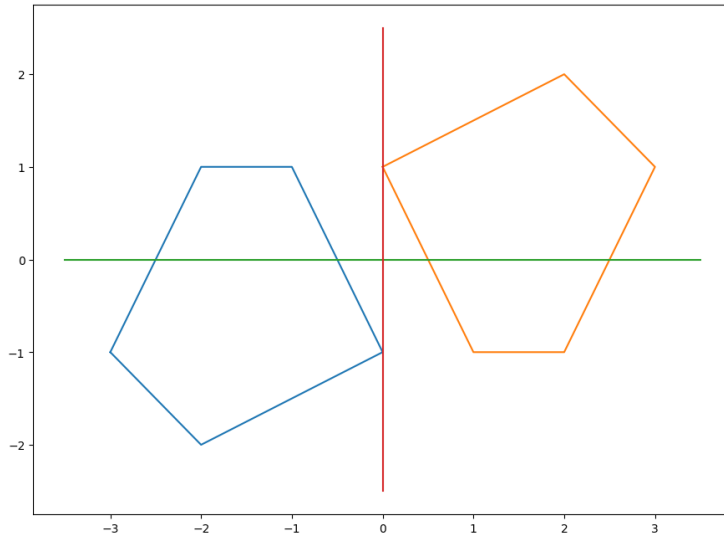
В начало: [1](#) К описанию матрицы: [4](#)



Рыжий – оригинал, синий – его образ

3.4 4)

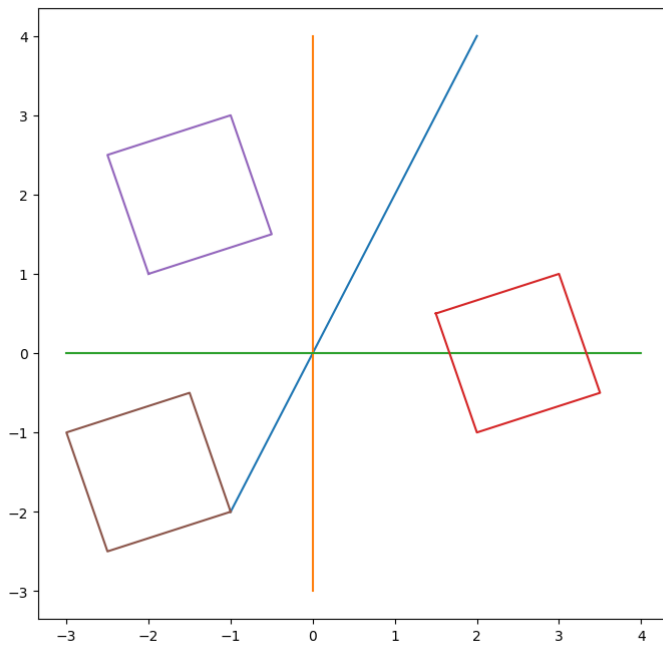
В начало: [1](#) К описанию матрицы: [4](#)



Рыжий – оригинал, синий – его образ (но они переходят друг в друга вообще)

3.5 5)

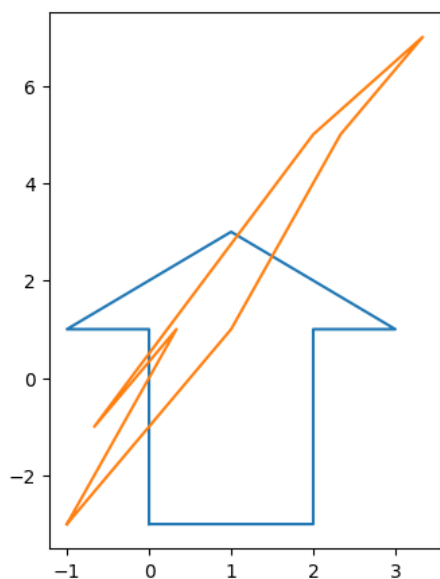
В начало: [1](#) К описанию матрицы: [5](#)



Красный квадратик отразили, получили фиолетовый. Фиолетовый повернули на 90 градусов – получили коричневый.

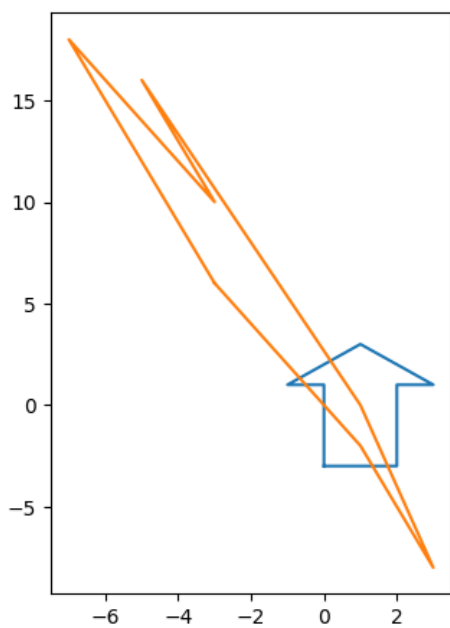
3.6 6)

В начало: [1](#) К описанию матрицы: [6](#)



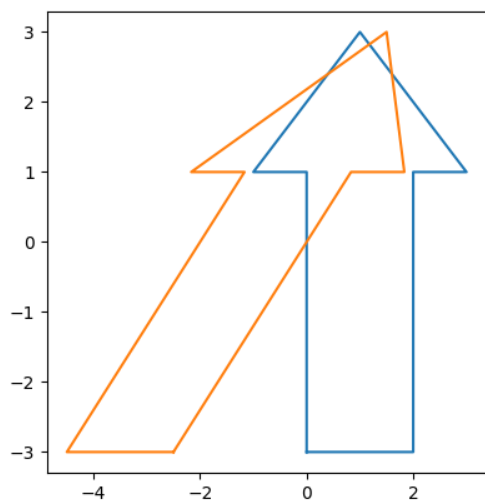
3.7 7)

В начало: [1](#) К описанию матрицы: [6](#)



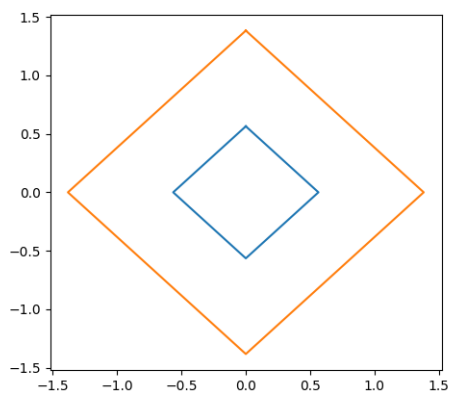
3.8 8)

В начало: [1](#) К описанию матрицы: [7](#)



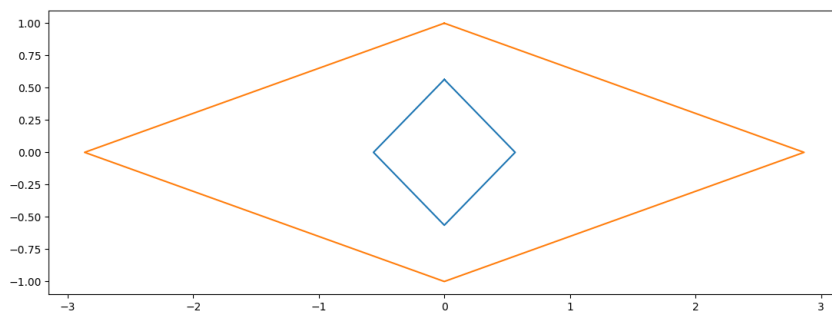
3.9 9)

В начало: [1](#) К описанию матрицы: [8](#)



3.10 10)

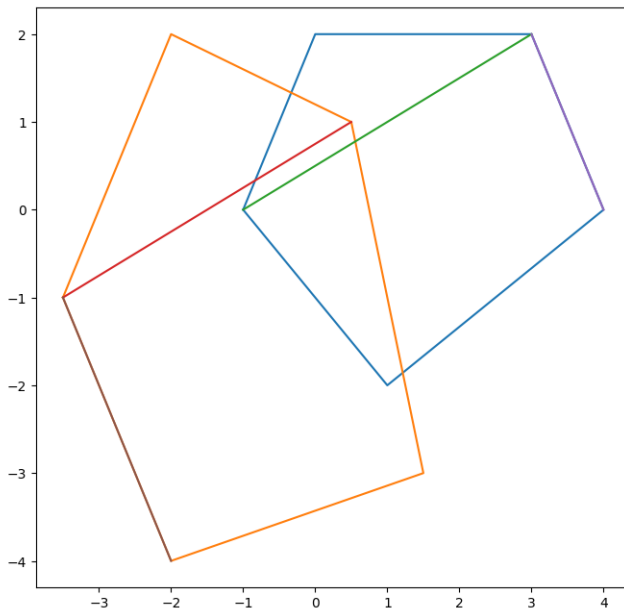
В начало: [1](#) К описанию матрицы: [8](#)



Синий – оригинал, рыжий – образ.

3.11 11) !

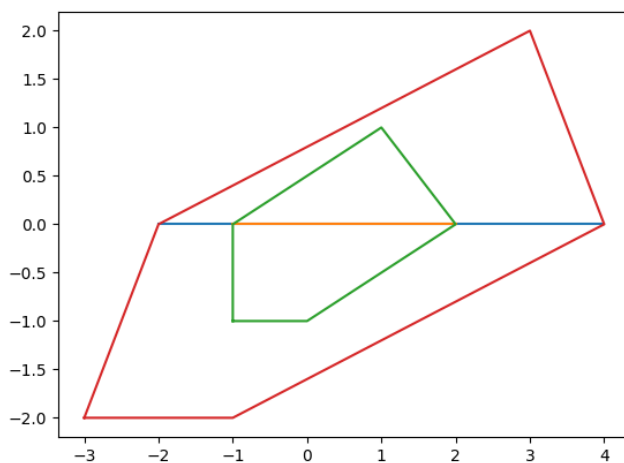
В начало: [1](#) К описанию матрицы: [9](#) Собственные векторы: [13](#)



Синий – оригинал, рыжий – его образ. Причём зелёный и красный отрезки остались параллельными, аналогично – фиолетовое и коричневое рёбра. Это произошло потому, что они коллинеарны собственным векторам матрицы данного отображения и не меняют направления, меняют только длину в соответствии с собственными числами.

3.12 12) !

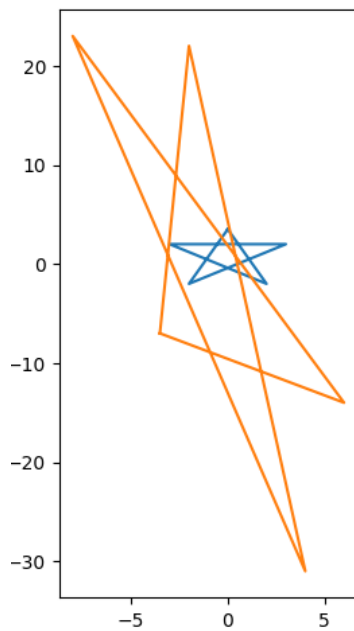
В начало: [1](#) К описанию матрицы: [9](#) Собственные векторы: [13](#)



Зелёный – оригинал, красный – его образ. У данного отображения только один собственный вектор, и вот, не изменили направление только рыжий отрезок, перейдя в синий.

3.13 13)

В начало: [1](#) К описанию матрицы: [9](#)

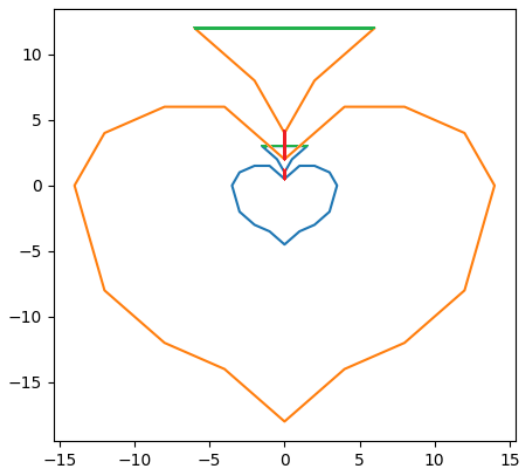


У этого отображения нет вещественных собственных векторов, то есть все векторы будут поворачиваться. Оригинал синий, его образ рыжий.

3.14 14) !

В начало: [1](#) К описанию матрицы: [10](#) Собственные векторы: [14](#)

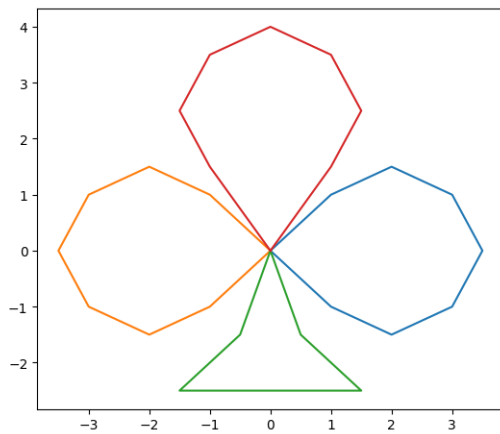
Можно заметить, что перемычка, соединяющая «треугольник» и «типо сердечко» не поменяла направления. Также горизонтальная часть «треугольника» не поменяла своего направления. Эти отрезки коллинеарны собственным векторам. (отмечены красным и зелёным)



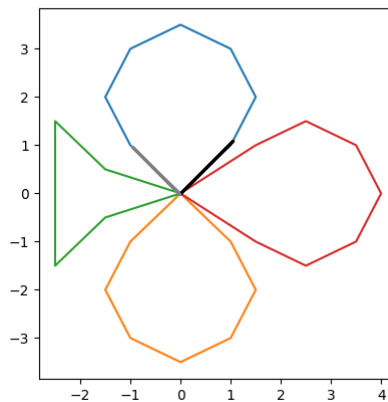
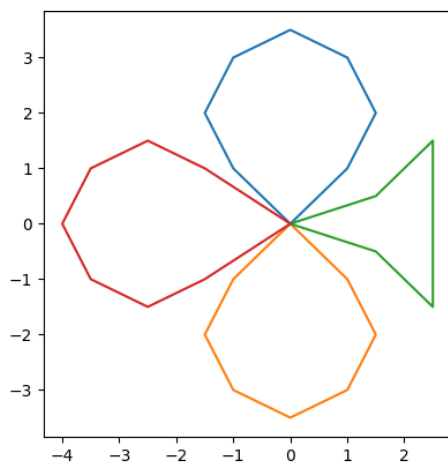
3.15 15) !

В начало: [1](#) К описанию матрицы: [10](#) Собственные векторы: [14](#)

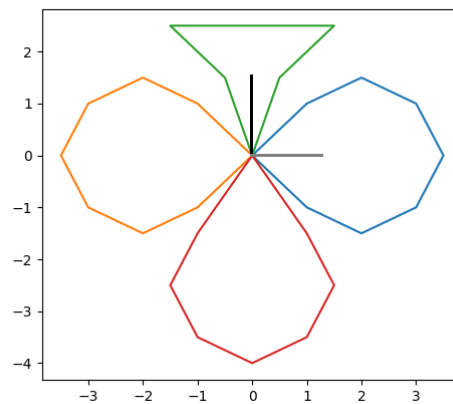
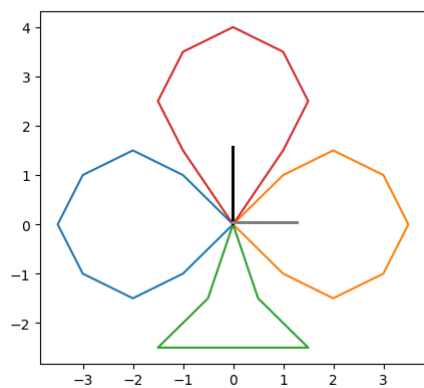
Исходное изображение:



Действие матриц A (нет вещ. собств. в-в), B (серый и чёрный)



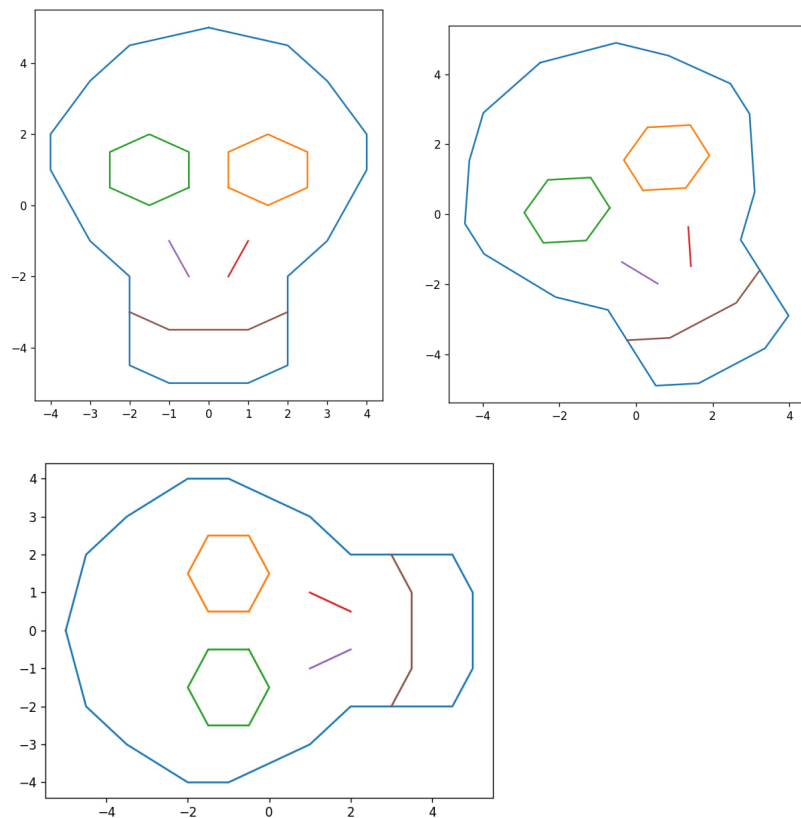
Действие матриц AB и BA (отражение по вертикали и по горизонтали)



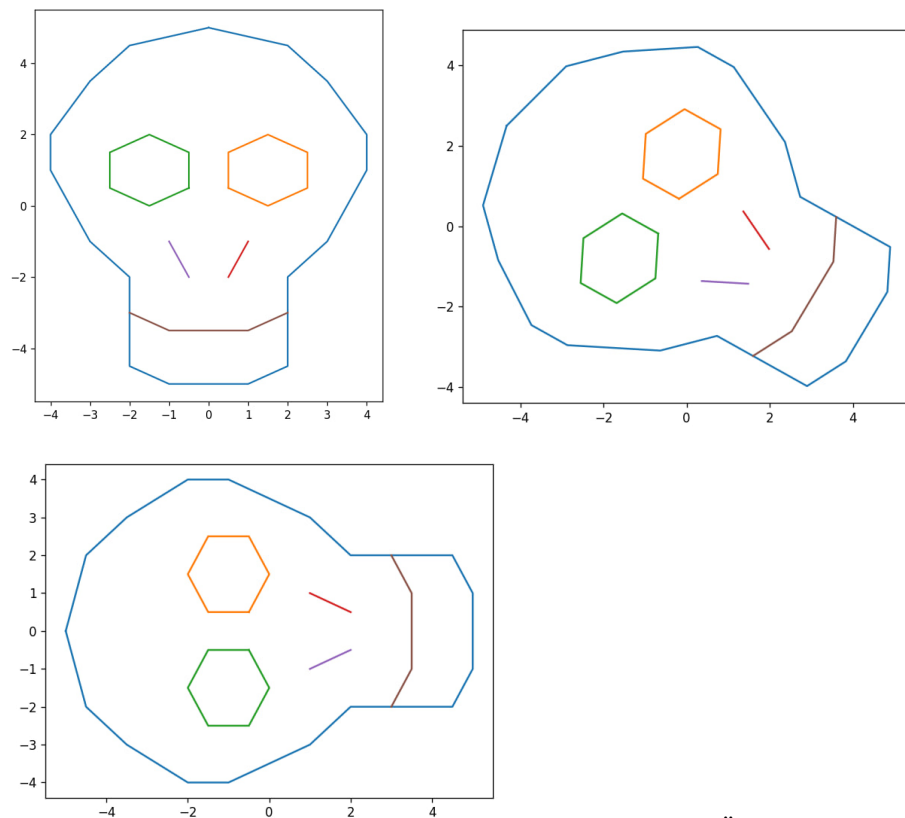
3.16 16) !

Собственные векторы этих отображений комплексные, поэтому изобразить их не представляется возможным.

Сначала на 30, потом на 60:



Сначала на 60, потом на 30:



я вертела это всё короче

4 Приложение (код)

В начало: [1](#).

А здесь мой уважаемый читатель может наконец увидеть небольшую выжимку из самых важных функций моего проекта.

4.1 Вспомогательные функции для подсчёта определителя (с++)

```
144  /*Calculates minor of the matrix.
145  *... Arguments:
146  *..... pointer to the matrix nxm:
147  *..... matrix* M.
148  *..... indexes of row and column to be deleted:
149  *..... size_t x, size_t y.
150  *... Returns:
151  *..... (matrix) -- minor matrix (n-1)x(m-1)
152  */
153  matrix::math::matr_minor(matrix* M, size_t x, size_t y) {
154      size_t n = M->n, m = M->m;
155      double** a = M->arr;
156      double** res = allocate_memory(n - 1, m - 1);
157      size_t i_m = 0, j_m;
158      for (size_t i = 0; i < n; i++) {
159          if (i == x) continue;
160          j_m = 0;
161          for (size_t j = 0; j < m; j++) {
162              if (j != y && i_m < (n - 1) && j_m < (m - 1)) {
163                  res[i_m][j_m] = a[i][j];
164                  j_m++;
165              }
166          }
167          if (i != x) {
168              i_m++;
169          }
170      }
171      return matrix{ n - 1, m - 1, res };
172  }
```

```

174  /* Calculates the determinant.
175  *...Arguments:
176  *...--pointer to the matrix:
177  *...--matrix *M.
178  *...Returns:
179  *...--(double)--the determinant.
180  */
181  double math::determinant(matrix* M){
182      size_t n = M->n;
183      if (n != M->m) {
184          std::cout << "This matrix is not squared, determinant cannot be calculated."
185          << std::endl;
186      }
187      double** a = M->arr;
188      if (M->n == 1) return a[0][0];
189      if (M->n == 2) return a[0][0] * a[1][1] - a[0][1] * a[1][0];
190
191      double res = 0, coef = 1;
192      for (size_t i = 0; i < n; i++) {
193          matrix cur_minor = matr_minor(M, 0, i);
194          res += coef * a[0][i] * determinant(&cur_minor);
195          coef *= -1;
196      }
197      return res;
198  }
199

```

4.2 Функции для работы с матрицами для первого задания (с++)

```

200  /* Calculates inverted matrix.
201  *...Arguments:
202  *...--pointer to matrix to be inverted:
203  *...--matrix *M.
204  *...Returns:
205  *...--(matrix)--the inverted matrix.
206  */
207  matrix math::matr_invert(matrix* M){
208      size_t n = M->n, m = M->m;
209      if (n != m){
210          std::cout << "This matrix cannot be inverted." << std::endl;
211          return matrix{ 0, 0, NULL };
212      }
213      double** arr = allocate_memory(n, m);
214      double coef;
215      //adjusted
216      for (size_t i = 0; i < n; i++) {
217          for (size_t j = 0; j < n; j++) {
218              if (i % 2 != j % 2) {
219                  coef = -1;
220              }
221              else coef = 1;
222              matrix minorr = matr_minor(M, i, j);
223              arr[i][j] = determinant(&minorr) * coef;
224          }
225      }
226      //transpose
227      matrix res = matr_transpose(matrix{ n, m, arr });
228      //multiply to inv to det
229      return num_mul_matr(1.0 / determinant(M), res);
230  }

```

```

39  /* Multiplies two matrices. Validates sizes.
40  *** Arguments:
41  *-----first matrix:
42  *-----matrix A,
43  *-----second matrix:
44  *-----matrix B.
45  *** Returns:
46  *----- (matrix) -- result of multiplication.
47  *** Notes:
48  *----- Returns {0, 0, NULL}, if matrices are incompatible.
49  */
50  matrix math::matr_mul_matr(matrix A, matrix B) {
51      size_t n = A.n, m = B.m, l = A.m;
52      // validate sizes
53      if (l != B.n) {
54          printf("Those matrices cannot be multiplied\n");
55          printf("A: %zux%zu, B: %zux%zu\n", n, l, B.n, m);
56          matrix res = {0, 0, NULL};
57          return res;
58      }
59      double** a = A.arr, ** b = B.arr;
60      double** res = allocate_memory(n, m);
61      // calculations
62      for (size_t i = 0; i < n; i++) {
63          for (size_t j = 0; j < m; j++) {
64              res[i][j] = 0;
65              for (size_t k = 0; k < l; k++) {
66                  res[i][j] += a[i][k] * b[k][j];
67              }
68          }
69      }
70      matrix matr = {n, m, res};
71      return matr;
72  }
73
232 /* There is an equation  $A * B = C$ . Get matrix A.
233 *** Arguments:
234 *-----matrix B:
235 *-----matrix B,
236 *-----matrix C:
237 *-----matrix C.
238 *** Returns:
239 *----- (matrix) -- matrix A.
240 */
241 matrix math::get_matrix(matrix B, matrix C) {
242     return matr_mul_matr(C, matr_invert(&B));
243 }

```

4.3 Функции для автоматического подсчёта образа многоугольника (python)

Многоугольник здесь может состоять из нескольких фигур. Его вбиваете в was. Также нужна сама матрица преобразования, M. Будет изображён оригинал и образ.

```
127 def countPoint(m, x, y):
128     X = m[0][0]*x + m[0][1]*y
129     Y = m[1][0]*x + m[1][1]*y
130     return [X,Y]
131
132 def getRes(frame, m):
133     framerexX = []
134     framerexY = []
135     for i in range(len(frame[0])):
136         tmp = countPoint(m, frame[0][i], frame[1][i])
137         framerexX.append(tmp[0])
138         framerexY.append(tmp[1])
139     return [framerexX, framerexY]
140
```

```
178 res = []
179 for i in was:
180     plt.plot(i[0],i[1])
181     res.append(getRes(i, M))
182
183 for i in res:
184     plt.plot(i[0],i[1])
185
186 plt.show()
187
```