Module 4

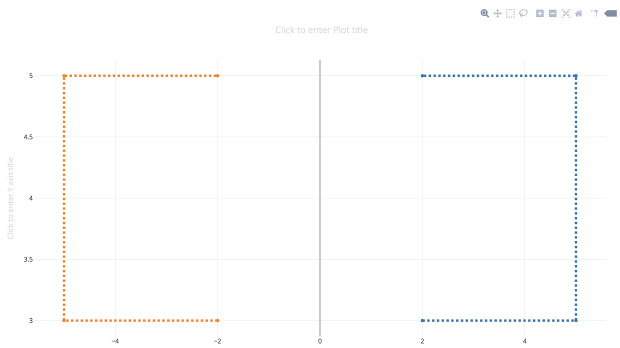
Aakash Shah
Team 4: Pause&Ponder*

February 22, 2018

1 A few examples and some questions they raise

1.1 In-Class Exercise 1

On a piece of paper, draw the points (2,3),(5,3),(5,5) and (2,5) and join the dots to get a shape. Now, treating each of these tuples as a 2D vector, multiply each separately by the matrix A = $\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$ to get four new vectors. Draw the "points" (heads) corresponding to these vectors. What is the geometric relationship between the two shapes?



$$\begin{bmatrix} 2 & 3 \end{bmatrix} * \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} -2 & 3 \end{bmatrix} \qquad \begin{bmatrix} 5 & 3 \end{bmatrix} * \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} -5 & 3 \end{bmatrix}$$

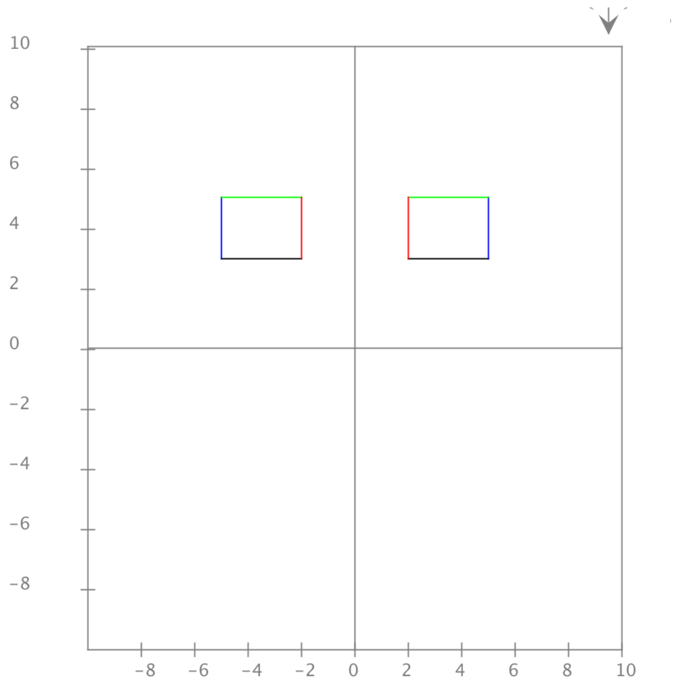
$$\begin{bmatrix} 5 & 5 \end{bmatrix} * \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} -5 & 5 \end{bmatrix} \qquad \begin{bmatrix} 2 & 5 \end{bmatrix} * \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} -2 & 5 \end{bmatrix}$$

These are meant to be vectors. The relationship between the two shapes is that they are just the mirror image of each other. The matrix A just rotates the vectors by an clockwise angle such that the magnitude does not change. And hence the -ve sign. The shape is reflecting the rectangle about the y-axis.

1.2 In-Class Exercise 2

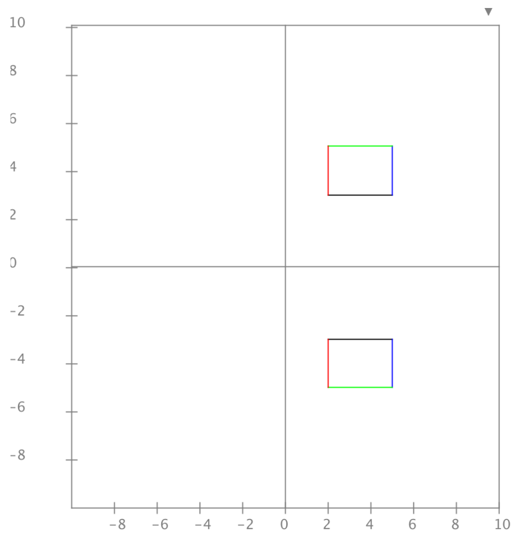
Download GeomTransExample.java and MatrixTool.java, and add your matrix-vector multiplication code from earlier to MatrixTool. Then, confirm your calculations above. You will also need DrawTool.java.

*Team Member: Rohan Shetty



1.3 In-Class Exercise 3

What matrix would resulting in reflecting a shape about the x-axis?



$$\begin{bmatrix} 2 & 3 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} 2 & -3 \end{bmatrix}$$

$$\begin{bmatrix} 5 & 3 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} 5 & -3 \end{bmatrix}$$

$$\begin{bmatrix} 5 & 5 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} 5 & -5 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 5 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} 2 & -5 \end{bmatrix}$$

1.4 In-Class Exercise 4

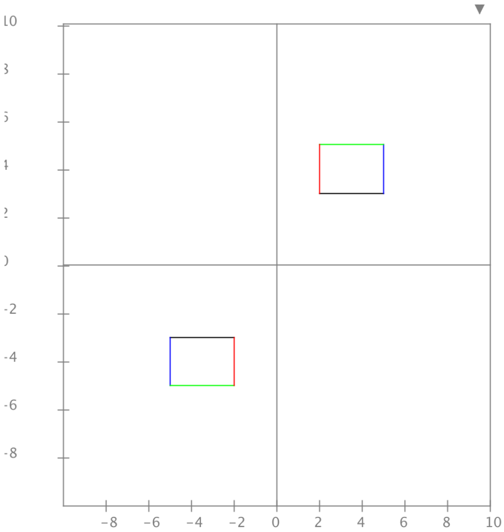
On paper, draw the same rectangle (the points (2,3),(5,3),(5,5),(2,5)) and reflect the rectangle about the origin. Use guesswork to derive the matrix that will achieve this transformation. Apply the matrix (call this matrix B) to the four corners to verify.

$$\begin{bmatrix} 2 & 3 \end{bmatrix} * \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} -2 & -3 \end{bmatrix} \qquad \begin{bmatrix} 5 & 3 \end{bmatrix} * \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} -5 & -3 \end{bmatrix}$$

$$\begin{bmatrix} 5 & 5 \end{bmatrix} * \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} -5 & -5 \end{bmatrix} \qquad \begin{bmatrix} 2 & 5 \end{bmatrix} * \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} -2 & -5 \end{bmatrix}$$

1.5 In-Class Exercise 5

Download GeomTransExample2.java, and modify the matrix B in the code to achieve reflection about the origin, and confirm your calculations above.

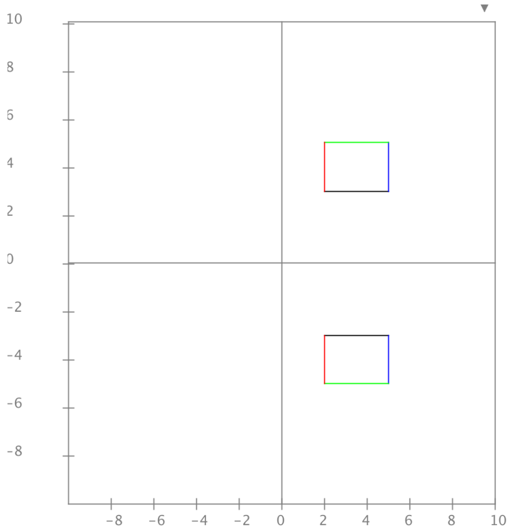


1.6 In-Class Exercise 6

Download GeomTransExample3.java, and insert the entries from the A and B matrices from above. Work out the product BA by hand and apply to the four corners of the rectangle to confirm what the program produces. Argue that the resulting matrix is intuitively what one would expect.

$$BA = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} * \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

It is true that it will show the same image as if it was reflecting about the x-axis because We rotated the vectors about the origin and then reflected along y axis which brings it to the 4th quadrant.



1.7 In-Class Exercise 7

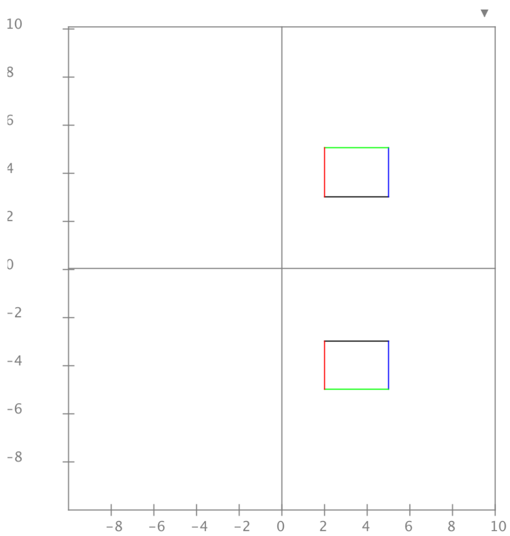
What theoretical property of multiplication do we need to be true for matrices to resolve questions 1 and 2 above?

The associativity property has to be true. Associative property of multiplication
$(AB)C = A(BC)$
$(AB)C = A(BC)$

1.8 In-Class Exercise 8

In your earlier program, GeomTransExample3.java, change the matrix multiplication order from the product BA to the product AB. What do you see? Does the order matter? Confirm by hand calculation. Is there a geometric reason to expect the result? What do you conclude about whether matrix multiplication is commutative?

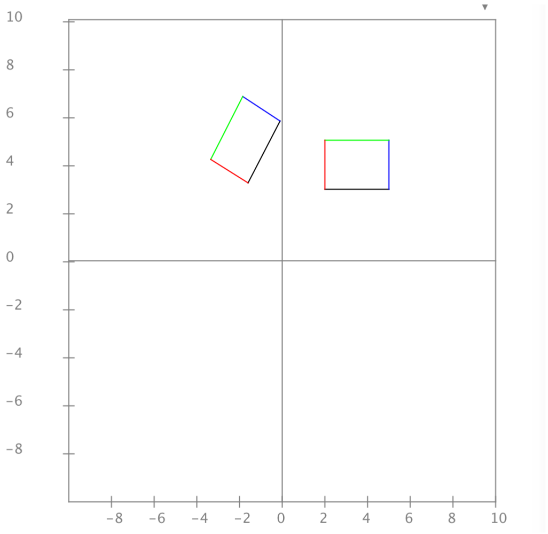
The above example will show that $BA = AB$ because when the vectors rotate they do not change the magnitude. They just change their directions. The property will not work when the two vectors undergo a rotation and a stretch/shrink. It is not necessary that if the vectors shrink/stretch first and then rotate, the result will not be the same.



1.9 In-Class Exercise 9

Download GeomTransExample4.java, compile and execute. How would you describe the transformation?

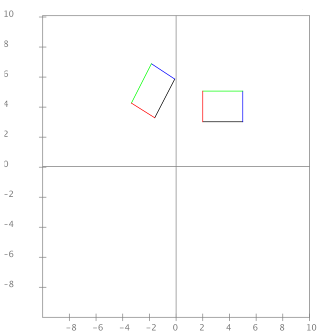
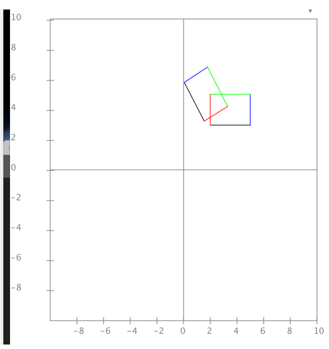
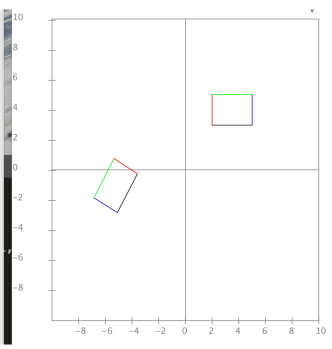
The above code produces a reflection about the y axis but there is a stretch/shrink included in it which causes the vector to change its magnitude along with its direction.



1.10 In-Class Exercise 10

Now, let's apply the earlier reflection about y-axis and the transformation above in sequence. In GeomTransExample4.java, first apply AC and then change to the order to CA. What do you see? Does the order matter? Is there a geometric reason to expect the result?

Yes, the order matters because the vectors undergo a stretch and rotation which is not necessarily equal to a vector undergoing rotation and a stretch.

1.11 In-Class Exercise 11

Download GeomTransExample5.java, compile and execute. Observe that we apply two transformations in sequence: first, a matrix A and then a matrix B. What is the net effect in transforming the rectangle? From that, can you conclude what matrix B would achieve when applied by itself to a vector? What is the product matrix $C = BA$ printed out? Consider the generic vector $v = (v1, v2)$ and compute by hand the product Cv.

The net effect in transforming this rectangle is 0. The effect of doing a transformation with its inverse is a identity matrix. The intuition is that when we apply a transforming operation the determinant is calculated which is basically the amount of area changed during the transformation; In this case the net result of change in area is 0 and hence the transformation just rotates it for one full cycle resulting in an Identity matrix.

Value of B $= A^{-1}$. The product of matrix $C = BA$ is :

$$\begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}$$

The C matrix is an Identity matrix which causes any generic vector in 2 dimensions to stay as is. The answer is :

$$\begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix} * \begin{bmatrix} v1 \\ v2 \end{bmatrix} = \begin{bmatrix} v1 \\ v2 \end{bmatrix}$$

2 Properties of matrix multiplication

2.1 In-class Exercise 13

Do the properties above directly imply that $(A + B)C = (AC) + (BC)$? Or is a separate proof needed?

We know that (A+B)C = AC + BC or BC + AC. We just proved that matrix multiplication distributes over matrix addition and hence, this relation will hold true.

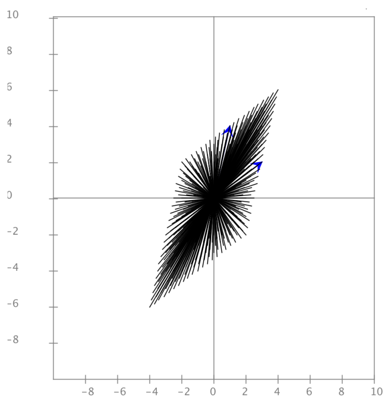
3 Two key ideas: span and basis

3.1 In-class Exercise 14

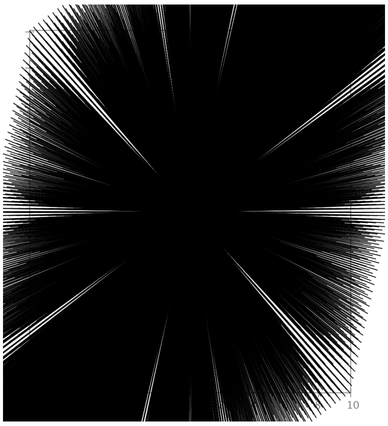
To explore this notion, write code in *ExploreSpan.java* to compute a linear combination. Observe the systematic exploration of different values of α, β. Change the range to see if you can "fill up" the space.

After visualizing the Explore Span and playing with it. We realize that changing the range of alpha and beta we can span the entire coordinate plan. The linear combination basically spans the whole coordinate system when these are distinct non-zero vectors, and they should not be multiples of each other which is essentially stretching the same vector in the same direction.

This is for the following 2 vectors. $\begin{bmatrix} 1 & 4 \\ 3 & 2 \end{bmatrix}$



This is for the alpha beta range from -5 to 5

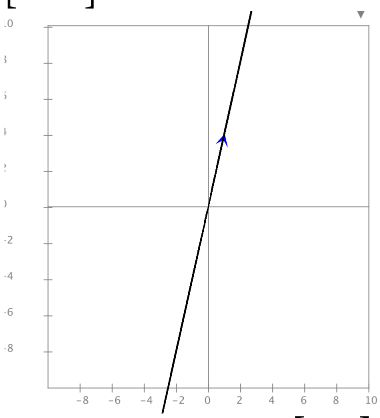


3.2 In-class Exercise 15

What's an example of vectors u,v whose span is not the whole space of 2D vectors? Similarly, what's an example of 3D vectors u1,u2,u3, whose span is not the whole space of 3D vectors?

This is for the following 2 vectors, The span is just the straight line and not the whole 2D space.

$$\begin{bmatrix} 1 & 4 \\ 3 & 12 \end{bmatrix}$$



Example of 3d vectors : $\begin{bmatrix} 1 & 4 \\ 3 & 2 \\ 0 & 0 \end{bmatrix}$

3.3 In-class Exercise 16

Is there a pair of 3D vectors that spans the whole space of 3D vectors?

Example of 3d vectors : $\begin{bmatrix} 1 & 4 \\ 3 & 2 \\ 5 & 6 \end{bmatrix}$

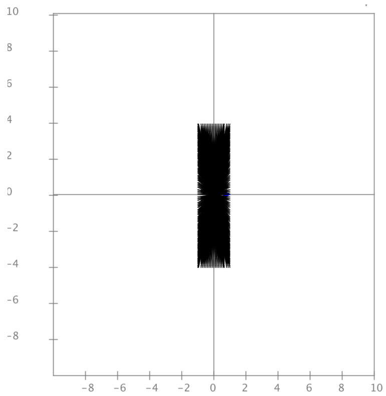
3.4 In-class Exercise 17

Consider the pair of vectors $e1 = (1,0), e2 = (0,1)$.
1. Express the vector (1,4) as a linear combination of e1,e2.
2. Express the vector (2,3) as a linear combination of e1,e2.
3. Do e1,e2 span 2D space?
4. What are the corresponding three vectors that span 3D space?

Answer :

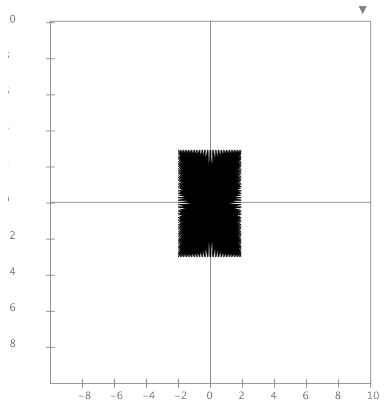
1. Basis vector in our world * $\begin{bmatrix} 1 \\ 4 \end{bmatrix}$ = Vector in our world

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 \\ 4 \end{bmatrix} = 1 * \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 4 * \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \end{bmatrix}$$

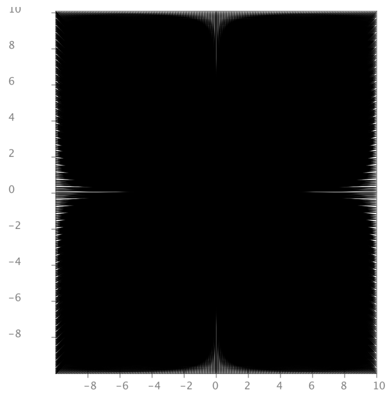


2. Basis vector in our world * $\begin{bmatrix} 1 \\ 4 \end{bmatrix}$ = Vector in our world

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} * \begin{bmatrix} 2 \\ 3 \end{bmatrix} = 2 * \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 3 * \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$



3.



4.

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

4 Understanding how to build your own transforms

4.1 In-class Exercise 18

Explain why this is so. That is, why do we get (cos(θ),sin(θ)) and (-sin(θ),cos(θ)) ?

The unit circle along with the basis vectors of the real world making a unit triangle in the co-ordinate system. To know the angle of a vector we use the trigonometric functions to calculate θ. The Pythagoras principle helps to calculate the length of the hypotenuse. The sin, cos covers the x and y axis respectively.

4.2 In-class Exercise 19

Use the step-by-step approach above to derive transformation matrices for:
1. Clockwise rotation by θ.
2. Reflection about the x-axis.

1. Clockwise rotation by θ.

$$\begin{bmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & -cos(\theta) \end{bmatrix}$$

2. Reflection about the x-axis

$$\begin{bmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & -cos(\theta) \end{bmatrix}$$

4.3 In-class Exercise 20

What happens when the approach is applied to translation? That is, suppose we want each point (x, y) to be translated to $(x + 1, y + 2)$.

When we apply rotation to translation, the coordinate system moves to another location to the point it is translated to. If each point (x,y) is translated to (x + 1, y + 2)

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x+1 \\ y+2 \\ 1 \end{bmatrix}$$

The coordinate system changes to 3-Dimensional.

4.4 In-class Exercise 21

Prove that no matrix multiplication can achieve translation. That is, no 2×2 matrix can transform every vector (x , y) to (x + p , y + q).

When we apply rotation to translation, the coordinate system moves to another location to the point it is translated to. If each point (x,y) is translated to (x + p, y + q)

$$\begin{bmatrix} 1 & 0 & p \\ 0 & 1 & q \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x+p \\ y+q \\ 1 \end{bmatrix}$$

The coordinate system changes to 3-Dimensional.

5 The affine trick (for translation)

5.1 In-class Exercise 22

Consider some 2D matrix $\mathbf{C} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$ applied to a 2D vector (x,y). What is the resulting vector? What is the result of applying $\begin{bmatrix} c_{11} & c_{12} & 0 \\ c_{21} & c_{22} & 0 \\ 0 & 0 & 1 \end{bmatrix}$ to (x,y,1)?

Answer :

$$\mathbf{C} * \mathbf{v} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} * \begin{bmatrix} x \\ y \end{bmatrix} = x \begin{bmatrix} c_{11} \\ c_{21} \end{bmatrix} + y \begin{bmatrix} c_{12} \\ c_{22} \end{bmatrix} = \begin{bmatrix} xc_{11} + yc_{12} \\ xc_{21} + yc_{22} \end{bmatrix}$$

This is the resulting vector. This is the translated coordinates based on the basis vectors given in C. This is the coordinates of the vectors position in our world. So that we can see how does it look in the 2D space where the basis vectors are C.

Similarly, for 3Dimensions if we apply the above concept :

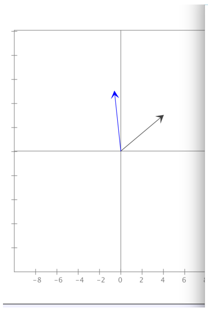
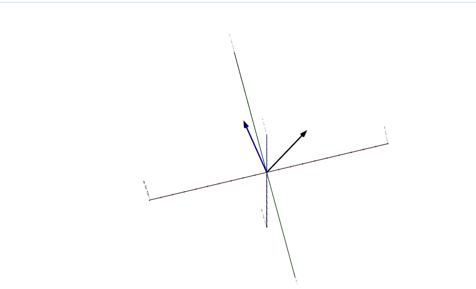
$$\mathbf{C} * \mathbf{v} = \begin{bmatrix} c_{11} & c_{12} & 0 \\ c_{21} & c_{22} & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = x \begin{bmatrix} c_{11} \\ c_{21} \\ 0 \end{bmatrix} + y \begin{bmatrix} c_{12} \\ c_{22} \\ 0 \end{bmatrix} + 1 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} xc_{11} + yc_{12} \\ xc_{21} + yc_{22} \\ 1 \end{bmatrix}$$

It did not make any difference but extended the space from 2D to 3D.

5.2 In-class Exercise 23

Examine the code in AffineExample.java, then compile and execute. Then, do the same for Affine3DExample.java. Move the viewpoint so that you see the (x,y) axes in the usual way (looking along the z-axis), and compare with the 2D drawing.

Answer :

5.3 In-class Exercise 24

Show that this is indeed the case.

To prove that $\mathbf{BAu} = \text{proj}(\text{affine}(\mathbf{B})\ \text{affine}(\mathbf{A})\ \text{affine}(\mathbf{u}))$
Let us assume values for all the variables.

$$\text{B} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}; A = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}; u = \begin{bmatrix} 9 \\ 10 \end{bmatrix}$$

$$\text{affine(B)} = \begin{bmatrix} 1 & 2 & 0 \\ 3 & 4 & 0 \\ 0 & 0 & 1 \end{bmatrix}; affine(A) = \begin{bmatrix} 5 & 6 & 0 \\ 7 & 8 & 0 \\ 0 & 0 & 1 \end{bmatrix}; affine(u) = \begin{bmatrix} 9 \\ 10 \\ 1 \end{bmatrix}$$

$$\text{proj(affine(B) * affine(A) * affine(u))} = \begin{bmatrix} 1 & 2 & 0 \\ 3 & 4 & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 5 & 6 & 0 \\ 7 & 8 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 19 & 22 & 0 \\ 43 & 50 & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 9 \\ 10 \\ 1 \end{bmatrix} = \begin{bmatrix} 391 \\ 887 \\ 1 \end{bmatrix} = \begin{bmatrix} 391 \\ 887 \end{bmatrix}$$

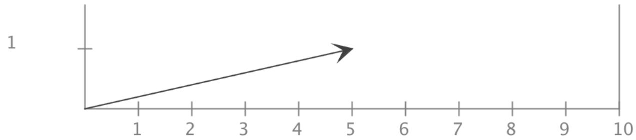
$$\mathbf{BAu} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} * \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} * \begin{bmatrix} 9 \\ 10 \end{bmatrix} = \begin{bmatrix} 391 \\ 887 \end{bmatrix}$$

Hence proved ■

5.4 In-class Exercise 25

Examine AffineExample2.java to see how rotation (by $60\,\text{deg}$) followed by translation by (3,4) works via combining the results into a single matrix, and applying it to the point (5,1). Compile and execute. It's confusing to see what translation does, so it's best to draw an arrow from the shifted origin, (3,4), to the new coordinates.

This vector is $\mathbf{u} \begin{bmatrix} 5 \\ 1 \end{bmatrix}$



A transformation C is applied to this vector. If we decompose the transformation C, it looks like

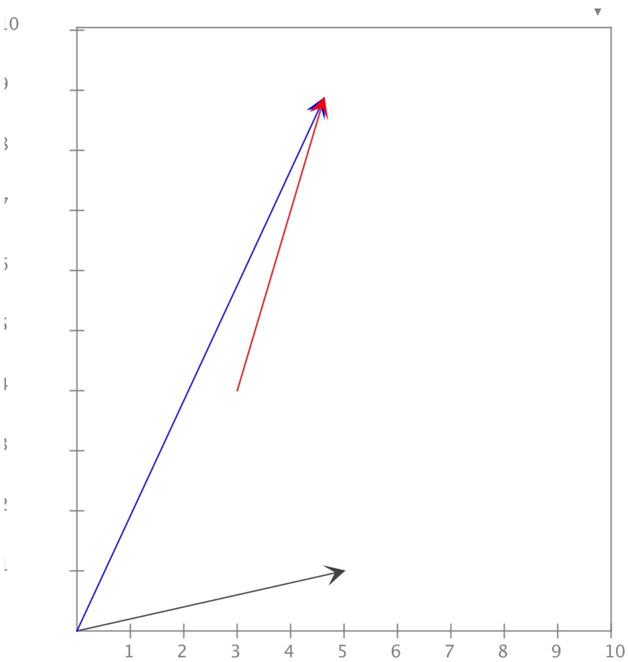
$$\begin{bmatrix} 0.500 & -0.866 & 0.000 \\ 0.866 & 0.500 & 0.000 \\ 0.000 & 0.000 & 1.000 \end{bmatrix}$$

It is a result of a matrix multiplication or translation of BA:

```
double[][] A = {
                {cos(), -sin(), 0},
                {sin(), -cos(), 0},
                {0,     0,      1}
};
// Translation by (3,4)
double[][] B = {
                    {1, 0, 3},
                    {0, 1, 4},
                    {0, 0, 1}
};

double[][] C = MatrixTool.matrixMult(B, A);
```

This means that the angle of rotation is specified in Matrix A and the Matrix B is almost an identity matrix with a 3rd vector in 3D space. This transformation leads to the rotation of the vector by 60deg in the positive/anticlockwise direction. With it stretches the 3rd vector B times. Here is the final outcome.



5.5 In-class Exercise 26

Prove the above result. Is it true that $AI = A$ Is it possible to have an identity matrix for an $m \times n$ matrix?

Intuition behind an identity matrix is that it is a matrix of basis vectors in the n dimensional space we are working with. And it is obvious that a vector which is extended only in its dimension will give us a nxn Identity matrix. It is usually preferred and is also a compulsion to have the Identity matrix as nxn. It also has a property of commutativity because of its multiply-compatibility. For proving $AI = A$

$$A * I = I * A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = A$$

Hence proved. ■

6 Matrix multiplication: a different view

6.1 In-class Exercise 27

Why is this true?

When we multiply two matrices, the first row of first matrix is multiplied to the first column and then the second, the third and so forth. So we just represent each column of B in the form of b_i

$$\mathbf{A}\begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \mathbf{b}_1 & \mathbf{b}_2 & \cdots & \mathbf{b}_n \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \mathbf{c}_1 & \mathbf{c}_2 & \cdots & \mathbf{c}_n \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

Since, each matrix multiplication follows the same rule of multiplication, we represent the solution as a collection of columns, ci.

7 The reverse of a transformation

7.1 In-class Exercise 28

What is the "undo" matrix for the reflection about the y-axis? Multiply the undo matrix (on the left) and the original.

The undo matrix for the reflection about the y-axis is the same as the original matrix. Since anything reflected about the y-axis on the 2nd quadrant resides in 1st quadrant and vice versa. So multiplying the undo matrix with the original matrix results in an Identity matrix.

Original Matrix $= \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$

Undo Matrix $= \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$

Undo Matrix * Original Matrix $= \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} * \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I$

Which proves that the undo matrix is an Inverse matrix. That basically gives the inverse effect of a transformation.

7.2 In-class Exercise 29

What is the "undo" affine-extended matrix for translation by (p,q)? Multiply the undo matrix (on the left) and the original.

$$\text{Affine extended matrix} = \begin{bmatrix} 1 & 0 & p \\ 0 & 1 & q \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x+p \\ y+q \\ 1 \end{bmatrix}$$

$$\text{Undo Matrix} = \begin{bmatrix} 1 & 0 & -p \\ 0 & 1 & -q \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x-p \\ y-q \\ 1 \end{bmatrix}$$

$$\text{Undo Matrix * Original Matrix} = \begin{bmatrix} 1 & 0 & -p \\ 0 & 1 & -q \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & p \\ 0 & 1 & q \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = I$$

8 Change of coordinate frame

8.1 In-class Exercise 30

Follow the steps to compute the affine-extended matrix that, when applied to the point (4,3), produces the coordinates in the new frame. Confirm by implementing in CoordChange2.java.

The affine extended matrix is : $\begin{bmatrix} 0.5 & 0.886 & 0 \\ -0.866 & 0.5 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ is then applied to the vector v $\begin{bmatrix} 4 \\ 3 \\ 1 \end{bmatrix}$ which results in $\begin{bmatrix} 4.598 \\ -1.964 \\ 1 \end{bmatrix}$

And CoordChange2 also results in the same result

```
Matrix (3x3):
  0.500  0.866  0.000
 -0.866  0.500  0.000
  0.000  0.000  1.000
Vector:  4.598 -1.964  1.000
```

8.2 In-class Exercise 31

Use the matrix for B^{-1} you computed earlier, and insert the matrix in CoordChange3.java to compute the coordinates in the new frame.

Discuss in class.

8.3 In-class Exercise 32

Is $A^{-1}B^{-1} = B^{-1}A^{-1}$? Add a few lines of code to CoordChange3.java to see. Now, go back to the picture with the three frames above. If we were to first rotate the standard frame by 60$\circ$ and then translate by (1,2), would the resulting frame be the

same as if we were to first translate and then rotate? Why then do we get different results when applying a different order to the inverse matrices?

Discuss in class.

8.4 In-class Exercise 33

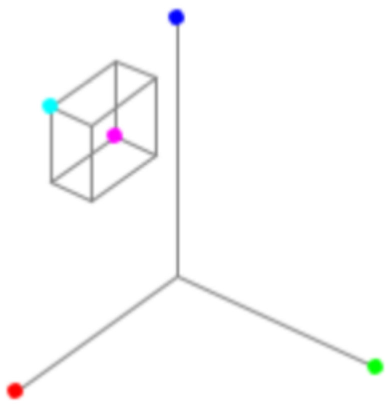
Compile and execute CoordChange3D.java to see a cuboid drawn along with the position of an eye. Now move the view so that the eye lines up with the origin. Where do you see the cuboid? This is the 2D view we will build.

After we execute the program and rotate the figure to align with our eye position, the image appears to be in a 2D plane.

8.5 In-class Exercise 34

Examine the code in CoordChange2D.java to see all the transforms implemented and multiplied. Compile and execute to see that we do indeed get the desired view.

The output looks like this :



8.6 In-class Exercise 35

Verify this by hand.

We have to verify this equation: $v = B\alpha u$ stretches the length by α.

$\begin{bmatrix}\alpha & 0\\ 0 & \alpha\end{bmatrix} \cdot \begin{bmatrix}u_1\\ u_2\end{bmatrix} = \begin{bmatrix}\alpha u_1\\ \alpha u_2\end{bmatrix} = \alpha \begin{bmatrix}u_1\\ u_2\end{bmatrix}$

Since we know that the coordinates are u1,u2 and then multiplied by α will increase the length.

8.7 In-class Exercise 36

What happens when v=u in $\mathbf{v}.\mathbf{u} = |\mathbf{v}||\mathbf{u}|\cos\theta$?

if v = u

$\cos\theta = \frac{u.u}{|u||u|}$

$\cos\theta = \frac{u^2}{|u||u|}$

$\cos\theta = \frac{1}{|u||u|}$

$\cos\theta = 1$

$\theta = 0\,\text{deg}$

The angle between both the vectors is 0 degrees.

8.8 In-class Exercise 37

Implement dot product and norm in MatrixTool and test with NormExample.java.

```
u dot v = 41.0
|u| = 8.06225774829855 |v|=5.830951894845301
```

The above example proves the property.

8.9 In-class Exercise 38

Verify that the columns of the reflection and rotation matrices are orthogonal vectors. Are they orthonormal too?

These columns are orthogonal because the rotations and reflections matrix will be n dimensional and perpendicular to each other.

8.10 In-class Exercise 39

If A is an orthogonal matrix, what is $A^T A$? What does this say about the inverse of A?

If A is an orthogonal matrix, It means that the vectors lie in n-dimensional space. These vectors are a multiple of the basis vectors and hence each vector is extended in its own dimensions. The matrix A looks like : $\begin{bmatrix} \vdots & \vdots & \vdots \\ x_1 & x_2 & x_3 \\ \vdots & \vdots & \vdots \end{bmatrix}_{?*n}$

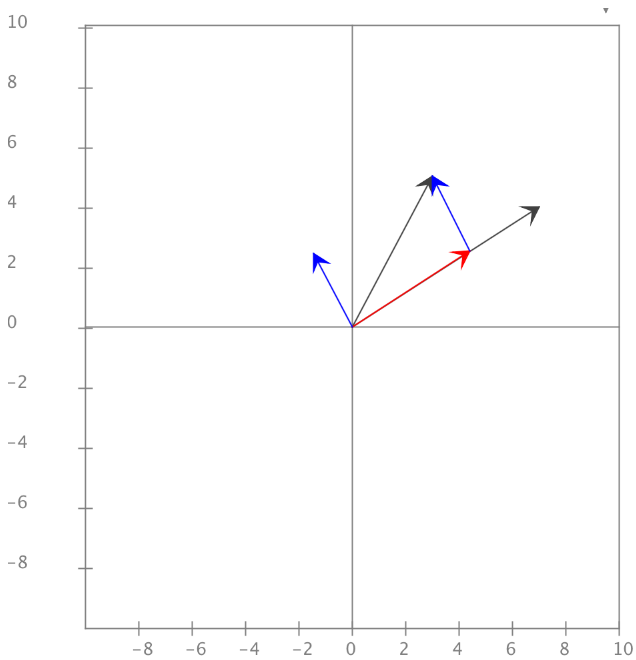
Every column is pairwise perpendicular, which means they are all mutually perpendicular. We know that it has n vectors which are mutually perpendicular. So the columns are n, Every pair of vector has to have a dot product of 0 for the $\cos\theta = 90\,\text{deg}$ That means it will have $n-1$, 0 vectors. This is the reason it results in an Identity matrix. In this case $A^{-1} = A^T$.

8.11 In-class Exercise 40

Implement projection in MatrixTool and test with ProjectionExample.java. Confirm that you get the same vectors as in the diagram above.

The output is :

```
Vector:   4.415   2.523
Vector:  -1.415   2.477
u dot z = -1.7763568394002505E-15
```



9 Complex vectors

9.1 In-class Exercise 41

What is the sum of $(1+2i, i, -3+4i, 5)$ and $(-2+i, 2, 4, 1+i)$?

By the above definition:
$\mathbf{u}+\mathbf{v} = (u_1+v_1, u_2+v_2, \ldots, u_n+v_n)$ where $\mathbf{u} = (u_1, \ldots, u_n)$ and $\mathbf{v} = (v_1, \ldots, v_n)$

$=>$ $(1+2i, i, -3+4i, 5) + (-2+i, 2, 4, 1+i)$
$=>$ $(1+2i-2+i, i+2, -3+4i+4, 5+1+i)$
$=>$ $((-1+3i), i+2, (1+4i), (6+i))$
$=>$ This is complex addition!

9.2 In-class Exercise 42

What is the scalar product of $\alpha = (1-2i)$ and $u = (1+2i, i, -3+4i, 5)$?

By the above definition:
$\alpha\mathbf{u} = \alpha(u_1, u_2, \ldots, u_n) = (\alpha u_1, \alpha u_2, \ldots, \alpha u_n)$

but here $\alpha = v$ and
$\mathbf{u}\cdot\mathbf{v} = u_1\overline{v_1} + u_2\overline{v_2} + \ldots + u_n\overline{v_n}$

$=>$ $(1+2i)1+2i \ + \ (1+2i)i \ + \ (1+2i)-3+4i \ + \ (1+2i)5$
$=>$ $(1+2i)^2 \ + \ i+2i^2 \ + \ -3-6i+8i^2+4i \ + \ 5+10i$
$=>$ $(1+2i)^2 \ + \ i-2 \ + \ -3-6i-8+4i \ + \ 5+10i$
$=>$ $-3+4i \ + \ i-2 \ + \ -11-2i \ + \ 5+10i$
$=>$ $-11+13i$

The squared magnitude of the complex number a+bi is $\sqrt{a^2+b^2} = \sqrt{121+169} = \sqrt{290}$

=> This is a scalar product!

9.3 In-class Exercise 43

Work out the two products $(a+bi)(a+bi)$ and $(a+bi)(a-bi)$

=> $(a+bi)(a+bi)$
=> $(a+bi)^2$
=> $a^2+(bi)^2+2abi$
=> a^2-b^2+2abi
=> **So,** $a^2+b^2 \neq (a+bi)(a+bi)$ ■

=> $(a+bi)(a-bi)$
=> $a^2-(bi)^2)$
=> $a^2-b^2(-1)$
=> a^2+b^2
=> **So,** $a^2+b^2=(a+bi)(a-bi)$ ■

9.4 In-class Exercise 44

For the complex vector $z=(z1,z2,z3)=(1+2i,i,5)$, compute both $z \cdot z$ and $z_1^2+z_2^2+z_3^2$

=> $z=(z1,z2,z3)=(1+2i,i,5)$
=> $z.z=(z1,z2,z3).(z1,z2,z3)=(1+2i,i,5).(1+2i,i,5)$
=> $z.z=(1+2i)(1-2i)+(i)(-i)+(5)(5)$
=> $z.z=(1-(2i)^2)-i^2+25$
=> $z.z=5+1+25$ $\quad |u_1|^2+|u_2|^2+\ldots+|u_n|^2$
=> $z.z=(5)+26$
=> $z.z=31$

Since, $\mathbf{u}\cdot\mathbf{u}=(u_1,u_2,\ldots,u_n)\cdot(\overline{u_1},\overline{u_2},\ldots,\overline{u_n})$
=> $u_1\overline{u_1}+u_2\overline{u_2}+\ldots+u_n\overline{u_n}$
=> $|u_1|^2+|u_2|^2+\ldots+|u_n|^2$
=> $|\mathbf{u}|^2$

=> $z_1^2+z_2^2+z_3^2$
=> $(1+2i)(1+2i)+(i)(i)+(5)(5)$
=> $5+1+25=31$

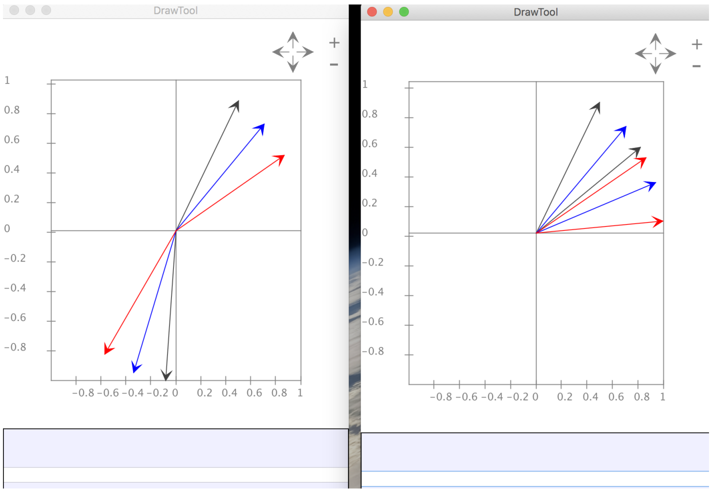
=> $(1+2i)^2+24$
=> $(1-4+4i)+24$
=> $21+4i$

10 Exploring the action of some matrices on vectors

10.1 In-class Exercise 45

Examine the code in OrthoExplore.java to confirm the generation method. You will also need UniformRandom.java (for this, and other exercises). Run a few times to see the results. Confirm that the matrix rotates the vectors but does not change their length.

When a matrix undergoes rotation, the matrix does not change the magnitude or length of the vector. It just alters the direction of the vectors. This property is due to the angular transformation of the matrix. The dot product of matrix A with the vector u leads to vector v. Which is just a phase/angle shift and same magnitude. We can see it in the following image.



10.2 In-class Exercise 46

Examine the code in OrthoExplore2.java to confirm the above approach. You will also need Function.java and SimplePlotPanel.java. Run a few times to see the results. Explain the graph of α vs. θ

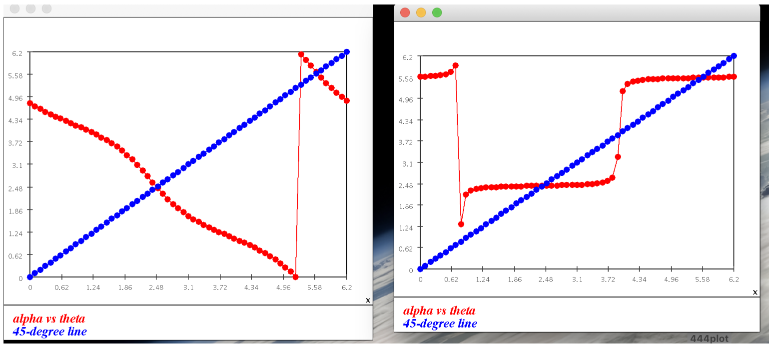
The following code shows that alpha and theta are independednt of the length of the vectors. An orthogonoal matrix like A will be a square matrix with n dimensions and each vector has n parts.

```
A = random orthogonal matrix
        for \theta =0 to 2 \pi
           u = (cos\theta, sin\theta)
           z =  A u
           \theta = angle of z
           Add (\theta, \alpha) to data set
       endfor
       Plot data set
```

10.3 In-class Exercise 47

Examine the code in MatrixExplore.java to confirm the above approach. Run a few times to see the results. What do you conclude about the points of intersection of the curve and the line?

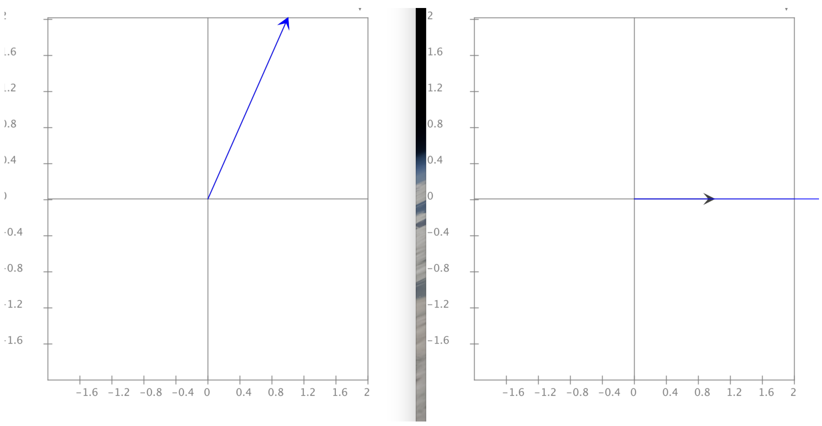
The output of the code is :



10.4 In-class Exercise 48

Examine the code in MatrixExplore2.java to see that it's a simple example of a matrix that transforms the vector $u = (1, 2)$ into another vector. Compile and execute to draw the two vectors. Then try $u = (1, 0)$

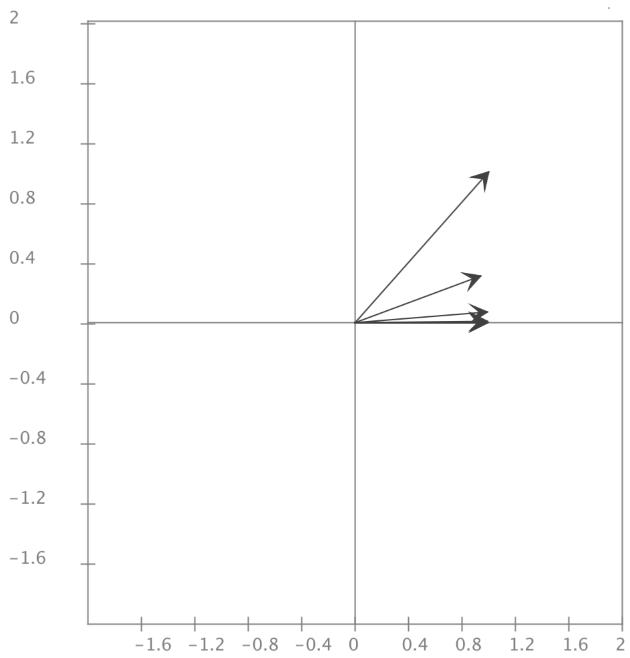
The two vectors show :



10.5 In-class Exercise 49

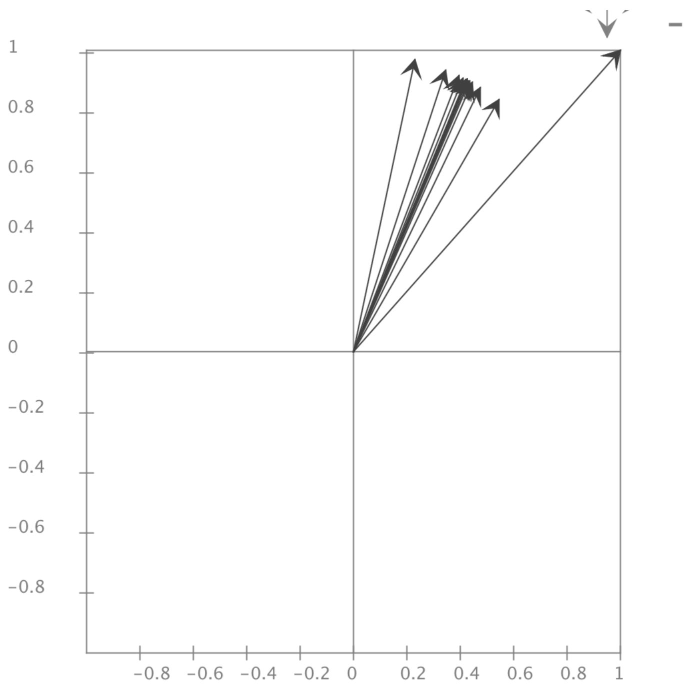
Examine the code in MatrixExplore3.java to see that the same matrix as in the previous exercise iteratively multiplies as above. What do you observe? Try a larger number of iterations, and different starting vectors.

Since the vectors are getting normalized, the number of iterations literally does not make a difference. The starting vector changes the orientation.



10.6 In-class Exercise 50

The program MatrixExplore4.java produces a random matrix each time it's executed. This matrix is applied iteratively to a starting vector. How often do you see fixed points emerging?



Very often.