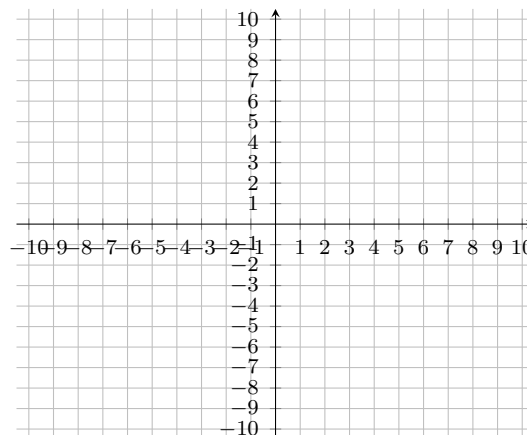


THE GEOMETRY OF ROW OPERATIONS AND SOLUTION SETS

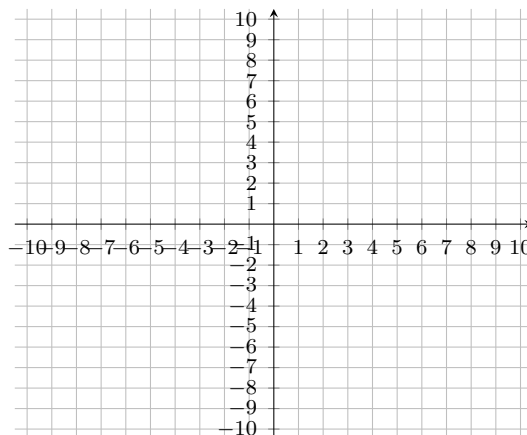
Practice using row operations to solve the following linear systems in two variables. After you determine the solution set, graph the original two equations to verify the number of solutions.

Students should attempt the highlighted problem BEFORE attending the live session on Sept. 12.

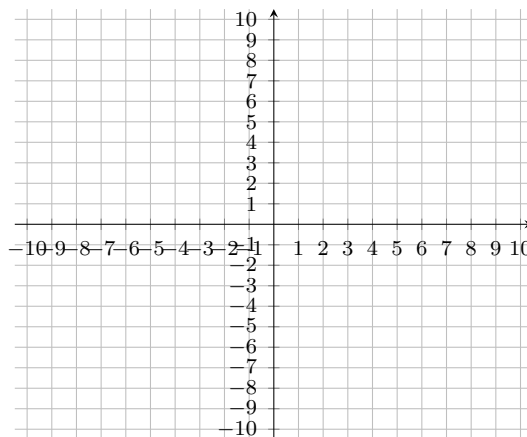
$$\begin{array}{rclcl} 6x & - & 3y & = & 3 \\ 2x & - & y & = & 4 \end{array}$$



$$\begin{array}{rclcl} 2x & - & y & = & 9 \\ x & + & 3y & = & 8 \end{array}$$



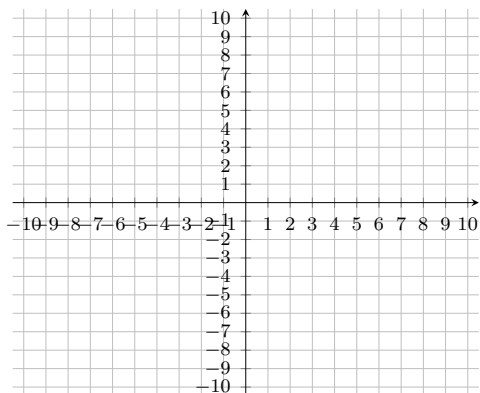
$$\begin{array}{rclcl} 4x & + & 3y & = & 6 \\ 12x & + & 9y & = & 18 \end{array}$$



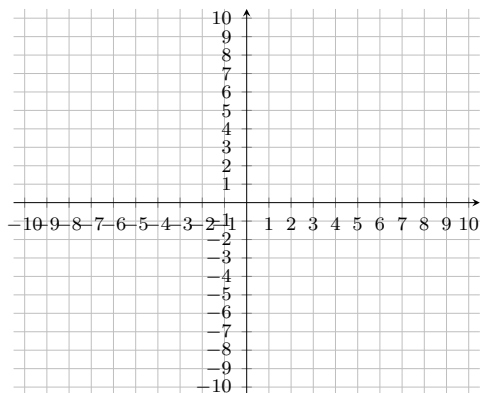
Let's take a closer look at how row operations effect the geometry of the solution set. We'll examine why the point of intersection remains the same. Consider the system:

$$\begin{aligned} 2x + 4y &= 16 \\ 3x + y &= 4 \end{aligned}$$

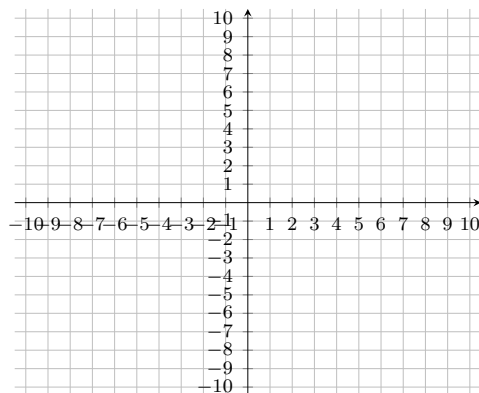
First, graph the original system of linear equations:



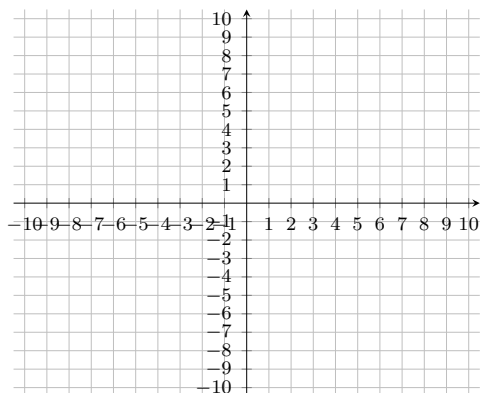
Now, perform each row operation and graph the linear system you obtain:



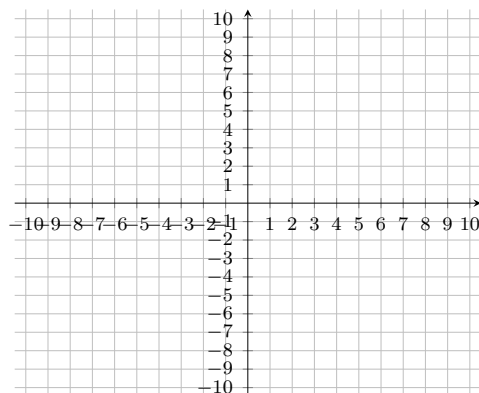
$$\frac{1}{2}(\text{Equation 1})$$



$$-3(\text{Equation 1}) + (\text{Equation 2})$$



$$-\frac{1}{5}(\text{Equation 2})$$



$$-2(\text{Equation 2}) + (\text{Equation 1})$$

HOW DOES GPS WORK?

Imagine you are lost in the woods, and you must rely on GPS (Global Positional System) to help find your way out. We will explore how GPS locates your position using a system of linear equations.

To help with our computations, we will make some initial assumptions. Even though the earth is an irregularly shaped ellipsoid, we will assume the earth is a unit sphere (this will help simplify the computations when we are solving the system by hand). So, the equation for our earth will be:

$$x^2 + y^2 + z^2 = 1.$$

(Hint: remember this equation at the end!) Time will be measured in hundredths of seconds. Before we get started with the set-up of the system, let's take a minute to explore how GPS actually works.

The basic idea of GPS is to use radio signals to contact satellites at known locations and to use this information to identify your current location. The time that it takes for these radio signals from the satellites to hit the GPS receiver is used to compute the distances. A location on our "unit sphere" earth can then be determined using three different satellites.

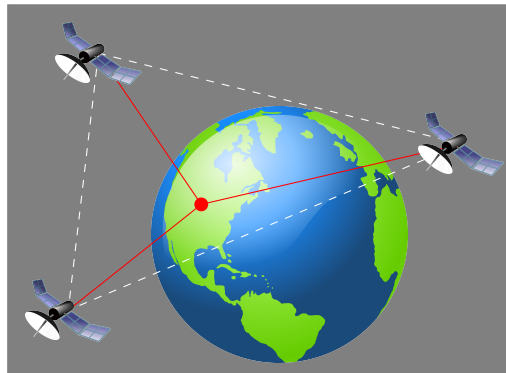
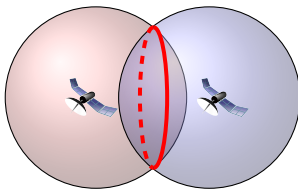
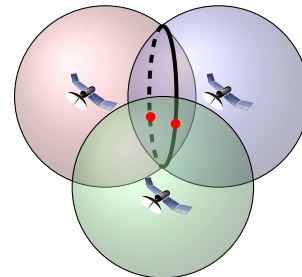


FIGURE 1. Three satellites in space, with the path of the radio signals indicated in red.

In reality, our earth is not a unit sphere. So, in practice, four satellites are used to account for the altitude of our current location. The data from each satellite puts us in a sphere around that satellite. So, data from two satellites will give us an infinite collection of possible locations, all lying on the circle of intersection of the two spheres (see Figure A). Data from three satellites will give us two possible location points (see Figure B). However, if we have four satellites, the intersection



(A) The intersection of two spheres emanating from satellites is a circle.



(B) The intersection of three spheres emanating from satellites is a set consisting of two points.

will be a single location point. You might remember that previously it was mentioned that on our “unit sphere” earth, only three satellites would be needed. This is because the “unit sphere” earth would be considered as the fourth sphere. We will practice the linear algebra behind the GPS using four satellites, to keep things closer to reality.

To figure out our location, we will need four different satellites. This data is given in the following chart (with positions of each satellite given in 3-dimensional coordinates as (x, y, z)), along with the time the signal took to reach each satellite and return back to our GPS receiver (given in seconds):

Satellite	Position	Time
1	(1.11, 2.55, 2.14)	1.29
2	(2.87, 0.00, 1.43)	1.31
3	(0.00, 1.08, 2.29)	2.75
4	(1.54, 1.01, 1.23)	4.06

We will let:

(x, y, z) : our position in the woods
 t : the time when the signal arrives to our GPS receiver

We can then use that $d = rt$, where the rate r is the traveling speed of the signal (and is assumed to be 0.47 Earth radii/ 10^{-2} sec. Let's use this with the first satellite's information:

Satellite	Position	Time
1	(1.11, 2.55, 2.14)	1.29

Then we get:

$$d = rt = 0.47(t - 1.29).$$

We can also express distance using the distance formula for two points in three-space, namely (x_1, y_1, z_1) and (x_2, y_2, z_2) :

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}.$$

Using the first satellite's information, we get:

$$d = \sqrt{(x - 1.11)^2 + (y - 2.55)^2 + (z - 2.14)^2}.$$

Combining the two ways to calculate the distance, d , we get:

$$0.47(t - 1.29) = \sqrt{(x - 1.11)^2 + (y - 2.55)^2 + (z - 2.14)^2}.$$

Now, doing some algebra (and rounding all decimals to two decimal digits), we get:

$$\begin{aligned} 0.47(t - 1.29) &= \sqrt{(x - 1.11)^2 + (y - 2.55)^2 + (z - 2.14)^2} \\ [0.47(t - 1.29)]^2 &= (x - 1.11)^2 + (y - 2.55)^2 + (z - 2.14)^2 \\ 0.22(t - 1.29)^2 &= (x - 1.11)^2 + (y - 2.55)^2 + (z - 2.14)^2 \\ 0.22(t^2 - 2.58t + 1.66) &= x^2 - 2.22x + 1.23 + y^2 - 5.1y + 6.50 + z^2 - 4.28z + 4.58 \\ 0.22t^2 - 0.57t + 0.37 &= x^2 - 2.22x + 1.23 + y^2 - 5.10y + 6.50 + z^2 - 4.28z + 4.58 \\ 2.22x + 5.10y + 4.28z - 0.57t &= x^2 + y^2 + z^2 - 0.22t^2 + 11.94 \end{aligned}$$

This isn't a linear equation, but have faith! Before we continue, your first task is to do generate similar equations for each of the other three satellites. Record them in the following table:

Satellite 1	$2.22x + 5.10y + 4.28z - 0.57t = x^2 + y^2 + z^2 - 0.22t^2 + 11.94$
Satellite 2	
Satellite 3	
Satellite 4	

So, how do we get a linear system from these?? Notice that the nonlinear (the squared terms) are all the same! Here, we will see the reason why we needed four satellites. We are going to get a linear system by subtracting satellite 1's equation from each of the other three equations to give us a system of three linear equations. Record those differences in the following table:

(Satellite 2)-(Satellite 1)	
(Satellite 3)-(Satellite 1)	
(Satellite 4)-(Satellite 1)	

Write down your system of three linear equations in the variables x , y , z , and t :

Now, solve this system using your graphing calculator or a computer:

x	=	
y	=	
z	=	

(★)

Substitute your solutions for x , y , and z back into the original equation for Satellite 1:

$$0.47(t - 1.29) = \sqrt{(x - 1.11)^2 + (y - 2.55)^2 + (z - 2.14)^2}$$

and simplify it down ... you should get a quadratic equation in the variable t ! Solve that equation!

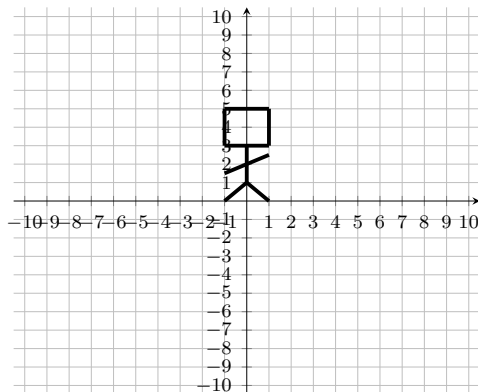
Quadratic equation with variable t :
Solutions: $t =$

Now check these values of t in the (\star) equations to find the possible positions (x, y, z) . Record the possible positions here:

Which of these positions is our correct position? And why is the other one incorrect?

COMPUTER ANIMATION - activity designed by **Rachelle Bouchat**

Let's look at the idea of animating an object using linear transformations. Let me introduce my good friend Blocko. He is a highly rated actor and has starred in my award-winning films.



Notice that Blocko is defined by the coordinate pairs that make up his shape, namely

$$\{(-1, 0), (0, 1), (1, 0), (0, 3), (-1, 1.5), (1, 2.5), (-1, 3), (-1, 5), (1, 5), (1, 3)\}.$$

We can think of any movement of Blocko as:

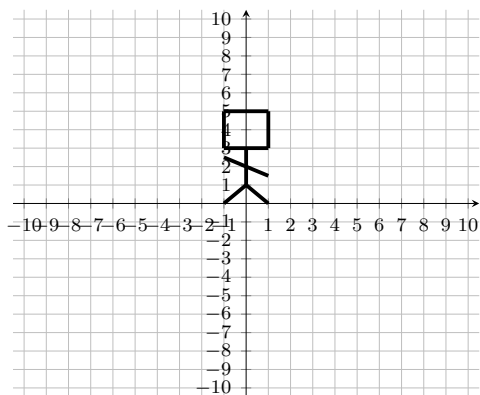
$$(1) \quad \begin{aligned} x_{\text{new}} &= ax_{\text{old}} + by_{\text{old}} + c \\ y_{\text{new}} &= dx_{\text{old}} + ey_{\text{old}} + f \end{aligned}$$

where $(x_{\text{old}}, y_{\text{old}})$ represents the starting coordinates of the movement and $(x_{\text{new}}, y_{\text{new}})$ represents the ending coordinates of the movement (also, $a, b, c, d, e, f \in \mathbb{R}$). We want to think of this as a linear transformation $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$. To do this, write System (1) in the form:

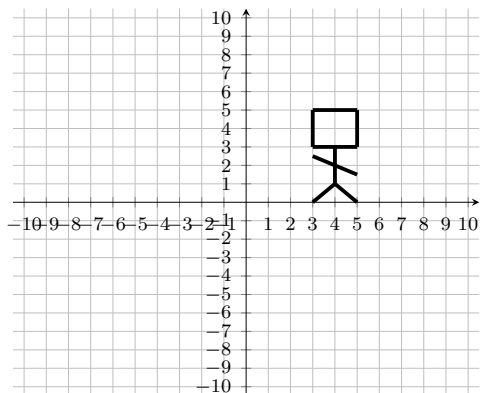
$$M\vec{v} = \begin{bmatrix} x_{\text{new}} \\ y_{\text{new}} \end{bmatrix},$$

where M is a matrix (hint: you might need a 1 in the vector \vec{v}):

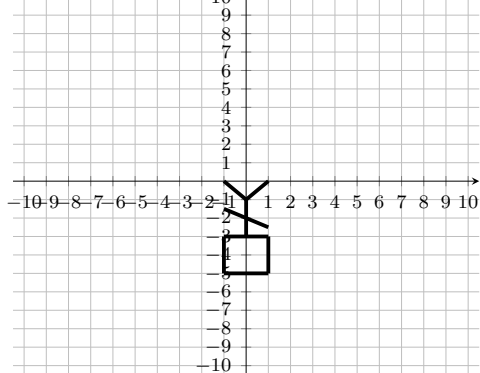
Determine the linear transformation $T(\vec{v}) = M\vec{v}$ that would make Blocko end in this position:



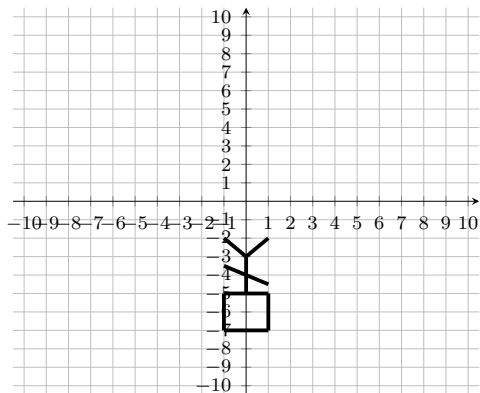
Determine the linear transformation $T(\vec{v}) = M\vec{v}$ that would make Blocko end in this position:



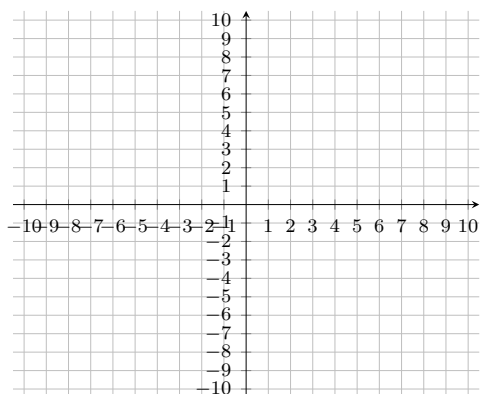
Determine the linear transformation $T(\vec{v}) = M\vec{v}$ that would make Blocko end in this position:



Determine the linear transformation $T(\vec{v}) = M\vec{v}$ that would make Blocko end in this position:



Determine the linear transformation $T(\vec{v}) = M\vec{v}$ that would rotate Blocko by 45° counterclockwise. Use that matrix and computation to draw a picture of Blocko.



CODING THEORY: LINEAR BLOCK CODES

What exactly is *coding theory*? Every day, we send information over the internet (credit card numbers for purchases, passwords, etc). Much of this data is sensitive, so we don't want other people to be able to get it easily. What happens is that the data we send, like our credit card number, is changed the numbers are modified and extra characters are added to the end before the data is sent. This process is called *encoding*. It is important for us to modify the data sent so that people can't steal the data in the middle of the transmission. However, we want to make sure that the intended recipient gets the correct information. So, we have to be able to explain to the intended recipient how to figure out the original information sent. This process is called *decoding*.

There are also other things that might happen in the transmission. For instance, maybe we have a bad internet connection and there ends up being a small mistake in the transmission. It would be great if the recipient could not only catch the mistake, but also correct it. This area of study on "how to safely transfer data and possible corrected errors" is called *coding theory*. In this activity, we will look at how adding extra digits to a *message* can be used to detect and possibly correct errors using matrix operations.

To work through this activity, we first need to learn how to work in the binary number system, i.e. with the set $\{0,1\}$. So, we will deal with *strings* (or sequences) of zeros and ones. When we add them, we do it component-wise, using that $1 + 1 = 0$, $0 + 0 = 0$, and $1 + 0 = 1$. Let's look at an example:

$$\begin{array}{rccccccc} & 1 & 0 & 1 & 1 & 1 & 0 \\ + & 1 & 1 & 0 & 1 & 0 & 1 \\ \hline 0 & 1 & 1 & 0 & 1 & 0 & 1 \end{array}$$

Now, you give it a try!

$$1101101 + 1111111 =$$

$$1110110 + 0010101 =$$

$$0111101 + 1010101 =$$

$$1011101 + 1001111 =$$

The set-up for coding theory is that we want to send a message, so we change that message into a codeword (so the information is secure). We then send it over the internet (and errors might happen along the way). The codeword that is received needs to be decoded. Hopefully, we are able to recover the correct original message:



The way we encode a message is through matrix multiplication (but with our new rules for addition). Here is an example of a *code matrix* C which takes a message of length 3 and turns it into a codeword of length 7:

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

To encode the message 011, we compute $[0 \ 1 \ 1] C$:

$$[0 \ 1 \ 1] C = [0 \ 1 \ 1] \cdot \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix} = [0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0].$$

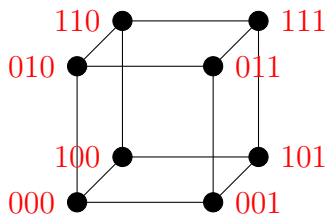
So, we encoding the message 011 as the codeword 0111010. If we consider all possible messages that have length three and use only 0's and 1's, we get: 000, 001, 010, 100, 011, 101, 110, and 111. We encoded the message 011 above, so now encode the other seven messages using the matrix C :

000		011	0111010
001		101	
010		110	
100		111	

What do you notice about the first 3 digits of each codeword? Why do you think that happens?

So, in our simple coding example, we could just take a received word like 0111010 ... and remove the last four digits to recover the original message. But, what if there was an error?! Can we know there was an error? Can we fix that error?

Let's consider all the possible messages of length 3, we can label them as the vertices of a cube as follows:



Then the shortest distance between the nodes (the number of edges you have to travel on) on the cube is the *Hamming distance*. For example:

$$\begin{aligned} d(010, 110) &= 1 \\ d(010, 101) &= 3 \end{aligned}$$

Find the following Hamming distances:

$$d(100, 111) =$$

$$d(110, 100) =$$

$$d(101, 010) =$$

$$d(001, 100) =$$

Notice that this is the same as just counting how many digits they differ by. From above, we have $d(010, 101) = 3$ and if we compare them digit by digit, all three are different:

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

Similarly, we had $d(010, 110) = 1$ and if we compare the digits, only one of the digits is different:

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

Use this idea to compute the distance between the following codewords:

$$d(0010111, 1001011) =$$

$$d(0010111, 1011100) =$$

$$d(1110001, 1001011) =$$

$$d(1110001, 0010111) =$$

Copy the codewords that you found on page 3 for each of the eight possible messages:

Message	Codeword
000	
001	
010	
100	
011	
101	
110	
111	

Now, imagine we tried to send the message 010, so we sent the codeword 0101101. But, there was an error in the transmission, and the person received the word 0111101. Can the person still figure out what we sent (if they just took the first three digits, they wouldn't get the correct message)?

We are going to take the received word 0111101 and calculate the distance from each of the possible codewords (in the chart above). We will assume that the one with the smallest distance is the intended codeword, because this would be the one with the fewest errors. Let's see how this works by calculating the distance between 0111101 and each of the possible codewords:

$$\begin{aligned} d(0111101, 0000000) &= 5 \\ d(0111101, 0010111) &= 3 \\ d(0111101, 0101101) &= 1 \\ d(0111101, 1001011) &= 5 \\ d(0111101, 0111010) &= 3 \\ d(0111101, 1011100) &= 3 \\ d(0111101, 1100110) &= 5 \\ d(0111101, 1110001) &= 3 \end{aligned}$$

Since there is a unique codeword that has a smallest distance, we will assume that the codeword was 0101101, and so we recovered the intended message of 010 (by removing the last four digits from our decoded message). So, our matrix multiplication code not only found the error (we knew, because it wasn't one of our eight codewords), it corrected the error!!!

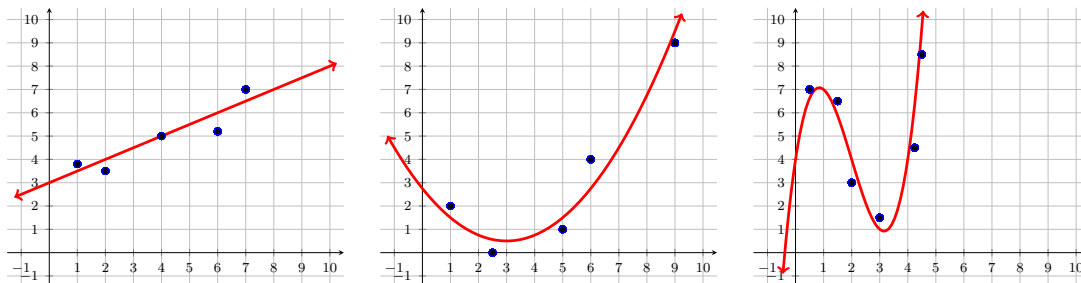
See if you can figure out the intended message for each of the following received words (in each case there is at least one error). In some cases, you might not be able to decode the message.

Received Word	Intended Message
1110110	
1011011	
1111011	
0111101	

Coding theory is a current area of mathematical research. It is also known as *algebraic coding theory*. There are lots of interactions with linear algebra, and the computations are generally done over a finite field (which are intimately related to prime numbers).

CURVE FITTING

Given a collection of data points, $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, we want to obtain a mathematical formula $y = f(x)$ that best “fits” the data. What do we mean by “fit”?? We mean a curve where the distance between the data points and points on our curve is minimized. Here is what that might look like in pictures (the blue points make up the data set, and the red curve is the best “fit” curve for each data set):



Let’s start by thinking about finding a best fit line, $y = mx + b$, for a collection of data points $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$. If the data points are all on the same line (i.e. collinear), then we could just solve the system of linear equations:

$$\begin{aligned} y_1 &= mx_1 + b \\ y_2 &= mx_2 + b \\ &\vdots \\ y_n &= mx_n + b \end{aligned}$$

(Well, in reality, we would only need two of those equations). We can rewrite this system as:

$$\begin{array}{ccc} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} & = & \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix} \\ \uparrow & & \uparrow \quad \uparrow \\ \vec{y} & = & M \quad \vec{v} \end{array}$$

If the points are all collinear, then:

$$\vec{y} - M\vec{v} = \vec{0}$$

has a solution for the vector \vec{v} . If the data points are not collinear, then it is impossible to find such a vector \vec{v} . In this case, we want the vector $\vec{y} - M\vec{v}$ to be as close to $\vec{0}$ as possible. To measure this “closeness,” we will use the Euclidean length. For a vector $\vec{v} = \begin{bmatrix} a \\ b \end{bmatrix}$, the Euclidean length of \vec{v} is

$$\|\vec{v}\| := \sqrt{a^2 + b^2}$$

Notice that the Euclidean length is really just the length of the vector.

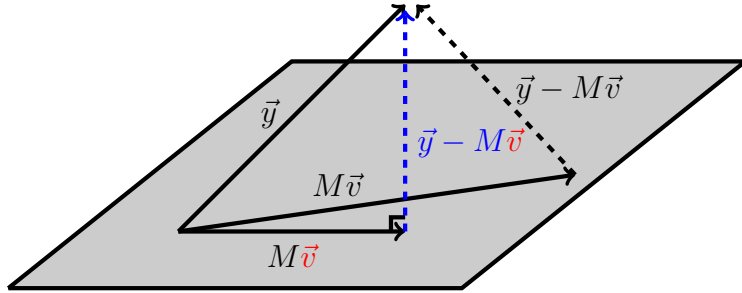
To practice, Calculate each of the following Euclidean lengths.

- $\|[2, 5]^T\| =$
- $\|[-3, 4]^T\| =$

We want to find a vector \vec{v} that minimizes the Euclidean length

$$\|\vec{y} - M\vec{v}\|.$$

So, how do we do this?? Notice that the collection of vectors given by $M\vec{v}$ form a subspace of \mathbb{R}^n (it is actually the column space of the matrix M).



Let \vec{v} be the vector such that $\vec{y} - M\vec{v}$ is orthogonal to the column space, i.e.

$$(M\vec{v})^\top(\vec{y} - M\vec{v}) = 0$$

for all vectors \vec{v} . Then it follows that

$$\begin{aligned} (M\vec{v})^\top(\vec{y} - M\vec{v}) &= 0 \\ (\vec{v})^\top M^\top(\vec{y} - M\vec{v}) &= 0 \\ (\vec{v})^\top (M^\top\vec{y} - M^\top M\vec{v}) &= 0 \end{aligned}$$

for all vectors \vec{v} . But, this is only possible if

$$M^\top\vec{y} - M^\top M\vec{v} = \vec{0}.$$

So,

$$M^\top\vec{y} = M^\top M\vec{v}.$$

Since M is an $n \times 2$ matrix, the matrix $M^\top M$ is a 2×2 matrix. This means that if $M^\top M$ is invertible, then the system has a unique solution, namely:

$$\vec{v} = (M^\top M)^{-1} M^\top\vec{y}$$

(Note: $M^\top M$ is not invertible if and only if the n data points lie on a vertical line in the xy -plane.)

Let's practice and find the least squares line to fit the four data points

$$\{(0, 1), (1, 3), (2, 4), (3, 4)\}.$$

- Step #1: Find the matrix M .

$$M =$$

- Step #2: Find $\vec{v} = (M^\top M)^{-1} M^\top\vec{y}$.

$$\vec{v} =$$

- Step #3: Determine the best fit line, $y = mx + b$.

$$y =$$

Generalize this process to find a best fit quadratic curve (i.e. a curve of the form $y = ax^2 + bx + c$) to the five data points

$$\{(0.1, -0.26), (0.2, 0.3), (0.3, 1.17), (0.4, 2.34), (0.5, 3.83)\}.$$

- Step #1: Find the matrix M .

$$M =$$

- Step #2: Find $\vec{v} = (M^T M)^{-1} M^T \vec{y}$.

$$\vec{v} =$$

- Step #3: Determine the best fit quadratic curve, $y = ax^2 + bx + c$.

$$y =$$

GENETIC INHERITANCE WITH EIGENVALUES & EIGENVECTORS



Let’s look at the genetic inheritance of traits in animals and plants. A greenhouse owner is growing snapdragons (pictured above) and wants to crossbreed red and white snapdragons. We will use eigenvalues and eigenvectors to see what will happen to the colors of snapdragons the greenhouse owner will have in the long term. First, we need to look to biology for some information about dominant and recessive genes. The inherited trait considered will come from a set of two genes:

- $A \longleftarrow$ dominant gene (red snapdragons)
- $a \longleftarrow$ recessive gene (white snapdragons)

The population will possess two of these genes, so the possibilities are: AA , Aa , aA , and aa . Since order doesn’t matter, we really just have three possibilities: AA , Aa , and aa . This pair of genes is called the individual’s *genotype*, and it determines how the trait controlled by the gene manifests in the individual. For the snapdragons, we have:

- $AA \longleftarrow$ red flowers
- $Aa \longleftarrow$ pink flowers
- $aa \longleftarrow$ white flowers

Now, let’s think about probabilities of inheriting each trait. For the snapdragons:

- If both snapdragon “parents” have genotype AA (both red genes), then there is a 100% chance that the offspring will have genotype AA (and visually have red flowers).

	A	A
A	AA	AA
A	AA	AA

- If one of the snapdragon parents has genotype AA (both red genes) and the other parent has genotype aa (both white genes), then the offspring are guaranteed to have genotype Aa (and visually have pink flowers).

	A	A
a	Aa	Aa
a	Aa	Aa

Continue this reasoning to fill in the probabilities (as decimals) for the rest of the possibilities.

	$AA - AA$	$AA - Aa$	$AA - aa$	$Aa - Aa$	$Aa - aa$	$aa - aa$
AA	1		0			
Aa	0		1			
aa	0		0			

The greenhouse owner wants to undertake a breeding program in which each plant in the population is always fertilized with a red snapdragon (genotype AA). We want to be able to find what fraction of the snapdragon population will be red (genotype AA), pink (genotype Aa), and white (genotype aa) after n generations ($n = 0, 1, 2, 3, \dots$). To begin, set

$$\begin{aligned} r_n &= \text{fraction of the red snapdragons (genotype } AA) \text{ in the } n\text{-th generation} \\ p_n &= \text{fraction of the pink snapdragons (genotype } Aa) \text{ in the } n\text{-th generation} \\ w_n &= \text{fraction of the white snapdragons (genotype } aa) \text{ in the } n\text{-th generation} \end{aligned}$$

Using this notation, r_0 , p_0 , and w_0 are the initial fractions of red, pink, and white snapdragons (respectively). We also know that for the n -th generation, we have:

$$r_n + p_n + w_n = 1.$$

We can use the table of probabilities we created above to determine the genotype distribution of the next generation (remember, we are intentionally breeding with red snapdragons, genotype AA , so we are only looking at the first three columns of our table). The first equation that is written says that all of the offspring bred with a plant of genotype AA (red) will be of genotype AA (red) and half of the offspring bred with a plant of genotype Aa (pink) will be of genotype AA (red). Determine equations for p_n and w_n :

$$r_n = r_{n-1} + \frac{1}{2}p_{n-1}$$

$$p_n =$$

$$w_n =$$

Write this system as a matrix equation, $M \begin{bmatrix} r_{n-1} \\ p_{n-1} \\ w_{n-1} \end{bmatrix} = \begin{bmatrix} r_n \\ p_n \\ w_n \end{bmatrix}$:

We want to find an expression for M^n . To do this, we are going to factor M as:

$$M = PDP^{-1},$$

where D is a diagonal matrix. This is called a *diagonalization* of the matrix. To do this, we need to find the eigenvalues and the eigenvectors of M . Record the eigenvalues and a basis for each corresponding eigenspace here:

Form the diagonal matrix D as follows:

$$D = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} =$$

Now, form the matrix P by putting an eigenvector that goes with λ_1 in the first column of P , an eigenvector that goes with λ_2 in the second column of P , and an eigenvector that goes with λ_3 in the third column of P . Then use it to find P^{-1} :

$$P = \left[\begin{array}{|c|c|c|} \hline \text{ } & \text{ } & \text{ } \\ \hline \end{array} \right] = \qquad P^{-1} =$$

Write down the diagonalization, $M = PDP^{-1}$, and verify that it is true by computing PDP^{-1} :

Now, let's investigate D . Calculate D^2 , D^3 , and D^4 to find a pattern for D^n . Record your expressions for D^n below:

$$D^n =$$

Now, let's see how this helps us compute M^n . In your calculator, compute M^n and PD^nP^{-1} for $n = 1, 2, 3$, and record your findings below:

	M^n	PD^nP^{-1}	How do they compare?
n=1			
n=2			
n=3			

Make a conjecture about M^n .

Let's put this all together to see what will happen to the snapdragon population in the long run. First, compute M^n using your conjecture:

What will the values of the entries of M^n approach in the long run as n gets very large?

What does this say about the fraction of the snapdragons that will be red (r_n), pink (p_n), and white (w_n) in the long term?

PLTL MATH 171 - RANKING SPORTS TEAMS

How can we use eigenvalues and eigenvectors to rank sports teams or players? Let's assume that there are five tennis players that we want to rank. They all played each other once in a round-robin tennis tournament, with the outcomes of these games detailed below:



- Player 1: Won against players 2, 4, and 5
- Player 2: Won against players 3, 4, and 5
- Player 3: Won against player 1 and 4
- Player 4: Won against player 5
- Player 5: Won against player 3

We want to use this data to rank the five players from best to worst. To do this, we first encode the win-loss data in a matrix (place a 0 in entry (i, j) if player i lost to player j , and place a 1 in entry (i, j) if player i beat player j):

$$\begin{array}{c}
 \text{Player 1} \rightarrow \\
 \text{Player 2} \rightarrow \\
 \text{Player 3} \rightarrow \\
 \text{Player 4} \rightarrow \\
 \text{Player 5} \rightarrow
 \end{array}
 \left[\begin{array}{ccccc}
 \swarrow \text{Player 1} & & & & \\
 \swarrow \text{Player 2} & & & & \\
 \swarrow \text{Player 3} & & & & \\
 \swarrow \text{Player 4} & & & & \\
 \swarrow \text{Player 5} & & & &
 \end{array} \right] = M$$

We want to find a ranking vector \vec{r} , where all of the entries of \vec{r} are nonnegative. Then the highest number will be our top player, the next highest number will be the second best player, etc. So, how do we use the matrix M to find a ranking vector \vec{r} ?

To start, we will develop a system of equations by using the fact that we want r_i to be proportional to the sum of the rankings of those players that player i defeated (r_1 is done for you):

$$\begin{aligned}
 r_1 &= \alpha(r_2 + r_4 + r_5) \\
 r_2 &= \\
 r_3 &= \\
 r_4 &= \\
 r_5 &=
 \end{aligned}$$

Write this system as a matrix equation (involve the matrix of wins and losses, M , from above):

So, you got $\alpha M\vec{r} = \vec{r}$. Rewriting this gives:

$$M\vec{r} = \frac{1}{\alpha}\vec{r}$$

↑
 λ

Since $\frac{1}{\alpha}$ is just a constant, we can replace it with λ . Then, we see that \vec{r} is just an eigenvector of the matrix M ! But how do we know there is an eigenvector \vec{r} with all nonnegative entries?

Perron–Fröbenius Theorem:

Let A be an $n \times n$, regular matrix whose entries are nonnegative. Then A has an eigenvalue λ that is real and positive such that there is an associated eigenvector with nonnegative entries.

What does the word *regular* mean? It means there is a power k where the entries of A^k are all positive. In this type of sports' team problem, the matrix A is regular as long as every team has played every other team. What does this theorem tell us? The Perron–Fröbenius Theorem guarantees to us that there is an eigenvector that has nonnegative entries (i.e. it guarantees to us that there is a ranking vector)!!! So, let's use the Perron–Fröbenius Theorem to rank our five tennis players.

First, find the eigenvalues for the matrix, M , and record them here:

Compute the eigenvectors for the eigenvalues that are positive, real numbers. Record them here:

There should be one eigenvector with all positive entries. It will give you the ranking of your players! Record that eigenvector here, and remember that the entries correspond to players (as illustrated below). You now have your ranking ... the largest entry is the top-ranked player, the next largest entry is the second-ranked player, etc. Write your ranking for the tennis players to the right of the ranking vector.

Player 1 →

Player 2 →

Player 3 →

Player 4 →

Player 5 →

$$\left[\begin{array}{c} \\ \\ \\ \\ \end{array} \right]$$

Now let's look at a similar situation where there are three basketball teams, namely the Aardvarks, Buffaloes, and Cheetahs. In this setup, the teams are playing full seasons that include multiple games against each other, and we want to rank the teams relative to one another based on their games within the group. Here are the group head-to-head results:

The Aardvarks beat the Buffaloes twice and beat the Cheetahs once

The Buffaloes beat the Cheetahs once

The Cheetahs beat the Aardvarks once and beat the Buffaloes once

What would be a win-loss matrix M for this situation?

How do you know that this matrix M will have a positive eigenvalue and an eigenvalue with positive entries?

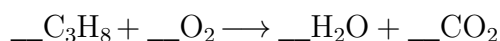
Find all the eigenvalues and the eigenvector with all positive entries.

What does this mean about the comparative rankings of the three teams?

BALANCING CHEMICAL EQUATIONS

In chemistry class, you learn that matter is neither created nor destroyed, but converted from one form to another. This means that the number of each type of atom present before and after a chemical reaction must be the same. Balancing a chemical equation means finding the smallest positive integer of the molecules (**combinations of atoms**) such that the number of atoms of each type before and after the reaction is the same. Let's see how we can use linear algebra to help us!

Consider the following reaction. We need to find the number of each molecule that will balance the chemical equation:



Set up a vector representing each atom used (C, H, and O), and turn the chemical equation into:

$$\begin{bmatrix} C \\ H \\ O \end{bmatrix} : \quad x_1 \begin{bmatrix} 3 \\ 8 \\ 0 \end{bmatrix} + x_2 \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} = x_3 \begin{bmatrix} 0 \\ 2 \\ 1 \end{bmatrix} + x_4 \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}$$

Move everything to one side to get:

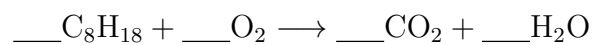
$$x_1 \begin{bmatrix} 3 \\ 8 \\ 0 \end{bmatrix} + x_2 \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} - x_3 \begin{bmatrix} 0 \\ 2 \\ 1 \end{bmatrix} - x_4 \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

solve this vector equation:

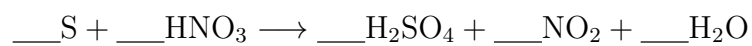
Now, we want the smallest positive integers for x_1 , x_2 , x_3 , and x_4 . How do we find those?

Use linear algebra to balance each of the following chemical equations.

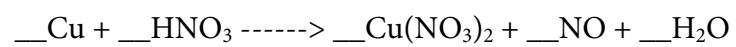
- (1) Octane reacts with oxygen to produce carbon dioxide and water:



- (2) Sulfur reacts with nitric acid to produce sulfuric acid, nitrogen dioxide, and water:



(3) Copper reacts with nitric acid to produce copper nitrate, nitrogen dioxide, and water:



PLTL Calc 1 Session 7 - Subspaces

In this activity we are going to look at the ideas of column space, null space, kernel and range, as well as working with the ideas of a basis and row reduction. The activity is meant to stimulate thought and generate a deeper understanding of the concepts.

1. For an $m \times n$ matrix A , define the column space of A which we will denote as $\text{Col } A$ and define the null space of A which we will denote as $\text{Nul } A$.
2. Consider the matrix

$$C = \begin{bmatrix} 8 & -2 & -9 \\ 6 & 4 & 8 \\ 4 & 0 & 4 \end{bmatrix}$$

Determine if $\vec{w} = [2, 1, -2]^T$ is in $\text{Col } A$, and determine if \vec{w} is in $\text{Nul } A$.

3. Now consider the matrix

$$A = \begin{bmatrix} 2 & 4 & -2 & 1 \\ -2 & -5 & 7 & 3 \\ 3 & 7 & -8 & 6 \end{bmatrix}$$

- (a) If $\text{Col } A$ is a subspace of \mathbb{R}^k , what is k ?
- (b) Find a nonzero vector in $\text{Col } A$.
- (c) Is $\text{Col } A$ all of \mathbb{R}^3 ?
- (d) Find a basis for $\text{Col } A$. Is this also a basis for \mathbb{R}^3 ? Why or why not?
- (e) If $\text{Nul } A$ is a subspace of \mathbb{R}^k , what is k ?
- (f) Find a nonzero vector in $\text{Nul } A$.
- (g) Is $\text{Nul } A$ all of \mathbb{R}^4 ?
- (h) Find a basis for $\text{Nul } A$.

Relation to Linear Transformations

Let's revisit the definition of linear transformations, this time tailored to vector spaces.

Definition - linear transformation

Let V and W be vector spaces. A linear transformation $T : V \rightarrow W$ is a rule that assigns to each vector \vec{x} in V a unique vector $T\vec{x}$ in W with these properties: for all \vec{u}, \vec{v} in V and all scalars c :

$$(i) \quad T(\vec{u} + \vec{v}) = T\vec{u} + T\vec{v}$$

$$(ii) \quad T(c\vec{u}) = cT\vec{u}$$

We also use the term **kernel of T** to refer to the null space of the linear transformation matrix associated to T . So, the kernel of T is the set of all \vec{u} in V such that $T\vec{u} = \vec{0}$; here, $\vec{0}$ is in W .

The **range of T** is the set of all vectors in W of the form $T\vec{x}$ for some \vec{x} in V . This is equivalent to the column space of the linear transformation matrix associated to T .

Theorem

For vector spaces V and W and a linear transformation $T : V \rightarrow W$, the kernel of T is a subspace of V and the range of T is a subspace of W .

1. Consider \mathbb{P}_2 , the vector space of all polynomials of degree at most 2. Define the linear transformation $T : \mathbb{P}_2 \rightarrow \mathbb{R}^2$ by $T(\vec{p}) = \begin{bmatrix} \vec{p}(0) \\ \vec{p}(1) \end{bmatrix}$.

For example, if $\vec{p}(t) = 3 + 5t + 7t^2$, $T(\vec{p}) = \begin{bmatrix} 3 \\ 15 \end{bmatrix}$.

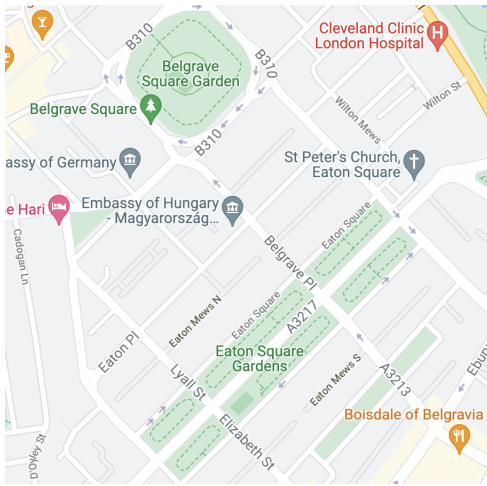
- (a) Find a set of polynomials in \mathbb{P}_2 that spans the kernel of T .
- (b) Describe the range of T .

2. Define the linear transformation $L : \mathbb{P}_2 \rightarrow \mathbb{R}^2$ by $L(\vec{p}) = \begin{bmatrix} \vec{p}(0) \\ \vec{p}'(0) \end{bmatrix}$.

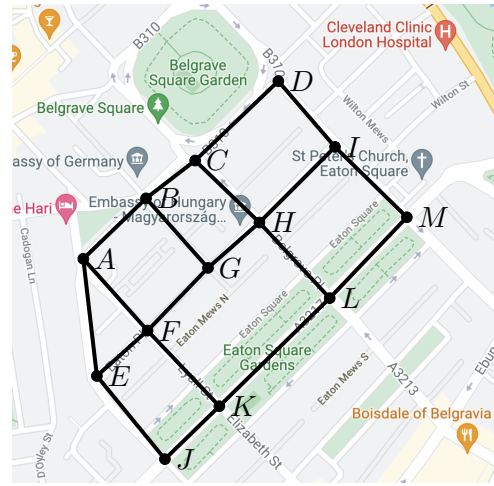
- (a) Find a set of polynomials in \mathbb{P}_2 that spans the kernel of T .
- (b) Describe the range of T .

FINDING ALL PATHS IN A GIVEN NETWORK

Mathematics is often used to find efficient ways of transporting goods from one place to another. In mathematics, we can represent intersections of roads by dots (called vertices) and the sections of roads between the intersections by line segments (called edges). Let's consider the following street network from London, England:



Street network



Graph associated to central streets

A *path of length n* in a graph going from vertex i to vertex j is a sequence of n edges in the graph, starting at vertex i and ending at vertex j (think of them as a set of directions for someone to walk from the intersection labeled i to the intersection labeled j !). Here is one path of length 2 from vertex A to vertex G , find one more:

$$A \rightarrow F \rightarrow G \quad \text{and} \quad \underline{\hspace{2cm}}$$

Here is one path of length 3 from vertex C to vertex M , find one more:

$$C \rightarrow H \rightarrow I \rightarrow M \quad \text{and} \quad \underline{\hspace{2cm}}$$

Find a path of length 6 from vertex F to vertex M :

Find a path of length 7 from vertex E to vertex D :

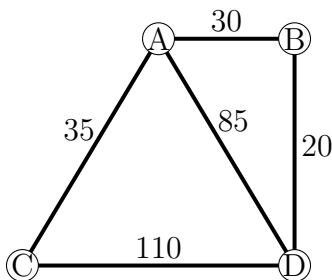
From the graph, we can form the *adjacency* matrix, which has a 1 in entry (i, j) if vertex i is connected to vertex j and a 0 otherwise. Complete the adjacency matrix for the graph above:

$$\begin{array}{c}
 A \\
 B \\
 C \\
 D \\
 E \\
 F \\
 G \\
 H \\
 I \\
 J \\
 K \\
 L \\
 M
 \end{array}
 \begin{bmatrix}
 A & B & C & D & E & F & G & H & I & J & K & L & M \\
 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 1 & & & & & & & & & & & & \\
 0 & & & & & & & & & & & & \\
 0 & & & & & & & & & & & & \\
 1 & & & & & & & & & & & & \\
 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
 0 & & & & & & & & & & & & \\
 0 & & & & & & & & & & & & \\
 0 & & & & & & & & & & & & \\
 0 & & & & & & & & & & & & \\
 0 & & & & & & & & & & & & \\
 0 & & & & & & & & & & & & \\
 0 & & & & & & & & & & & &
 \end{bmatrix}
 := P$$

We will refer to this matrix as matrix P . Using a calculator or software program, compute P^2 . What do the entries of the matrix tell you? Now, compute P^3 and determine what the entries of the matrix signify. Let's generalize:

For $n \in \{1, 2, 3, \dots\}$, the (i, j) -th entry of P^n gives:

What if we wanted to also record the distances of the roads? Consider the (smaller) graph:



In this scenario, we are indicating that the distance between intersection C and intersection D is 110 miles. That means that following the path $C \rightarrow A \rightarrow B \rightarrow D$ is less miles (85 miles) than traveling directly on the edge $C \rightarrow D$ (110 miles)!

We can use a modification of matrix multiplication, where we change up the addition and multiplication to find the length of the shortest path between any two given vertices! First, let's form the adjacency matrix:

$$\begin{array}{c} \text{A} \quad \text{B} \quad \text{C} \quad \text{D} \\ \text{A} \begin{bmatrix} 0 & 30 & 35 & 85 \end{bmatrix} \\ \text{B} \begin{bmatrix} 30 & 0 & 0 & 20 \end{bmatrix} \\ \text{C} \begin{bmatrix} 35 & 0 & 0 & 110 \end{bmatrix} \\ \text{D} \begin{bmatrix} 85 & 20 & 110 & 0 \end{bmatrix} \end{array}$$

But, let's make one modification. Since our matrix now encodes distances, it makes sense to have a 0 in the (A, A) entry, as there is a distance of 0 miles between vertex A and itself. However, it doesn't make sense to have a 0 in the (B, C) entry, since you can't get there directly (i.e., by traveling on one edge). So, in this latter case, we will replace those zeros by ∞ . Thus, our adjacency matrix becomes:

$$\begin{array}{c} \text{A} \quad \text{B} \quad \text{C} \quad \text{D} \\ \text{A} \begin{bmatrix} 0 & 30 & 35 & 85 \end{bmatrix} \\ \text{B} \begin{bmatrix} 30 & 0 & \infty & 20 \end{bmatrix} \\ \text{C} \begin{bmatrix} 35 & \infty & 0 & 110 \end{bmatrix} \\ \text{D} \begin{bmatrix} 85 & 20 & 110 & 0 \end{bmatrix} \end{array} := P$$

So, how are we going to modify the matrix multiplication? Instead of addition, we will take the minimum; and instead of multiplication, we will add. Let's look closer at this:

$$P^2 = \begin{bmatrix} 0 & 30 & 35 & 85 \\ 30 & 0 & \infty & 20 \\ 35 & \infty & 0 & 110 \\ 85 & 20 & 110 & 0 \end{bmatrix} \begin{bmatrix} 0 & 30 & 35 & 85 \\ 30 & 0 & \infty & 20 \\ 35 & \infty & 0 & 110 \\ 85 & 20 & 110 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 30 & 35 & 50 \\ & & & \\ & & & \\ & & & \end{bmatrix}$$

Here are the computations of the first row of P^2 :

$$(1,1)\text{-entry} \quad [0 \ 30 \ 35 \ 85] [0 \ 30 \ 35 \ 85]^T = \min\{0 + 0, 30 + 30, 35 + 35, 85 + 85\} = 0$$

$$(1,2)\text{-entry} \quad [0 \ 30 \ 35 \ 85] [30 \ 0 \ \infty \ 20]^T = \min\{0 + 30, 30 + 0, 35 + \infty, 85 + 20\} = 30$$

$$(1,3)\text{-entry} \quad [0 \ 30 \ 35 \ 85] [35 \ \infty \ 0 \ 110]^T = \min\{0 + 35, 30 + \infty, 35 + 0, 85 + 110\} = 35$$

$$(1,4)\text{-entry} \quad [0 \ 30 \ 35 \ 85] [85 \ 20 \ 110 \ 0]^T = \min\{0 + 85, 30 + 20, 35 + 110, 85 + 0\} = 50$$

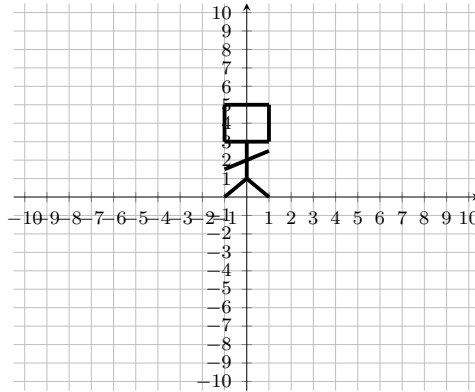
Looking at this first row, we see that the $(1, 4)$ -entry of matrix P recorded that the distance from vertex A to vertex D was 85 miles. Now, when we look at P^2 , we see that the $(1, 4)$ -entry is now 50. Looking at the graph, we can see that it is shorter to go along the path $A \rightarrow B \rightarrow D$ (traveling on 2 edges) for a total length of 50 miles. So, P^2 looks for the shortest paths between any two vertices, using up to 2 edges. If we keep going and compute P^3 , P^4 , etc. we can find the shortest route between all pairs of vertices in the graph! Note, you can stop computing when $P^n = P^{n+1}$. Finish out computing P^2 , and then continue with P^3 , P^4 , etc. until you have found the shortest distance between all pairs of vertices in the graph. Record them here:

	Distance
$A \rightarrow B$	
$A \rightarrow C$	
$A \rightarrow D$	
$B \rightarrow C$	
$B \rightarrow D$	
$C \rightarrow D$	

It should be noted that we don't need to record both directions (for example, both $A \rightarrow B$ and $B \rightarrow A$) as the length of the shortest paths will be the same. For those who are more familiar with graph theory, this process also works for directed graphs (these are graphs where the distance from $A \rightarrow B$ and $B \rightarrow A$ might be different).

COMPUTER ANIMATION

Let's look at the idea of animating an object using linear transformations. Let me introduce my good friend Blocko. He is a highly rated actor and has starred in my award-winning films.



Notice that Blocko is defined by the coordinate pairs that make up his shape, namely

$$\{(-1, 0), (0, 1), (1, 0), (0, 3), (-1, 1.5), (1, 2.5), (-1, 3), (-1, 5), (1, 5), (1, 3)\}.$$

We can think of any movement of Blocko as:

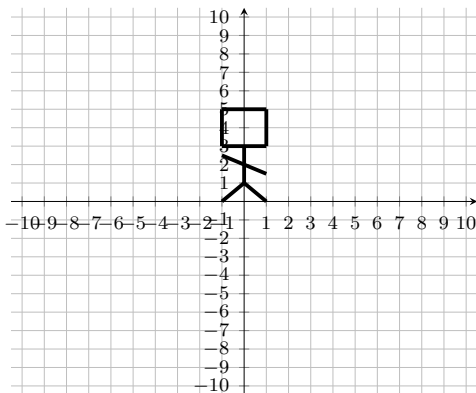
$$(1) \quad \begin{aligned} x_{\text{new}} &= ax_{\text{old}} + by_{\text{old}} + c \\ y_{\text{new}} &= dx_{\text{old}} + ey_{\text{old}} + f \end{aligned}$$

where $(x_{\text{old}}, y_{\text{old}})$ represents the starting coordinates of the movement and $(x_{\text{new}}, y_{\text{new}})$ represents the ending coordinates of the movement (also, $a, b, c, d, e, f \in \mathbb{R}$). We want to think of this as a linear transformation $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$. To do this, write System (1) in the form:

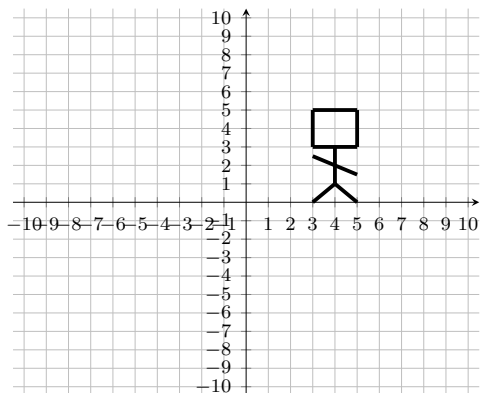
$$M\vec{v} = \begin{bmatrix} x_{\text{new}} \\ y_{\text{new}} \end{bmatrix},$$

where M is a matrix (hint: you might need a 1 in the vector \vec{v}):

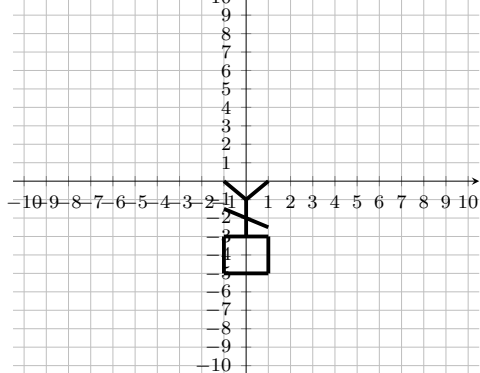
Determine the linear transformation $T(\vec{v}) = M\vec{v}$ that would make Blocko end in this position:



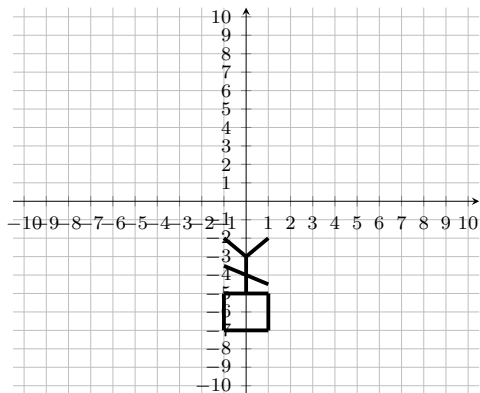
Determine the linear transformation $T(\vec{v}) = M\vec{v}$ that would make Blocko end in this position:



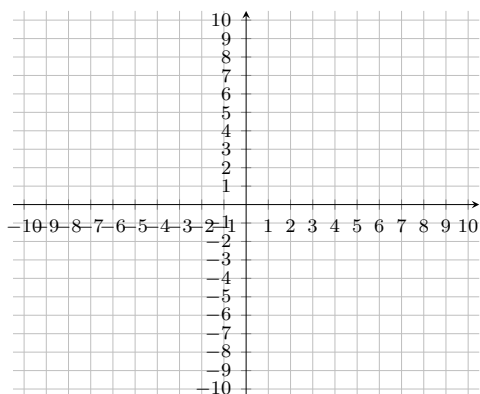
Determine the linear transformation $T(\vec{v}) = M\vec{v}$ that would make Blocko end in this position:



Determine the linear transformation $T(\vec{v}) = M\vec{v}$ that would make Blocko end in this position:



Determine the linear transformation $T(\vec{v}) = M\vec{v}$ that would rotate Blocko by 45° counterclockwise. Use that matrix and computation to draw a picture of Blocko.



FIBONACCI NUMBERS AND BINET'S FORMULA

The Fibonacci numbers are a sequence of numbers F_1, F_2, F_3, \dots where $F_1 = 1, F_2 = 1$, and $F_n = F_{n-1} + F_{n-2}$. This means that each Fibonacci number is the sum of the previous two. Write out the first 15 Fibonacci numbers:

The formula $F_n = F_{n-1} + F_{n-2}$ is a recursive formula, but we would like a *closed* formula, where we just plug in n and it gives us F_n (the n^{th} Fibonacci number). Let's see how matrix diagonalization can help us. Consider the matrix:

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

Calculate the following:

$$A^2 = \begin{bmatrix} & \\ & \end{bmatrix}, \quad A^3 = \begin{bmatrix} & \\ & \end{bmatrix}, \quad A^4 = \begin{bmatrix} & \\ & \end{bmatrix}, \quad A^5 = \begin{bmatrix} & \\ & \end{bmatrix}$$

What pattern do you notice?

$$(1) \quad A^n = \begin{bmatrix} & \\ & \end{bmatrix}$$

Why is this true?

Let's look at another way to compute A^n , using the diagonalization of A , $A = PDP^{-1}$. Then

$$A^n = \underbrace{(PDP^{-1})(PDP^{-1}) \cdots (PDP^{-1})}_{n \text{ times}} = PD^nP^{-1}$$

Diagonalize A and write down your diagonalization here:

$$A = \begin{bmatrix} & \\ & \end{bmatrix} \begin{bmatrix} & \\ & \end{bmatrix} \begin{bmatrix} & \\ & \end{bmatrix}$$

Now, use your diagonalization to compute A^n . To make things easier, set $\rho = \frac{1 + \sqrt{5}}{2}$ and $\bar{\rho} = \frac{1 - \sqrt{5}}{2}$. Write your result here:

$$(2) \quad A^n = \begin{bmatrix} & \\ & \end{bmatrix}$$

Setting the (2,1)-entry of the matrices in Equations 1 and 2 equal gives you Binet's Formula!!!

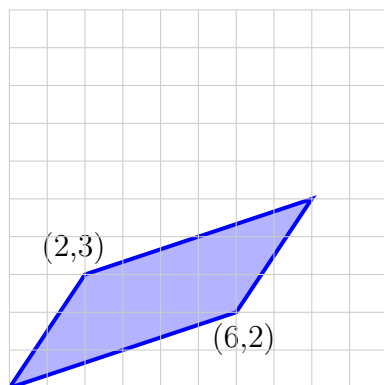
Binet's Formula: Let $n = 1, 2, 3, \dots$, then

$$F_n =$$

Use Binet's Formula to find the 25^{th} Fibonacci number:

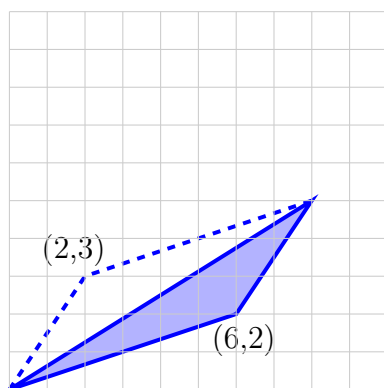
DETERMINING MORE GENERALIZED AREAS

We know that determinants can be used to find the area of a parallelogram. Here is an example to review this process:



$$\text{Area} = \left| \det \begin{bmatrix} 2 & 6 \\ 3 & 2 \end{bmatrix} \right| = |4 - 18| = 14$$

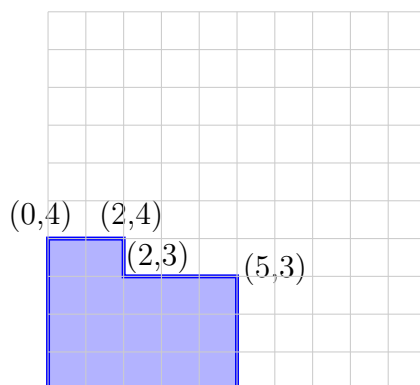
We can also find the area of a triangle by noticing it is half of a parallelogram. For example:



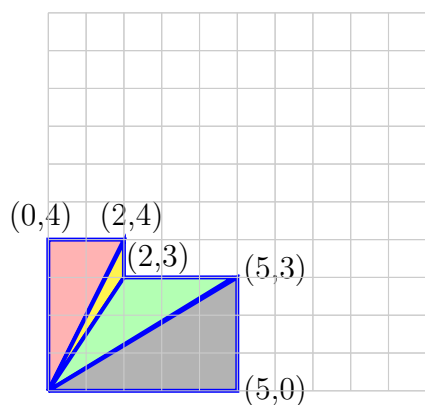
$$\text{Area} = \frac{1}{2} \left| \det \begin{bmatrix} 2 & 6 \\ 3 & 2 \end{bmatrix} \right| = \frac{1}{2} |4 - 18| = \frac{1}{2} (14) = 7$$

Notice that to use determinants to calculate area, one of the corners of the parallelogram (or triangle) must be located at the origin.

How can we use this idea to find the area of the following shape?

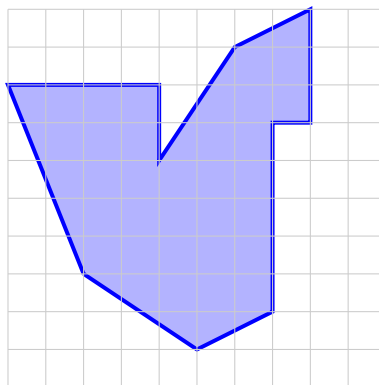
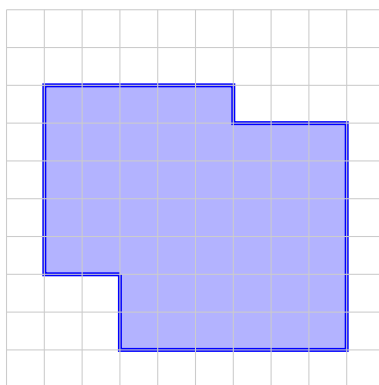


Let's think about cutting the shape up into triangles. Then we can find the area by summing up the areas of the rectangles!



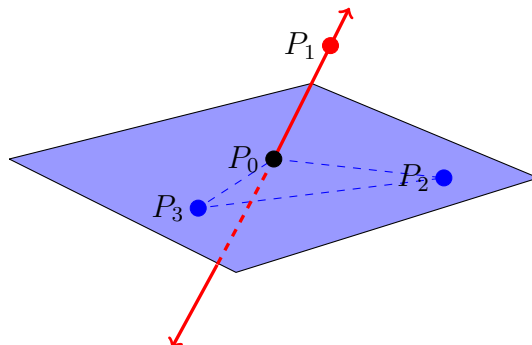
$$\begin{aligned}\text{Area} &= \frac{1}{2} \left(\begin{vmatrix} 0 & 2 \\ 4 & 4 \end{vmatrix} + \begin{vmatrix} 2 & 2 \\ 4 & 3 \end{vmatrix} + \begin{vmatrix} 2 & 5 \\ 3 & 3 \end{vmatrix} + \begin{vmatrix} 5 & 5 \\ 3 & 0 \end{vmatrix} \right) \\ &= \frac{1}{2} (8 + 2 + 9 + 15) = \frac{1}{2} (34) = 17\end{aligned}$$

Use determinants to find the area of the following shape:



FINDING EQUATIONS OF LINES AND PLANES WITH DETERMINANTS

When collecting data, we often want a curve (or function) that best fits the data. This means we want to be able to come up with a mathematical function that could be used to closely estimate the collected data, as well as to be able to predict other data points.



How can we use determinants to find descriptions of where all these points lie? In the picture above, how can we use derivatives to find the equation of the red line given the points P_0 and P_1 ? Or, how could we find the equation of the blue plane given the points P_0 , P_2 , and P_3 ? We will focus on finding equations of lines and planes that best fit a collection of points given in 2-dimensional and 3-dimensional space, respectively; however, you could generalize this process to higher dimensional spaces! Let's start thinking about using a determinant to find the line going through two points. Consider the points (x_1, y_1) and (x_2, y_2) . The line passing through them has the form:

$$ax + by + c = 0.$$

Since (x_1, y_1) and (x_2, y_2) lie on that line, we get two more equations through substitution:

$$ax_1 + by_1 + c = 0$$

← Fill in the second equation

If we transform these three equations into a matrix equation, we get:

$$\begin{bmatrix} x & y & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

\uparrow
 M

If the columns of a matrix are linearly dependent, then the determinant of the matrix is equal to zero. Since our points lie on the same line, the columns of M are linearly dependent. Then calculating the determinant of M will provide the equation of the line through the two points (x_1, y_1) and (x_2, y_2) . Write that equation down here:

This idea can be expanded to find equations of planes (and more general spaces) passing through a collection of points. The general equation of a plane is:

$$ax + by + cz + d = 0$$

Generalize the process from above with lines to find the equation of the plane passing through the points $(2, 3, 1)$, $(1, 0, 4)$, and $(5, 1, -3)$.

MODELING PREDATOR-PREY DYNAMICS



Deep in the redwood forests of California, dusky-footed wood rats provide up to 80% of the diet for the spotted owl, the main predator of the wood rat. Let O_k denote the number of owls and R_k denote the number of rats (in thousands) at time k (in months) in the region being studied. Suppose that:

$$\begin{aligned} O_{k+1} &= (.5)O_k + (.4)R_k \\ R_{k+1} &= (-p)O_k + (1.1)R_k \end{aligned}$$

where p is some positive parameter.

- What does the first equation say about the owl population and how it changes (or what affects its change)?
- What does the second equation say about the rat population and how it changes (or what affects its change)?

We want to determine the evolution of this system when the predation parameter p is 0.104 (i.e. we want to determine the long term effects on the owl and rat populations). Our system is:

$$\begin{aligned} O_{k+1} &= (0.5)O_k + (0.4)R_k \\ R_{k+1} &= (-0.104)O_k + (1.1)R_k \end{aligned}$$

Determine the two eigenvalues (rounded to two decimal places each) of the coefficient matrix of the above system, call them λ_1 and λ_2 (where $\lambda_1 < \lambda_2$); along with a corresponding eigenvector for each, call them \vec{v}_1 and \vec{v}_2 .

$$\lambda_1 = \quad \quad \vec{v}_1 =$$

$$\lambda_2 = \quad \quad \vec{v}_2 =$$

Thinking of the system as $\vec{x}_{k+1} = A\vec{x}_k$ where $\vec{x}_k = \begin{bmatrix} O_k \\ R_k \end{bmatrix}$, an initial \vec{x}_0 can be written as

$$\vec{x}_0 = c_1\vec{v}_1 + c_2\vec{v}_2.$$

This works because \vec{v}_1 and \vec{v}_2 are linearly independent and form a basis for \mathbb{R}^2 . Now, use the formula $\vec{x}_{k+1} = A\vec{x}_k$ to verify that

$$\vec{x}_k = c_1 \lambda_1^k \vec{v}_1 + c_2 \lambda_2^k \vec{v}_2.$$

We are interested in what happens in the long term, so as $k \rightarrow \infty$. Compute:

$$\lim_{k \rightarrow \infty} \vec{x}_k =$$

Explain why, if we are concerned with the long term, we can simplify $\vec{x}_k = c_1 \lambda_1^k \vec{v}_1 + c_2 \lambda_2^k \vec{v}_2$ to

$$\vec{x}_k = c_2 \lambda_2^k \vec{v}_2.$$

Furthermore, as $k \rightarrow \infty$:

$$\begin{aligned} \vec{x}_{k+1} &\approx c_2 \lambda_2^{k+1} \vec{v}_2 \\ &= \lambda_2 (c_2 \lambda_2^k \vec{v}_2) \\ &\approx \lambda_2 \vec{x}_k \end{aligned}$$

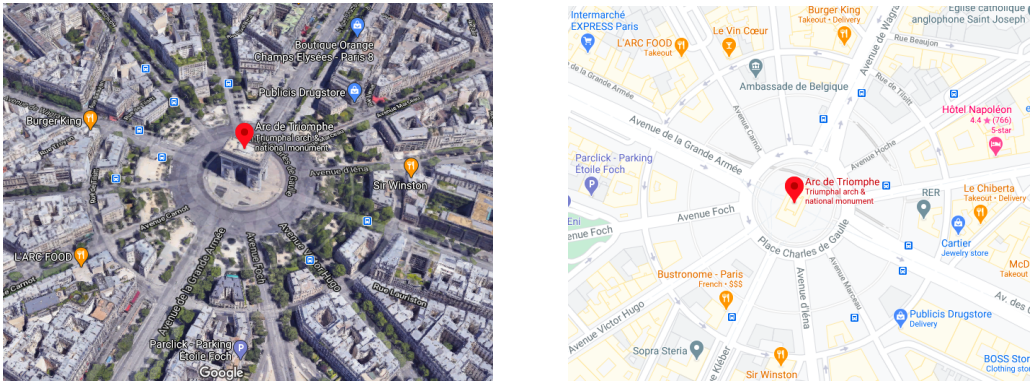
Fill in the value of λ_2 :

$$\vec{x}_{k+1} = \underline{\hspace{1cm}} \vec{x}_k.$$

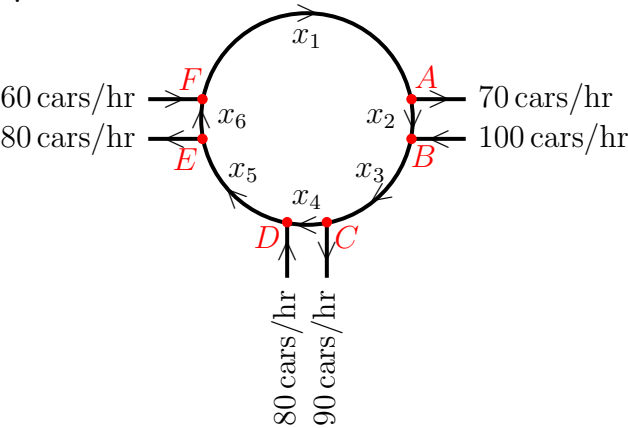
What does this predict about the owl and rat populations in the future?

ANALYZING TRAFFIC FLOW

Linear algebra can be used to study traffic flow on local streets and intersections. Imagine you are in Paris, tackling the traffic circle around L'Arc de Triomphe as picture below (from Google Maps):



Looking at the intricacies of the street network for L'Arc de Triomphe is daunting, so let's look at a simplified traffic circle. Consider the network of streets below, where the arrows denote the flow of traffic (indicating one-way streets). The variables x_1, x_2, \dots, x_6 indicate the unknown traffic flows on the corresponding sections of roads in cars per hour. The red dots indicate the intersections and are denoted by A, B, \dots, F .



To set up a system of equations, we use the idea that at each intersection, traffic going into the network equals traffic going out of the network (no cars mysteriously disappear into black holes). Also, the total traffic going into the network, here $60 + 100 + 80 = 240$ cars/hr, equals the traffic going out of the network, here $80 + 70 + 90$ cars/hr. The work for intersection A is done below. Complete the work for intersections B, C, \dots, F and solve the system. What does your solution say about maximum and minimum levels of traffic on each section of road?

	Traffic In	=	Traffic Out
A	x_1	=	$x_2 + 70$
B			
C			
D			
E			
F			