

A CHILD'S GARDEN OF GROUPS

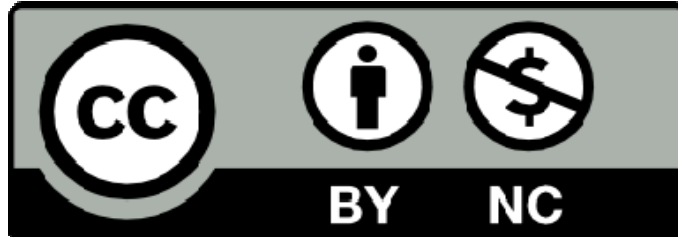
Matrix Groups

(Part 7)



by

Doc Benton



Creative Commons License

You are free to:

- Share this work
- Adapt this work
- Attribute all original materials to Doc Benton

You are not free to:

- Charge money for this work; Knowledge is free!

CONTENTS (PART 7)

Introduction (Part 7)	1
What is a <i>Matrix</i> ?	2
<i>Permutation Matrices</i>	17
Other <i>Matrix Groups</i>	22
How to Use GAP (Part 7)	34
Summary (Part 7)	53
Practice (Part 7)	54
Practice (Part 7) – Answers	56

INTRODUCTION (PART 7)

In Part 7 we introduce you to *matrices*, and we show you how every *group* can be expressed as a *matrix group*. We also show you how to use GAP to work with *matrices*, and we introduce you to some important *matrix groups* and their applications to both computer graphics and physics. Part 7 will change your life!

WHAT IS A MATRIX?

Matrices have a lot to do with *group theory*, and in mathematics, a *matrix* is usually defined as a rectangular array of numbers. However, as is the case with many things in mathematics, you really need to see an example in order to understand the definition. Thus, for instance, the following example shows how *matrices* are generally written.

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

We typically enclose the elements of a *matrix* in either parentheses or brackets, depending upon one's preference, and it is common to assign a capital letter to represent a *matrix* so that we don't have to write the whole thing over again. Also, the dimensions or order of the *matrix* refer to the number of rows and columns that a *matrix* has. Thus, when we write "2 by 3" or "2×3", we are trying to say that our *matrix* has 2 rows and 3 columns. Hence, the *matrix A* above is a 2×3 matrix. Furthermore, if the number of rows of a *matrix* is the same as the number of columns, then we call it a *square matrix*. Additionally, a *matrix* with only one row is called a *row matrix*, and a *matrix* with only one column is called a *column matrix*.

Square matrix: $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$

Row matrix: $[1 \ 2 \ 3]$

Column matrix: $\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$

Now let's look at a couple of 2×2 *square matrices*, $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ and $B = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$. In mathematics, whenever we discover a new object, we immediately begin to wonder if we can do arithmetic with that object. Thus, think about how we might add the above two *matrices* together. In other words, what would we like to be the sum of A and B ? If you think about it, what seems to be most natural is to simply add the corresponding entries of the two *matrices* together. In other words,

$$A + B = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} + \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 1+5 & 2+6 \\ 3+7 & 4+8 \end{bmatrix} = \begin{bmatrix} 6 & 8 \\ 10 & 12 \end{bmatrix}$$

And this is exactly how we do define addition of *matrices*, in terms of simply adding the corresponding elements of the *matrices*. However, this also implies that our matrices must have the same order or dimensions so that we do indeed have elements that correspond. Thus, for instance, the sum of a 3×1 *matrix* with a 1×3 *matrix* is not defined.

$$\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} + [4 \ 5 \ 6] = \text{does not exist}$$

However, we can easily add two 3×1 *matrices* together.

$$\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} + \begin{bmatrix} 3 \\ 4 \\ 5 \end{bmatrix} = \begin{bmatrix} 1+3 \\ 2+4 \\ 3+5 \end{bmatrix} = \begin{bmatrix} 4 \\ 6 \\ 8 \end{bmatrix}$$

We can also subtract one *matrix* from another in the same manner just so long as the two matrices have identical dimensions (same number of rows and same number of columns).

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

$$\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} - \begin{bmatrix} 3 \\ 4 \\ 5 \end{bmatrix} = \begin{bmatrix} 1-3 \\ 2-4 \\ 3-5 \end{bmatrix} = \begin{bmatrix} -2 \\ -2 \\ -2 \end{bmatrix}$$

Now let's think about how one might multiply *matrices*. If we followed the pattern we saw for adding and subtracting *matrices*, then we would guess that to multiply *matrices* together that you would just have to multiply corresponding entries as follows.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 1 \cdot 5 & 2 \cdot 6 \\ 3 \cdot 7 & 4 \cdot 8 \end{bmatrix} = \begin{bmatrix} 5 & 12 \\ 21 & 32 \end{bmatrix}$$

Yes, multiplying corresponding entries is neat and tidy, and it seems like a common sense way to do it, but, for better or worse, that's not how it's done. Instead, we multiply *matrices* in a way that is not at all intuitive and that amounts to, at each step, multiplying a *row matrix* by a *column matrix* as shown in the examples below.

$$\begin{bmatrix} 1 & 2 \end{bmatrix} \cdot \begin{bmatrix} 3 \\ 4 \end{bmatrix} = [1 \cdot 3 + 2 \cdot 4] = [3 + 8] = [11]$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 5 \\ 7 \end{bmatrix} = \begin{bmatrix} 1 \cdot 5 + 2 \cdot 7 \\ 3 \cdot 5 + 4 \cdot 7 \end{bmatrix} = \begin{bmatrix} 5 + 14 \\ 15 + 28 \end{bmatrix} = \begin{bmatrix} 19 \\ 43 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 1 \cdot 5 + 2 \cdot 7 & 1 \cdot 6 + 2 \cdot 8 \\ 3 \cdot 5 + 4 \cdot 7 & 3 \cdot 6 + 4 \cdot 8 \end{bmatrix} = \begin{bmatrix} 5 + 14 & 6 + 16 \\ 15 + 28 & 18 + 32 \end{bmatrix} = \begin{bmatrix} 19 & 22 \\ 43 & 50 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \cdot \begin{bmatrix} 7 \\ 8 \\ 9 \end{bmatrix} = \begin{bmatrix} 1 \cdot 7 + 2 \cdot 8 + 3 \cdot 9 \\ 4 \cdot 7 + 5 \cdot 8 + 6 \cdot 9 \end{bmatrix} = \begin{bmatrix} 7 + 16 + 27 \\ 28 + 40 + 54 \end{bmatrix} = \begin{bmatrix} 50 \\ 122 \end{bmatrix}$$

One of the things that you hopefully notice from the examples above is that we can only do *matrix* multiplication when the number of columns in the *matrix* on the left is the same as the number of rows in the *matrix* on the right. Furthermore, the order or dimensions of the product is equal to (number of rows in the left *matrix*) \times (number of columns in the right *matrix*). For example, we can multiply a 2×3 *matrix* on the left by a 3×1 *matrix* on the right since our *matrix* on the left has 3 columns and our *matrix* on the right has 3 rows. Additionally, the outer two numbers, 2 and 1, tell us that the dimensions of the product will be 2×1 .

So this answers the question of how you do multiplication of *matrices*, but we still haven't answered the question of why you do it this way. The explanation of this goes back to the origin of *matrices* as a tool for solving systems of equations. Thus, let's consider the following example of two equations in two unknowns.

$$\begin{aligned} 2x + 3y &= 8 \\ 5x + 7y &= 19 \end{aligned}$$

One can imagine that if you are spending all day studying systems of equations like this one, then after awhile you might start looking for shortcuts. And that just might be what Arthur Cayley, one of the developers of *matrix theory*, actually did. In any event, one streamlined way to rewrite the above system of equations is by putting coefficients, variables, and results each in their own *matrix*, and when we do so, we get the following *matrix equation*.

$$\begin{bmatrix} 2 & 3 \\ 5 & 7 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 8 \\ 19 \end{bmatrix}$$

And it is this form that defines the way we want to do *matrix* multiplication because if we multiply using our “row times column” method, then we essentially get back our original system of equations.

$$\begin{bmatrix} 2 & 3 \\ 5 & 7 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2x+3y \\ 5x+7y \end{bmatrix} = \begin{bmatrix} 8 \\ 19 \end{bmatrix} \Rightarrow \begin{cases} 2x + 3y = 8 \\ 5x + 7y = 19 \end{cases}$$

Furthermore, if we want to divide one *matrix* by another, then we essentially turn it back into a multiplication problem just like when we say that six divided by two is the same as six times the multiplicative *inverse* of two.

$$6 \div 2 = 6 \cdot \frac{1}{2} = 6 \cdot 2^{-1} = 3$$

When we did multiplication of *matrices*, we saw that there were some restrictions imposed upon the *matrices*. In particular, the number of columns in the one on the left must always equal the number of rows in the *matrix* on the right for multiplication to be defined. Otherwise, multiplication is not possible, and this makes *matrix* multiplication different from multiplication of numbers on the number line where the product of any two arbitrarily picked numbers always exists. In a similar way, there are going to be some restrictions on when we can “divide” one *matrix* by another.

The first restriction we impose is that “division” can only be done with *square matrices*, *matrices* that have the same number of rows as columns. And since we are really going to define our division in terms of multiplication by an *inverse*, such as $A \cdot B^{-1}$, let’s discuss exactly what we mean by the *inverse* of a *square matrix*, when such an *inverse* exists, and how to use the *inverse* of a *square matrix* to solve the system of equations.

When it comes to multiplicative *inverses*, notice that if we multiply 2 by $2^{-1} = \frac{1}{2}$, then the result is $2 \cdot 2^{-1} = 2 \cdot \frac{1}{2} = 1$ and 1 is the *identity element* with regard to multiplication of *real numbers*. Hence, if A is a *square matrix* that has an *inverse* A^{-1} , then we would expect that $A \cdot A^{-1}$ is equal to whatever the *identity* is with respect to *square matrices* of a given dimension. Well, without giving a proof, it is, nonetheless, easy to show you what such a multiplicative *identity* looks like. For example, if we are talking about 2×2 *square matrices*, then the *identity element* is $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, and if we are talking about 3×3 *square matrices*, then the

identity element is $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$. In other words, the *identity element* for

multiplication of $n \times n$ *square matrices* is an $n \times n$ *matrix* that has the number 1 along the diagonal from upper left to lower right and then has 0 everywhere else.

Furthermore, observe the examples below which show $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

indeed acting like *identity elements*.

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 1 \cdot 1 + 0 \cdot 3 & 1 \cdot 2 + 0 \cdot 4 \\ 0 \cdot 1 + 1 \cdot 3 & 0 \cdot 2 + 1 \cdot 4 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 \cdot 1 + 2 \cdot 0 & 1 \cdot 0 + 2 \cdot 1 \\ 3 \cdot 1 + 4 \cdot 0 & 3 \cdot 0 + 4 \cdot 1 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} = \begin{bmatrix} 1 \cdot 1 + 0 \cdot 4 + 0 \cdot 7 & 1 \cdot 2 + 0 \cdot 5 + 0 \cdot 8 & 1 \cdot 3 + 0 \cdot 6 + 0 \cdot 9 \\ 0 \cdot 1 + 1 \cdot 4 + 0 \cdot 7 & 0 \cdot 2 + 1 \cdot 5 + 0 \cdot 8 & 0 \cdot 3 + 1 \cdot 6 + 0 \cdot 9 \\ 0 \cdot 1 + 0 \cdot 4 + 1 \cdot 7 & 0 \cdot 2 + 0 \cdot 5 + 1 \cdot 8 & 0 \cdot 3 + 0 \cdot 6 + 1 \cdot 9 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

Thus, we can say, for instance, that if A is a 2×2 *matrix*, then an *inverse* for A would be another 2×2 *matrix* denoted by A^{-1} such that $AA^{-1} = A^{-1}A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, the *identity element* for multiplication of 2×2 *matrices*. Here is a quick example of how an *inverse matrix* might be used to solve a system of equations.

Original system of equations:

$$\begin{aligned} 2x - 3y &= -4 \\ 5x + 7y &= 19 \end{aligned}$$

Matrix equation:

$$\begin{bmatrix} 2 & -3 \\ 5 & 7 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -4 \\ 19 \end{bmatrix}$$

$$\text{Coefficient matrix} = A = \begin{bmatrix} 2 & -3 \\ 5 & 7 \end{bmatrix}$$

$$\text{Variable matrix} = X = \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\text{Result matrix} = R = \begin{bmatrix} -4 \\ 19 \end{bmatrix}$$

$$\text{Inverse matrix} = A^{-1} = \begin{bmatrix} \frac{7}{29} & \frac{3}{29} \\ \frac{-5}{29} & \frac{2}{29} \end{bmatrix} \quad (\text{Take my word for it!})$$

Algebraic solution:

$$\begin{aligned}A \cdot X &= R \\ \Rightarrow A^{-1}A \cdot X &= A^{-1} \cdot R \\ \Rightarrow I \cdot X &= A^{-1} \cdot R \\ \Rightarrow X &= A^{-1} \cdot R \\ \Rightarrow \begin{bmatrix} x \\ y \end{bmatrix} &= \begin{bmatrix} \frac{7}{29} & \frac{3}{29} \\ -\frac{5}{29} & \frac{2}{29} \end{bmatrix} \cdot \begin{bmatrix} -4 \\ 19 \end{bmatrix} = \begin{bmatrix} \frac{7(-4)}{29} + \frac{3 \cdot 19}{29} \\ -\frac{5(-4)}{29} + \frac{2 \cdot 19}{29} \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \\ \Rightarrow \begin{matrix} x = 1 \\ y = 2 \end{matrix} &\end{aligned}$$

Now let's address how to tell when a system of equations like $\begin{matrix} 2x - 3y = -4 \\ 5x + 7y = 19 \end{matrix}$

even has a solution. Usually what we might do to solve a system like this without using *matrices* is to, perhaps, multiply the top equation on both sides by 7 and then the bottom equation by 3 to obtain the following:

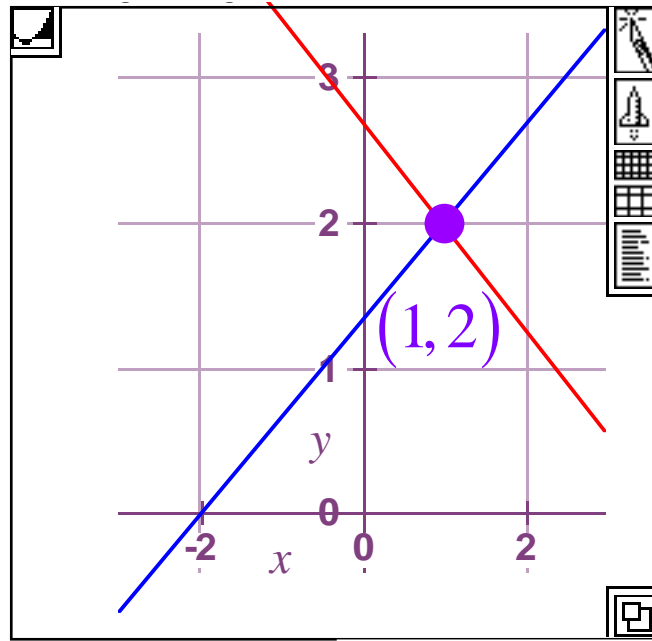
$$\begin{aligned}14x - 21y &= -28 \\ 15x + 21y &= 57\end{aligned}$$

We can now add the two equations together to eliminate the y variable, and that results in an equation that we can easily solve for x . And once we have a value for x , we can easily go back and solve for y .

$$\begin{aligned}14x - 21y &= -28 \\ 15x + 21y &= 57 \\ \hline 29x &= 29\end{aligned}$$

$$\begin{aligned}x &= 1 \\ y &= 2\end{aligned}$$

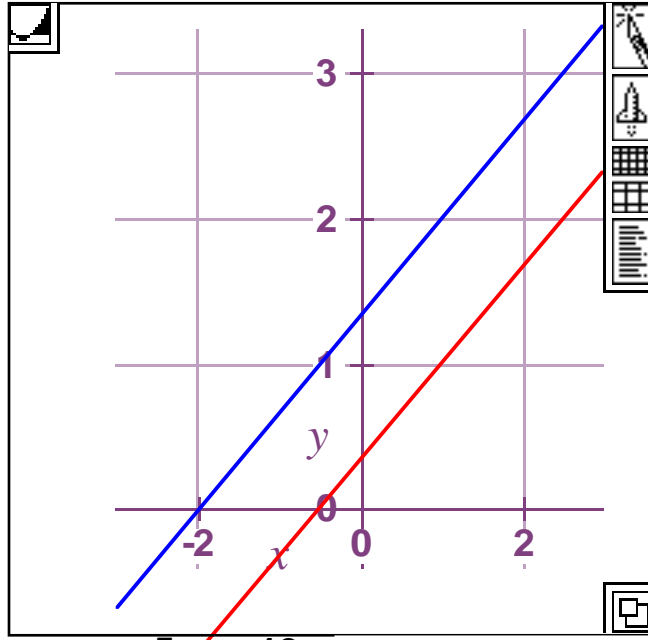
Additionally, note that the graph of each of our original equations is a straight line, and if we graph each line, then their point of intersection, $(1,2)$, represents the solution to the system.



If our system of equations consists of two linear equations like those above whose graphs are straight lines, then there are other possibilities besides the one above where the point of intersection represents our single, unique solution. For example, suppose we have the following system of equations.

$$\begin{aligned}2x - 3y &= -4 \\2x - 3y &= -1\end{aligned}$$

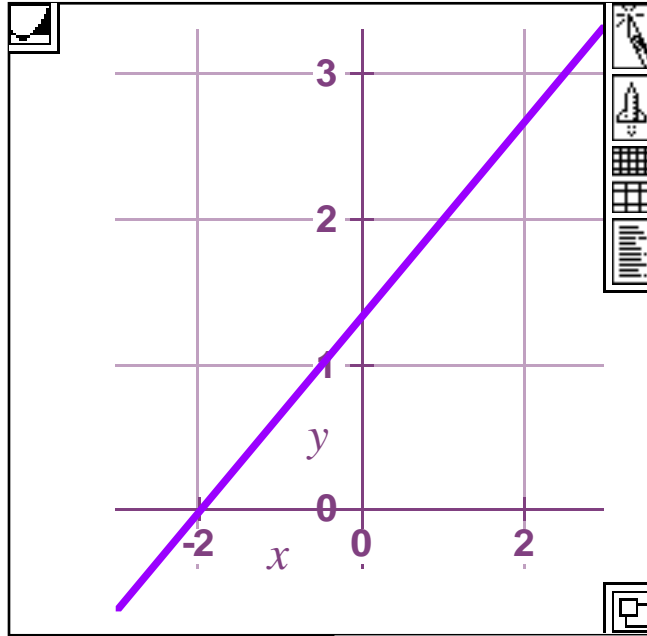
First, if you study the equations, then it should be clear that there can be no solution to the system because in each equation we have $2x - 3y$ on the left hand side, and that can't simultaneously add up to both -4 and -1 . Consequently, when we graph the two equations, we find that we get two parallel lines which never intersect, and, hence, have no simultaneous solution.



However, there is also a third possibility that is represented by the system of linear equations below.

$$\begin{aligned} 2x - 3y &= -4 \\ 4x - 6y &= -8 \end{aligned}$$

Notice that in this system the second equation is just a multiple of the first, and, thus, when graphed, we should get exactly the same line. Hence, in this case, every point on the line is a solution to the system!



From these examples, we (hopefully) can see that there are three possible outcomes when we try to solve a system of two linear equations in two unknowns.

- The equations represent straight lines that intersect in a single point, and there is one, unique solution.
- The equations represent parallel lines that never intersect, and there is not a solution.
- The equations represent the same line, and there are an infinite number of solutions.

Now let me illustrate how we can tell algebraically which case we are in, and I'll begin with our last system of equations where each equation gave us the same line.

$$\begin{aligned}2x - 3y &= -4 \\4x - 6y &= -8\end{aligned}$$

Notice that we can also write this as follows.

$$\begin{aligned} 2x - 3y &= -4 \\ 2 \cdot 2x - 2 \cdot 3y &= -8 \end{aligned}$$

If we now cross multiply our coefficients, then we obtain $2(-2 \cdot 3) = -12$ and $-3(2 \cdot 2) = -12$.

$$\begin{aligned} 2x - 3y &= -4 \\ 2 \cdot 2x - 2 \cdot 3y &= -8 \end{aligned}$$

The important point is that we got the same product in each case, and hence, the difference between the two products is zero. Also, notice that the reason we got the same products is a direct result of the left hand side of the bottom equation being just a multiple of the left hand side of the top equation. Now let's see what happens when our two equations give us two parallel lines.

$$\begin{aligned} 2x - 3y &= -4 \\ 2x - 3y &= -1 \end{aligned}$$

Again, if we cross multiply, then notice that the products are still the same, $2(-3) = -6 = 2(-3)$.

$$\begin{aligned} 2x - 3y &= -4 \\ 2x - 3y &= -1 \end{aligned}$$

However, when we do this cross multiplication with equations that have a single simultaneous solution, this won't happen because the coefficients in the second equation are no longer a multiple of the coefficients in the first equation, $2 \cdot 7 = 14$ and $-3 \cdot 5 = -15$.

$$\begin{aligned} 2x - 3y &= -4 \\ 5x + 7y &= 19 \end{aligned}$$

And now we have some insight into how to determine algebraically if systems of equations like the ones above have a unique solution or not. Here is a streamlined procedure for doing this.

1. Start with a system of equations.

$$\begin{aligned}2x - 3y &= -4 \\5x + 7y &= 19\end{aligned}$$

2. Write it as a *matrix equation*.

$$\begin{bmatrix} 2 & -3 \\ 5 & 7 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -4 \\ 19 \end{bmatrix}$$

3. Cross multiply the numbers in the *coefficient matrix*, and take their difference.

$$(2)(7) - (-3)(5) = 14 + 15 = 29 \neq 0$$

4. If that difference is not zero, then there is a unique solution, and if the difference is zero, then there is either no solution or an infinite number of solutions.

This number we obtained by cross multiplication is called the *determinant* of the *matrix*, and it tells us whether or not a unique solution exists. Furthermore, if we have a 2×2 *matrix* such as $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$, then we denote the *determinant* by enclosing the *matrix* in a pair of vertical bars and performing the following calculation.

$$\text{determinant} = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - cb$$

For a 3×3 *matrix* or *square matrix* of higher order, the procedure is a little bit more involved, but still very similar. For example, here is how you can reduce the problem of finding the *determinant* of a 3×3 *matrix* to one of finding several *determinants* of 2×2 *matrices*.

$$\begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} = a \cdot \begin{vmatrix} e & f \\ h & i \end{vmatrix} - b \cdot \begin{vmatrix} d & f \\ g & i \end{vmatrix} + c \cdot \begin{vmatrix} d & e \\ g & h \end{vmatrix}$$

Quite often these days, one simply uses technology such as a graphing calculator to find the *determinant* of a $n \times n$ *matrix*, but the bottom line is still that if the *determinant* is different from zero, then a unique solution to the system exists, and if the *determinant* is otherwise, then a unique solution fails to exist.

Furthermore, this also means that in a *matrix equation* like $\begin{bmatrix} 2 & -3 \\ 5 & 7 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -4 \\ 19 \end{bmatrix}$, a unique solution exists if and only if the *coefficient matrix* has an inverse, and the *coefficient matrix* has an inverse if and only if its *determinant* is not zero. We'll make use of all this information in our next section where we examine *groups* of *permutation matrices*.

And finally, before we leave *matrix multiplication* behind, there is one other type of multiplication we want to mention, and that is multiplication of a *matrix* by a number or *scalar* as physicists like to call it. It's really pretty simple. We just multiply each element of the *matrix* by whatever our number is. For example,

$$2 \cdot \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 2 \cdot 1 & 2 \cdot 2 \\ 2 \cdot 3 & 2 \cdot 4 \end{bmatrix} = \begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix}$$

Notice, too, that in arithmetic when we write $2 \cdot 3$, that means the same thing as $3 + 3$. It works exactly the same way with *matrices*. Thus,

$$2 \cdot \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 2 \cdot 1 & 2 \cdot 2 \\ 2 \cdot 3 & 2 \cdot 4 \end{bmatrix} = \begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix}$$

and

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 1+1 & 2+2 \\ 3+3 & 4+4 \end{bmatrix} = \begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix}$$

PERMUTATION MATRICES

Recall that in an earlier part of this work we stated that every *group* can be expressed as a *group of permutations* that act on a set of objects, and we even showed how to find the permutation corresponding to a particular element of a *group*. We now want to introduce the topic of *permutation matrices*, and so we'll begin with the following definition.

Definition: A $n \times n$ *permutation matrix* is a *square matrix* with n rows and n columns such that each row and each column contains a single entry of 1 and all the other entries in that row or column are 0.

Here are a few examples of legitimate *permutation matrices*. Notice that each row and column has a single instance of 1 with all the other elements in that row or column being 0.

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

permutation matrices

On the other hand, the following are not *permutation matrices*. The first *matrix* fails because it has nothing but zeros in the first row and third column. The second *matrix* fails because it has two occurrences of 1 in both the second row and the third column. And finally, the third *matrix* fails because it has an entry in the fourth row and second column which is different from 1.

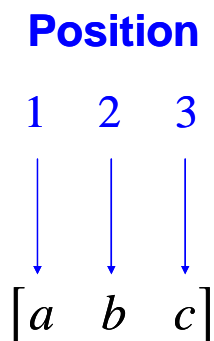
$$\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 \end{bmatrix}$$

not permutation matrices

Now let's give an example of how *permutation matrices* work and also how to find the *matrix* corresponding to the permutation that we're interested in. To begin, let's suppose that we are interested in permutations of the following 3-element set, $\{a,b,c\}$. Recall that the number of permutations we can make of three elements is $3!=3\cdot 2\cdot 1=6$, and for convenience we will express each permutation as a *row matrix*. Thus, in *matrix* form, the six possible permutations are as follows.

$$\begin{array}{ccc} [a & b & c] & [b & a & c] & [c & a & b] \\ [a & c & b] & [b & c & a] & [c & b & a] \end{array}$$

By now you should be familiar with cycle notation for permutations such as $(1,2,3)$. However, remember that what this notation actually means is that whatever is in position 1 is moved to position 2, whatever is in position 2 is moved to position 3, and whatever is in position 3 is moved to position 1. Also, for a *row matrix* like the ones above, we designate the positions in order from left to right.



If we now want to permute $[a \ b \ c]$ according to the cycle $(1,2,3)$, then here's how we construct the corresponding 3×3 *permutation matrix*. We start with the 3×3 *identity matrix*, and since our cycle says that whatever is in position 1 should be moved to position 2, we begin by moving column 1 of our *identity matrix* into the position of column 2.

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \xrightarrow{1 \text{ goes to } 2} \begin{bmatrix} 1 & & \\ & 0 & \\ & 0 & 0 \end{bmatrix}$$

Next our cycle says that what is in position 2 should be moved to position 3, and so we move column 2 of our *identity matrix* to column 3 in the *permutation matrix* we are constructing.

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \xrightarrow{\substack{1 \text{ goes to } 2 \\ 2 \text{ goes to } 3}} \begin{bmatrix} 1 & 0 & \\ & 0 & 1 \\ & 0 & 0 \end{bmatrix}$$

And last, our cycle says to move what is in position 3 to position 1, and so we take column 3 of our *identity matrix* and move it to column 1 in our new *permutation matrix*.

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \xrightarrow{\substack{1 \text{ goes to } 2 \\ 2 \text{ goes to } 3 \\ 3 \text{ goes to } 1}} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

And now we have the *permutation matrix* corresponding to our cycle $(1,2,3)$! However, to verify that this works our next step should be to multiply $[a \ b \ c]$ by

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}. \text{ Notice, though, that we have purposely set things up so that all}$$

multiplication will be done from left to right in order to make this consistent with the way we have been multiplying permutations in GAP and with Rubik's cube. Thus, we get the following which verifies the correctness of our procedure.

$$[a \ b \ c] \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} = [c \ a \ b]$$

And sure enough, a has moved to position 2, b has moved to position 3, and c has moved to position 1.

Now let's suppose that we have a product of two cycles such as $(1,2,3)(1,3)$. In this case we can convert each cycle to a *permutation matrix* and apply them in order from left to right. Here is the result.

$$[a \ b \ c] \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} = [c \ a \ b] \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} = [b \ a \ c]$$

Notice now that the product of our two cycles is $(1,2,3)(1,3) = (1,2)$. If we write this

last cycle as a *permutation matrix*, then we get $\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, and if we multiply

$[a \ b \ c]$ by this *matrix*, then the end result is the same as what we got when we multiplied $[a \ b \ c]$ by the two separate *permutation matrices* above. And that is how it should be.

$$[a \ b \ c] \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = [b \ a \ c]$$

Now let's dig a little deeper into our $n \times n$ *permutation matrices*. It turns out that every *permutation matrix* will have a *determinant* that is equal to either 1 or -1. In particular, that means that every *permutation matrix* has an *inverse*. Also, if our permutation is an *even permutation* (equivalent to the product of an even number of *transpositions*), then the *determinant* of our *permutation matrix* will be 1, and if our permutation is *odd permutation*, then the *determinant* of our *permutation matrix* will be -1. This is an alternative and perhaps better way to define *even* and *odd permutations*.

It is also true that the product of two *permutation matrices* will be a *permutation matrix*, that the *associative law* holds for *matrix multiplication*, and that the $n \times n$ *identity matrix* is an *identity element* for multiplication of $n \times n$ *matrices*. Hence, the set of $n \times n$ *permutation matrices* forms a *group*, and the size or order of this *group* is $n!$, the number of permutations that can be made of n objects. However, we cannot always count on the *commutative law* holding as is shown by the following example.

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

In summary,

- Every *group* can be expressed as a *group* of permutations of n objects for some number n .
- These permutations can be expressed as cycles, and these cycles can be expressed as *permutation matrices*.
- If the cycle is *even*, then the *determinant* of the *permutation matrix* is 1, and if the cycle is *odd*, then the *determinant* of the *permutation matrix* is -1.
- The $n \times n$ *permutation matrices* form a *group* of size $n!$.
- *Matrix theory* gives us new tools for studying *group theory*!

OTHER MATRIX GROUPS

The one thing that all *matrix groups* have in common is that they are all *square matrices* and they are all *invertible*. In other words, they all have *inverses* such that the product of the original *matrix* with its *inverse* gives us back the *identity matrix*. Additionally, the property of being *invertible* is equivalent to the *determinant* of the *matrix* being nonzero. In our last chapter, you were introduced to *groups of permutation matrices*, and now we'll list some other *matrix groups*, and some of these will have specific applications while others seem more abstract.

The General Linear group of degree n with real number entries

This is the *group* of all $n \times n$ *square matrices* with *real number* entries that have *determinants* different from zero, and that means that these *matrices* are *invertible*. The *group* is symbolized by $GL_n(\mathbb{R})$ where n tells us that we are talking about $n \times n$ *square matrices* while \mathbb{R} means that the entries are *real numbers*, numbers from the number line. Notice that the *permutation matrices* we looked at in the previous chapter form a *subgroup* of $GL_n(\mathbb{R})$. Also, notice

that $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ is an element of $GL_2(\mathbb{R})$ since its *determinant* is $1 \cdot 4 - 2 \cdot 3 = 4 - 6 = -2 \neq 0$,

but $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ is not an element of $GL_3(\mathbb{R})$ since this *matrix* does not have its

determinant equal to zero.

Invertible Upper Triangular Matrices

A square matrix is an *invertible upper triangular matrix* if its *determinant* is nonzero while all its entries below the main diagonal from upper left to lower right are zero. An good example of a 3×3 *invertible upper triangular matrix* is:

$$\begin{bmatrix} 1 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \end{bmatrix}$$

Notice that if we manually compute the *determinant* of this *matrix*, then we can see that its *determinant* will just equal the product of the entries along the diagonal. This sort of thing will always happen!

$$\begin{aligned} \begin{vmatrix} 1 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \end{vmatrix} &= 1 \cdot \begin{vmatrix} 4 & 5 \\ 0 & 6 \end{vmatrix} - 2 \cdot \begin{vmatrix} 0 & 5 \\ 0 & 6 \end{vmatrix} + 3 \cdot \begin{vmatrix} 0 & 4 \\ 0 & 0 \end{vmatrix} \\ &= 1(4 \cdot 6 - 5 \cdot 0) - 2(0 \cdot 6 - 5 \cdot 0) + 3(0 \cdot 0 - 4 \cdot 0) \\ &= 1 \cdot 4 \cdot 6 = 24 \end{aligned}$$

Also note that the product of two *invertible upper triangular matrices* is another *invertible upper triangular matrix*, as we illustrate with the product below.

$$\begin{bmatrix} 1 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \end{bmatrix} \begin{bmatrix} 7 & 8 & 9 \\ 0 & 10 & 11 \\ 0 & 0 & 12 \end{bmatrix} = \begin{bmatrix} 7 & 28 & 67 \\ 0 & 40 & 104 \\ 0 & 0 & 72 \end{bmatrix}$$

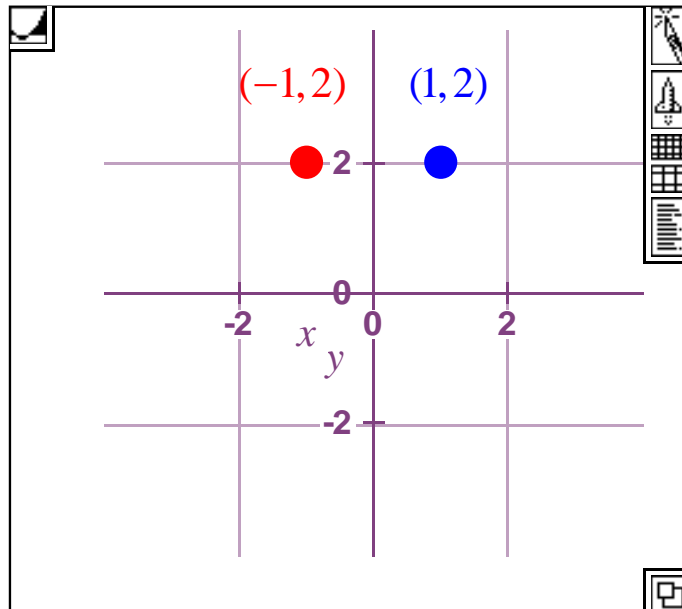
Invertible Lower Triangular Matrices

This *matrix group* is very similar to the previous one with the change that it is now the upper triangle above the diagonal that consists of entries of zero. An example of such a *matrix* is:

$$\begin{bmatrix} 1 & 0 & 0 \\ 2 & 3 & 0 \\ 4 & 5 & 6 \end{bmatrix}$$

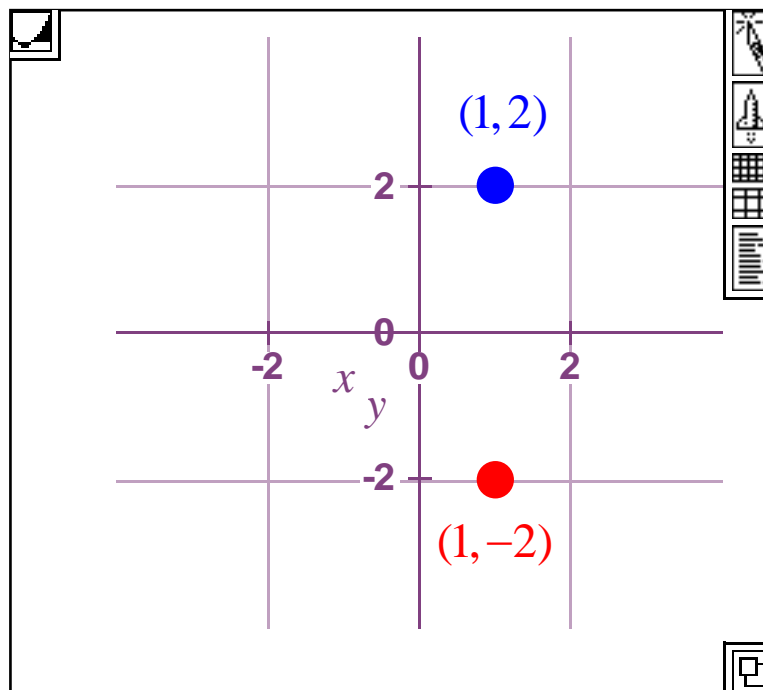
Reflection Matrices

Suppose we start with the 1×2 *matrix* $[1 \ 2]$ and we multiply it on the right by $\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$. Then the result is $[1 \ 2] \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} = [-1 \ 2]$. Notice that if we consider $[1 \ 2]$ as representing the point $(1,2)$, then the result is equivalent to a reflection of this point across the y -axis to transform it into the point $(-1,2)$.



More generally, since $[a \ b] \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} = [-a \ b]$, multiplication on the right by this *matrix* will always reflect a point (a,b) about the vertical axis. In other words, this *matrix* always results in a vertical reflection. Furthermore, since $\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, this *matrix* generates a *group* of order 2.

Similarly, since $[a \ b] \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = [a \ -b]$, multiplication on the right by this *matrix* will reflect a point (a,b) about the x -axis, the horizontal axis, and again since $\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, this *matrix* $\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ generates a *group* of order 2.

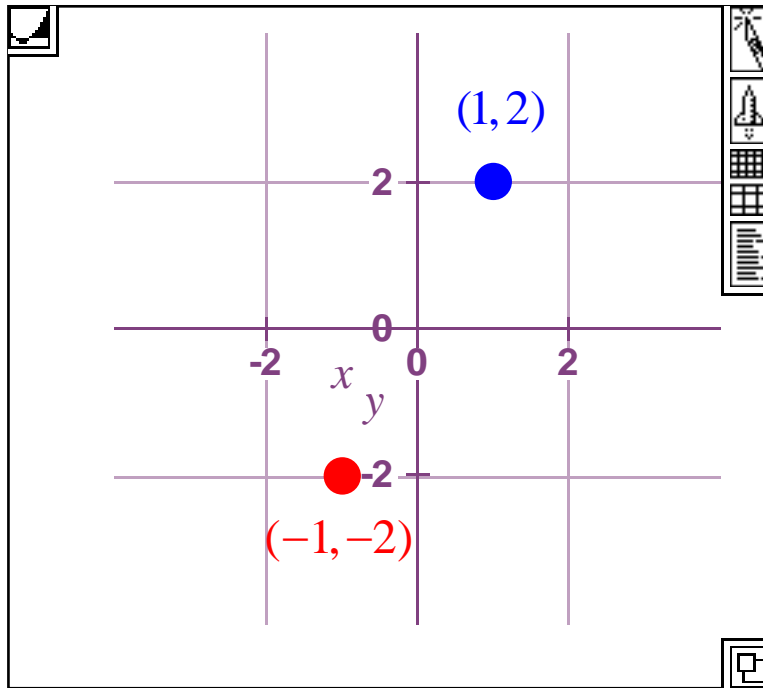


And finally, let's consider multiplication by the *matrix* $\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$. Since

$[a \ b] \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = [-a \ -b]$, multiplication on the right by this *matrix* transforms (a,b)

into $(-a, -b)$. This is equivalent to a reflection about the y -axis followed by a reflection about the x -axis. We also call this a reflection through the origin, the point $(0,0)$, and again the *matrix* $\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$ generates a *cyclic group* of order 2,

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



And finally, notice that we can think of all three *groups of reflection matrices* as *groups* that act upon points in the coordinate plane. Additionally, we could consider the group generated by all three of our reflection matrices, and if we let $I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $A = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$, $B = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$, and $C = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$, then a little exploration will verify that $I = AA = BB = CC$, $AB = BA = C$, and the *group* generated by A and B is *isomorphic* to $C_2 \times C_2 \cong \mathbb{Z}_2 \times \mathbb{Z}_2$, our old friend the familiar *Klein 4-group*.

Invertible Diagonal Matrices

These are *invertible square matrices* that have nonzero entries along the diagonal from upper left to lower right and zero everywhere else. Also, notice that the *determinant* of a *diagonal matrix* is equal to the product of the numbers along the diagonal and that the product of two *diagonal matrices* is another *diagonal matrix*. Thus, the *invertible $n \times n$ diagonal matrices* form a *group*.

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

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \cdot \begin{bmatrix} 4 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 6 \end{bmatrix} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 18 \end{bmatrix}$$

Invertible Scalar Matrices

The *invertible scalar matrices* form a *subgroup* of the *invertible diagonal matrices*, and these are going to be *diagonal matrices* that have the same nonzero number along the diagonal such as $\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$ and $\begin{bmatrix} 1/2 & 0 \\ 0 & 1/2 \end{bmatrix}$. In physics, a number is often called a *scalar* because multiplication by that number change the scale or size of that number. Thus, for example, if we multiply $\begin{bmatrix} 4 & 6 \end{bmatrix}$ on the right by $\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$, then each number in the first *matrix* doubles in size.

$$\begin{bmatrix} 4 & 6 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 8 & 12 \end{bmatrix}$$

Similarly, if we multiply $[4 \ 6]$ on the right by $\begin{bmatrix} 1/2 & 0 \\ 0 & 1/2 \end{bmatrix}$, then each number in the first *matrix* is halved in size.

$$[4 \ 6] \begin{bmatrix} 1/2 & 0 \\ 0 & 1/2 \end{bmatrix} = [2 \ 3]$$

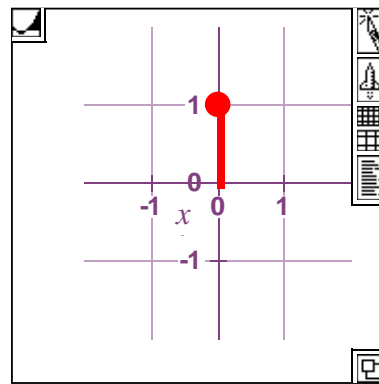
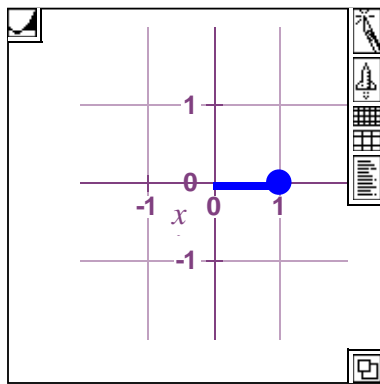
Now imagine that we have a picture such that each point in the picture corresponds to a point in the coordinate plane. Then *scalar matrices* could be used to either magnify or shrink that image, and you can imagine how a more sophisticated version of this could be used for generating special effects in movies!



Rotation Matrices

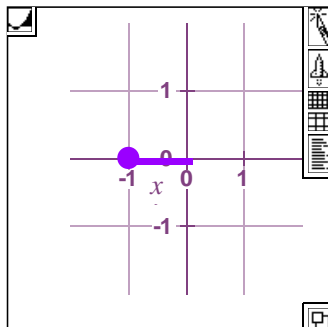
And now here is an example that will involve some knowledge of trigonometry. If we use the *row matrix* $[1 \ 0]$ to represent the point $(1,0)$ in the plane and if we draw a line from the point $(0,0)$ to $(1,0)$, then multiplication on the right by the *matrix* $\begin{bmatrix} \cos(90^\circ) & \sin(90^\circ) \\ -\sin(90^\circ) & \cos(90^\circ) \end{bmatrix}$ has the effect of producing a 90° rotation counterclockwise.

$$[1 \ 0] \begin{bmatrix} \cos(90^\circ) & \sin(90^\circ) \\ -\sin(90^\circ) & \cos(90^\circ) \end{bmatrix} = [1 \ 0] \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} = [0 \ 1]$$



And if we multiply on the right by this *matrix* a second time, then the result is a 180° counterclockwise rotation of the point $(1,0)$ about the origin.

$$[1 \ 0] \begin{bmatrix} \cos(90^\circ) & \sin(90^\circ) \\ -\sin(90^\circ) & \cos(90^\circ) \end{bmatrix} \begin{bmatrix} \cos(90^\circ) & \sin(90^\circ) \\ -\sin(90^\circ) & \cos(90^\circ) \end{bmatrix} = [1 \ 0] \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} = [-1 \ 0]$$



More generally, multiplication on the right by the *matrix* $\begin{bmatrix} \cos(\theta^\circ) & \sin(\theta^\circ) \\ -\sin(\theta^\circ) & \cos(\theta^\circ) \end{bmatrix}$ produces a counterclockwise rotation through an angle of θ° , and thus, we call it a *rotation matrix*. Furthermore, consider the following *matrix* product.

$$\begin{bmatrix} \cos(90^\circ) & \sin(90^\circ) \\ -\sin(90^\circ) & \cos(90^\circ) \end{bmatrix} \begin{bmatrix} \cos(45^\circ) & \sin(45^\circ) \\ -\sin(45^\circ) & \cos(45^\circ) \end{bmatrix}$$

On the one hand, this should be equivalent to a rotation through $90^\circ + 45^\circ = 135^\circ$.

$$\begin{bmatrix} \cos(135^\circ) & \sin(135^\circ) \\ -\sin(135^\circ) & \cos(135^\circ) \end{bmatrix}$$

But on the other hand, this is also equal to,

$$\begin{aligned} & \begin{bmatrix} \cos(90^\circ) & \sin(90^\circ) \\ -\sin(90^\circ) & \cos(90^\circ) \end{bmatrix} \begin{bmatrix} \cos(45^\circ) & \sin(45^\circ) \\ -\sin(45^\circ) & \cos(45^\circ) \end{bmatrix} \\ &= \begin{bmatrix} \cos(90^\circ)\cos(45^\circ) - \sin(90^\circ)\sin(45^\circ) & \cos(90^\circ)\sin(45^\circ) + \sin(90^\circ)\cos(45^\circ) \\ -\sin(90^\circ)\cos(45^\circ) - \cos(90^\circ)\sin(45^\circ) & -\sin(90^\circ)\sin(45^\circ) + \cos(90^\circ)\cos(45^\circ) \end{bmatrix} \end{aligned}$$

In particular this suggests that,

$$\cos(135^\circ) = \cos(90^\circ + 45^\circ) = \cos(90^\circ)\cos(45^\circ) - \sin(90^\circ)\sin(45^\circ)$$

And,

$$\sin(135^\circ) = \sin(90^\circ + 45^\circ) = \cos(90^\circ)\sin(45^\circ) + \sin(90^\circ)\cos(45^\circ)$$

And in this, those who remember their trigonometry will recognize two very familiar formulas.

$$\cos(A + B) = \cos(A)\cos(B) - \sin(A)\sin(B)$$

$$\sin(A + B) = \cos(A)\sin(B) + \sin(A)\cos(B)$$

Additionally, just as one can imagine that *matrices* that do reflections and scaling could be useful in computer animations, so can *matrices* that do rotations also be useful.

Orthogonal Matrices

A $n \times n$ *orthogonal matrix* is a *matrix* whose *inverse* is the same as its *transpose*, and by *transpose* we simply mean the *matrix* that results from converting all the original *matrix's* rows to columns. For example, if $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$, then $A^T = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$.

Now guess what? You already know some *orthogonal matrices* because it turns out that the *permutation matrices* we looked at in the previous chapter form a *subgroup* of the *group* of $n \times n$ *orthogonal matrices*. For example, if

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

then,

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

and,

$$A \cdot A^T = A \cdot A^{-1} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The usual notation for the *group* of $n \times n$ *orthogonal matrices* is $O(n)$. Also, a particular *subgroup* of this *group* consists of $n \times n$ *orthogonal matrices* that have *determinant* equal to 1. This *subgroup* is called the $n \times n$ *special orthogonal group*, and it is denoted by $SO(n)$

If we now allow the entries of our *matrix* to also include *complex numbers* (numbers of the form $a+bi$ where a and b are *real numbers* and i is the square root of -1), then we call a *square matrix* a *unitary matrix* if its *inverse* is equal to its *conjugate transpose*. Of course, the big question now is what do we mean by a *conjugate transpose*? We'll illustrate this by an example. First, if we have a *complex number* such as $2+3i$, then its *conjugate* is $2-3i$. And now, if $B = \begin{bmatrix} 1+i & 2-4i \\ 3 & 4+8i \end{bmatrix}$, then we find its *conjugate transpose* by simultaneously taking its *transpose* and replacing the coefficient in front of each occurrence of i by its opposite. Thus, the *conjugate transpose* of B is $B^* = \begin{bmatrix} 1-i & 3 \\ 2+4i & 4-8i \end{bmatrix}$. And if our original *matrix* is a *unitary matrix*, then we would have that $B^* = B^{-1}$. Also, the *group* of $n \times n$ *unitary matrices* is denoted by $U(n)$.

As was the case with $n \times n$ *orthogonal matrices*, the $n \times n$ *unitary matrices* have an important *subgroup* called the *special unitary group*. This is the *group* of *unitary matrices* that have *determinant* equal to 1, and it is denoted by $SU(n)$. Also, the *special unitary group* $SU(3)$ has a particular application to physics. Applications of *matrix groups* to physics usually involve symmetry and permutations, but it is also the case that many of these applications also involve much more advanced topics such as *eigenvalues* and *Hilbert space*. These topics are beyond the scope of this work, but we can at least give something of

the gist of how $SU(3)$ is involved with physics if you think of its elements as being akin to *permutation matrices*.

In physics, it is understood that fundamental particles like protons and neutrons are composed of even more basic particles known as quarks. Thus, for now think of quarks as *LEGO* pieces of different colors that may be combined in different orders. Also suppose that each different ordering or arrangement of our *Lego* pieces results in a different particle. In this case, we can think of our different arrangements corresponding to *permutation matrices*, and we can now imagine these *permutation matrices* as also being elements of $SU(3)$. Then in this instance, the different elements of $SU(3)$ will describe all the different particles that can be constructed from our quarks. And it was exactly this kind of analysis done by physicist Murray Gell-Mann that led him to hypothesize in 1962 the existence of a yet unobserved particle. This predicted particle was later found in 1964, and for his *group theory* inspired analysis, Murray Gell-Mann was awarded the Nobel Prize in physics in 1969!

HOW TO USE GAP (PART 7)

Once again we will begin by repeating all the GAP commands with learned up to this point so that you don't have to reference earlier parts of this work, and then at the end we'll introduce in red a few GAP commands that are useful for exploring *matrices*. Enjoy!

1. *How can I redisplay the previous command in order to edit it?*

Press down on the control key and then also press p. In other words, "Ctrl p".

2. *If the program gets in a loop and shows you the prompt "brk>" instead of "gap>", how can I exit the loop?*

Press down on the control key and then also press d. In other words, "Ctrl d".

3. *How can I exit the program?*

Either click on the "close" box for the window, or type "quit;" and press "Enter."

4. *How do I find the inverse of a permutation?*

```
gap> a:=(1,2,3,4);  
(1,2,3,4)  
gap> a^-1;
```

(1,4,3,2)

5. *How can I multiply permutations and raise permutations to powers?*

```
gap> (1,2)*(1,2,3);  
(1,3)
```

```
gap> (1,2,3)^2;  
(1,3,2)
```

```
gap> (1,2,3)^-1;  
(1,3,2)
```

```
gap> (1,2,3)^-2;  
(1,2,3)
```

```
gap> a:=(1,2,3);  
(1,2,3)
```

```
gap> b:=(1,2);  
(1,2)
```

```
gap> a*b;  
(2,3)
```

```
gap> a^2;  
(1,3,2)
```

```
gap> a^-2;  
(1,2,3)
```

```
gap> a^3;
```

```
()
```

```
gap> a^-3;
```

```
()
```

```
gap> (a*b)^2;
```

```
()
```

```
gap> (a*b)^3;
```

```
(2,3)
```

6. *How can I create a group from permutations, find the size of the group, and find the elements in the group?*

```
gap> a:=(1,2);
```

```
(1,2)
```

```
gap> b:=(1,2,3);
```

```
(1,2,3)
```

```
gap> g1:=Group(a,b);
```

```
Group([ (1,2), (1,2,3) ])
```

```
gap> Size(g1);
```

```
6
```

```
gap> Elements(g1);
```

```
[ (), (2,3), (1,2), (1,2,3), (1,3,2), (1,3) ]
```

```
gap> g2:=Group([(1,2),(1,2,3)]);
```

```
Group([ (1,2), (1,2,3) ])
```

```
gap> g3:=Group((1,2),(2,3,4));
```

```
Group([ (1,2), (2,3,4) ])
```

7. *How can I create a cyclic group of order 3?*

```
gap> a:=(1,2,3);
```

```
(1,2,3)
```

```
gap> g1:=Group(a);
```

```
Group([ (1,2,3) ])
```

```
gap> Size(g1);
```

```
3
```

```
gap> Elements(g1);
```

```
[ (), (1,2,3), (1,3,2) ]
```

```
gap> g2:=Group((1,2,3));
```

```
Group([ (1,2,3) ])
```

```
gap> g3:=CyclicGroup(IsPermGroup, 3);
```

```
Group([ (1, 2, 3) ])
```


8. *How can I create a multiplication table for the cyclic group of order 3 that I just created?*

```
gap> ShowMultiplicationTable(g1);
```

```
*      | ()      (1,2,3) (1,3,2)
-----+-----
()      | ()      (1,2,3) (1,3,2)
(1,2,3) | (1,2,3) (1,3,2) ()
(1,3,2) | (1,3,2) ()      1,2,3
```

9. *How do I determine if a group is abelian?*

```
gap> g1:=Group((1,2,3));
Group([ (1,2,3) ])
```

```
gap> IsAbelian(g1);
true
```

```
gap> g2:=Group((1,2),(1,2,3));
Group([ (1,2), (1,2,3) ])
```

```
gap> IsAbelian(g2);
false
```

10. *What do I type in order to get help for a command like “Elements?”*

```
gap> ?Elements
```

11. *How do I find all subgroups of a group?*

```

gap> a:=(1, 2, 3);
(1, 2, 3)
gap> b:=(2, 3);
(2, 3)

gap> g:=Group(a, b);
Group([ (1, 2, 3), (2, 3) ])

gap> Size(g);
6

gap> Elements(g);
[ (), (2, 3), (1, 2), (1, 2, 3), (1, 3, 2), (1, 3) ]

gap> h:=AllSubgroups(g);
[ Group(()), Group([ (2, 3) ]), Group([ (1, 2) ]), Group([ (1, 3) ]),
Group([ (1, 2, 3) ]), Group([ (1, 2, 3), (2, 3) ]) ]

gap> List(h, i->Elements(i));
[ [ () ], [ (), (2, 3) ], [ (), (1, 2) ], [ (), (1, 3) ], [ (), (1, 2, 3),
(1, 3, 2) ], [ (), (2, 3), (1, 2), (1, 2, 3), (1, 3, 2), (1, 3) ] ]

gap> Elements(h[1]);
[ () ]

gap> Elements(h[2]);
[ (), (2, 3) ]

gap> Elements(h[3]);
[ (), (1, 2) ]

gap> Elements(h[4]);
[ (), (1, 3) ]

gap> Elements(h[5]);
[ (), (1, 2, 3), (1, 3, 2) ]

gap> Elements(h[6]);
[ (), (2, 3), (1, 2), (1, 2, 3), (1, 3, 2), (1, 3) ]

```

12. How do I find the subgroup generated by particular permutations?

```

gap> g:=Group((1, 2), (1, 2, 3));
Group([ (1, 2), (1, 2, 3) ])

gap> Elements(g);
[ (), (2, 3), (1, 2), (1, 2, 3), (1, 3, 2), (1, 3) ]

gap> h:=Subgroup(g, [(1, 2)]);
Group([ (1, 2) ])

gap> Elements(h);
[ (), (1, 2) ]

```

13. How do I determine if a subgroup is normal?

```

gap> g:=Group((1, 2), (1, 2, 3));
Group([ (1, 2), (1, 2, 3) ])

```

```

gap> h1:=Group((1,2));
Group([ (1,2) ])
gap> IsNormal(g,h1);

gap> h2:=Group((1,2,3));
Group([ (1,2,3) ])

gap> IsNormal(g,h2);
true

```

14. How do I find all normal subgroups of a group?

```

gap> g:=Group((1,2),(1,2,3));
Group([ (1,2), (1,2,3) ])

gap> Elements(g);
[ (), (2,3), (1,2), (1,2,3), (1,3,2), (1,3) ]

gap> n:=NormalSubgroups(g);
[ Group([ (1,2), (1,2,3) ]), Group([ (1,3,2) ]), Group(()) ]

gap> Elements(n[1]);
[ (), (2,3), (1,2), (1,2,3), (1,3,2), (1,3) ]

gap> Elements(n[2]);
[ (), (1,2,3), (1,3,2) ]

gap> Elements(n[3]);
[ () ]

```

15. How do I determine if a group is simple?

```

gap> g:=Group((1,2),(1,2,3));
Group([ (1,2), (1,2,3) ])

gap> Elements(g);
[ (), (2,3), (1,2), (1,2,3), (1,3,2), (1,3) ]

gap> IsSimple(g);
false

gap> h:=Group((1,2));
Group([ (1,2) ])

gap> Elements(h);
[ (), (1,2) ]

gap> IsSimple(h);
true

```

16. How do I find the right cosets of a subset H of G ?

```
gap> g:=Group([(1,2,3),(1,2)]);
Group([ (1,2,3), (1,2) ])

gap> Elements(g);
[ (), (2,3), (1,2), (1,2,3), (1,3,2), (1,3) ]

gap> h:=Subgroup(g, [(1,2)]);
Group([ (1,2) ])

gap> Elements(h);
[ (), (1,2) ]

gap> c:=RightCosets(g, h);
[ RightCoset(Group([ (1,2) ]), ()), RightCoset(Group([ (1,2) ]), (1,3,2)),
  RightCoset(Group([ (1,2) ]), (1,2,3)) ]

gap> List(c, i->Elements(i));
[ [ (), (1,2) ], [ (2,3), (1,3,2) ], [ (1,2,3), (1,3) ] ]
gap> Elements(c[1]);
[ (), (1,2) ]

gap> Elements(c[2]);
[ (2,3), (1,3,2) ]

gap> Elements(c[3]);
[ (1,2,3), (1,3) ]

gap> rc:=RightCoset(h, (1,2,3));
RightCoset(Group([ (1,2) ]), (1,2,3))

gap> Elements(rc);
[ (1,2,3), (1,3) ]

gap> rc:=h*(1,2,3);
RightCoset(Group([ (1,2) ]), (1,2,3))

gap> Elements(rc);
[ (1,2,3), (1,3) ]
```

17. How can I create a quotient (factor) group?

```
gap> g:=Group([(1,2,3),(1,2)]);
Group([ (1,2,3), (1,2) ])

gap> Elements(g);
[ (), (2,3), (1,2), (1,2,3), (1,3,2), (1,3) ]

gap> n:=Group((1,2,3));
Group([ (1,2,3) ])

gap> Elements(n);
[ (), (1,2,3), (1,3,2) ]

gap> IsNormal(g, n);
true
```

```

gap> c:=RightCosets(g, n);
[ RightCoset(Group([ (1, 2, 3) ]), ()), RightCoset(Group([ (1, 2, 3) ]), (2, 3)) ]

gap> Elements(c[1]);
[ (), (1, 2, 3), (1, 3, 2) ]

gap> Elements(c[2]);
[ (2, 3), (1, 2), (1, 3) ]

gap> f:=FactorGroup(g, n);
Group([ f1 ])

gap> Elements(f);
[ <identity> of ..., f1 ]

gap> ShowMultiplicationTable(f);
*
-----+-----
<identity> of ... | <identity> of ... f1
-----+-----
f1                 | f1                 <identity> of ...

```

18. How do I find the center of a group?

```

gap> a:=(1, 2, 3);
(1, 2, 3)

gap> b:=(2, 3);
(2, 3)

gap> g:=Group(a, b);
Group([ (1, 2, 3), (2, 3) ])

gap> Center(g);
Group(())

gap> c:=Center(g);
Group(())

gap> Elements(c);
[ () ]

gap> a:=(1, 2, 3, 4);
(1, 2, 3, 4)

gap> b:=(1, 3);
(1, 3)

gap> g:=Group(a, b);
Group([ (1, 2, 3, 4), (1, 3) ])

gap> c:=Center(g);
Group([ (1, 3)(2, 4) ])

gap> Elements(c);
[ (), (1, 3)(2, 4) ]

```

19. How do I find the commutator (derived) subgroup of a group?

```

gap> a:=(1, 2, 3);
(1, 2, 3)

gap> b:=(2, 3);
(2, 3)

gap> g:=Group(a, b);
Group([ (1, 2, 3), (2, 3) ])

gap> d:=DerivedSubgroup(g);
Group([ (1, 3, 2) ])

gap> Elements(d);
[ (), (1, 2, 3), (1, 3, 2) ]

```

```

gap> a:=(1, 2, 3, 4);
(1, 2, 3, 4)

gap> b:=(1, 3);
(1, 3)

gap> g:=Group(a, b);
Group([ (1, 2, 3, 4), (1, 3) ])

gap> d:=DerivedSubgroup(g);
Group([ (1, 3)(2, 4) ])

gap> Elements(d);
[ (), (1, 3)(2, 4) ]

```

20. How do I find all Sylow p -subgroups for a given group?

```

gap> a:=(1, 2, 3);
(1, 2, 3)

gap> b:=(2, 3);
(2, 3)

gap> g:=Group(a, b);
Group([ (1, 2, 3), (2, 3) ])

gap> Size(g);
6

gap> FactorsInt(6);
[ 2, 3 ]

gap> sylow2:=SylowSubgroup(g, 2);
Group([ (2, 3) ])

gap> IsNormal(g, sylow2);
false

gap> c:=ConjugateSubgroups(g, sylow2);
[ Group([ (2, 3) ]), Group([ (1, 3) ]), Group([ (1, 2) ]) ]

gap> Elements(c[1]);
[ (), (2, 3) ]

gap> Elements(c[2]);
[ (), (1, 3) ]

gap> Elements(c[3]);

```

```
[ () , (1, 2) ]
gap> syl ow3: =Syl owSubgroup(g, 3);
Group([ (1, 2, 3) ])

gap> IsNormal (g, syl ow3);
true

gap> Elements(syl ow3);
[ () , (1, 2, 3), (1, 3, 2) ]
```

21. How can I create the Rubik's cube group using GAP?

First you need to save the following permutations as a pure text file with the name rubik.txt to your C-drive before you can import it into GAP.

```
r:=(25, 27, 32, 30) (26, 29, 31, 28) (3, 38, 43, 19) (5, 36, 45, 21) (8, 33, 48, 24) ;
l:=(9, 11, 16, 14) (10, 13, 15, 12) (1, 17, 41, 40) (4, 20, 44, 37) (6, 22, 46, 35) ;
u:=(1, 3, 8, 6) (2, 5, 7, 4) (9, 33, 25, 17) (10, 34, 26, 18) (11, 35, 27, 19) ;
d:=(41, 43, 48, 46) (42, 45, 47, 44) (14, 22, 30, 38) (15, 23, 31, 39) (16, 24, 32, 40) ;
f:=(17, 19, 24, 22) (18, 21, 23, 20) (6, 25, 43, 16) (7, 28, 42, 13) (8, 30, 41, 11) ;
b:=(33, 35, 40, 38) (34, 37, 39, 36) (3, 9, 46, 32) (2, 12, 47, 29) (1, 14, 48, 27) ;
```

And now you can read the file into GAP and begin exploring.

```
gap> Read("C: /rubi k. txt");

gap> rubi k: =Group(r, l, u, d, f, b);
<permutation group with 6 generators>

gap> Si ze(rubi k);
43252003274489856000
```

22. How can I find the center of the Rubik's cube group?

```
gap> c: =Center(rubi k);
Group([ (2, 34) (4, 10) (5, 26) (7, 18) (12, 37) (13, 20) (15, 44) (21, 28) (23, 42) (29, 36) (31, 45) (39, 47) ])

gap> Si ze(c);
2

gap> El ements(c);
[ () , (2, 34) (4, 10) (5, 26) (7, 18) (12, 37) (13, 20) (15, 44) (21, 28) (23, 42) (29, 36) (31, 45) (39, 47) ]
```


Group([(4, 36, 31, 39, 42, 12, 5, 21, 15, 13, 7)(10, 29, 45, 47, 23, 37, 26, 28, 44, 20, 18)])

```
gap> Elements(sylow11);
[ (), (4, 5, 36, 21, 31, 15, 39, 13, 42, 7, 12)(10, 26, 29, 28, 45, 44, 47, 20, 23, 18, 37),
(4, 7, 13, 15, 21, 5, 12, 42, 39, 31, 36)(10, 18, 20, 44, 28, 26, 37, 23, 47, 45, 29),
(4, 12, 7, 42, 13, 39, 15, 31, 21, 36, 5)(10, 37, 18, 23, 20, 47, 44, 45, 28, 29, 26),
(4, 13, 21, 12, 39, 36, 7, 15, 5, 42, 31)(10, 20, 28, 37, 47, 29, 18, 44, 26, 23, 45),
(4, 15, 12, 31, 7, 21, 42, 36, 13, 5, 39)(10, 44, 37, 45, 18, 28, 23, 29, 20, 26, 47),
(4, 21, 39, 7, 5, 31, 13, 12, 36, 15, 42)(10, 28, 47, 18, 26, 45, 20, 37, 29, 44, 23),
(4, 31, 42, 5, 15, 7, 36, 39, 12, 21, 13)(10, 45, 23, 26, 44, 18, 29, 47, 37, 28, 20),
(4, 36, 31, 39, 42, 12, 5, 21, 15, 13, 7)(10, 29, 45, 47, 23, 37, 26, 28, 44, 20, 18),
(4, 39, 5, 13, 36, 42, 21, 7, 31, 12, 15)(10, 47, 26, 20, 29, 23, 28, 18, 45, 37, 44),
(4, 42, 15, 36, 12, 13, 31, 5, 7, 39, 21)(10, 23, 44, 29, 37, 20, 45, 26, 18, 47, 28) ]
```

```
gap> IsNormal(rubik, sylow2);
false
```

```
gap> IsNormal(rubik, sylow3);
false
```

```
gap> IsNormal(rubik, sylow5);
false
```

```
gap> IsNormal(rubik, sylow7);
false
```

```
gap> IsNormal(rubik, sylow11);
false
```

NOTE: All of the *Sylow p-subgroups* found above have *conjugates*, but the sheer size of the *Rubik's cube group* makes it too difficult to pursue them on a typical desktop computer.

26. How do I determine if a group is cyclic?

```
gap> a:=(1, 2, 3)*(4, 5, 6, 7);
(1, 2, 3)(4, 5, 6, 7)

gap> g:=Group(a);
Group([ (1, 2, 3)(4, 5, 6, 7) ])

gap> Size(g);
12

gap> IsCyclic(g);
true
```

27. How do I create a dihedral group with $2n$ elements for an n -sided regular polygon?

```
gap> d4:=DihedralGroup(IsPermGroup, 8);
Group([ (1, 2, 3, 4), (2, 4) ])
```

```
gap> Elements(d4);
[ (), (2, 4), (1, 2)(3, 4), (1, 2, 3, 4), (1, 3), (1, 3)(2, 4), (1, 4, 3, 2), (1, 4)(2, 3) ]
```

28. How can I express the elements of a dihedral group as rotations and flips rather than as permutations?

```
gap> d3:=DihedralGroup(6);
<pc group of size 6 with 2 generators>
```

```
gap> Elements(d3);
[ <identity> of ..., f1, f2, f1*f2, f2^2, f1*f2^2 ]
```

```
gap> ShowMultiplicationTable(d3);
```

	<identity> of ...	f1	f2	f1*f2	f2^2	f1*f2^2
<identity> of ...	<identity> of ...	f1	f2	f1*f2	f2^2	f1*f2^2
f1	f1	<identity> of ...	f2	f1*f2	f2^2	f1*f2^2
f2	f2	f1*f2^2	f2^2	f1	<identity> of ...	f1*f2
f1*f2	f1*f2	f2^2	f1*f2^2	<identity> of ...	f1	f2
f2^2	f2^2	f1*f2	<identity> of ...	f1*f2^2	f2	f1
f1*f2^2	f1*f2^2	f2	f1	f2^2	f1*f2	f1

29. How do I create a symmetric group of degree n with n! elements?

```
gap> s4:=SymmetricGroup(4);
Sym([ 1 .. 4 ])
```

```
gap> Size(s4);
24
```

```
gap> Elements(s4);
[ (), (3, 4), (2, 3), (2, 3, 4), (2, 4, 3), (2, 4), (1, 2), (1, 2)(3, 4), (1, 2, 3),
(1, 2, 3, 4), (1, 2, 4, 3), (1, 2, 4), (1, 3, 2),
(1, 3, 4, 2), (1, 3), (1, 3, 4), (1, 3)(2, 4), (1, 3, 2, 4), (1, 4, 3, 2), (1, 4, 2), (1, 4, 3),
(1, 4), (1, 4, 2, 3), (1, 4)(2, 3) ]
```

30. How do I create an alternating group of degree n with $\frac{n!}{2}$ elements?

```
gap> a4:=AlternatingGroup(4);
Alt([ 1 .. 4 ])
```

```
gap> Size(a4);
12
```

```
gap> Elements(a4);
[ (), (2, 3, 4), (2, 4, 3), (1, 2)(3, 4), (1, 2, 3), (1, 2, 4), (1, 3, 2), (1, 3, 4),
(1, 3)(2, 4), (1, 4, 2), (1, 4, 3), (1, 4)(2, 3) ]
```

31. How do I create a direct product of two or more groups?

```

gap> g1:=Group((1, 2, 3));
Group([ (1, 2, 3) ])

gap> g2:=Group((4, 5));
Group([ (4, 5) ])

gap> dp:=DirectProduct(g1, g2);
Group([ (1, 2, 3), (4, 5) ])

gap> Size(dp);
6
gap> Elements(dp);
[ (), (4, 5), (1, 2, 3), (1, 2, 3)(4, 5), (1, 3, 2), (1, 3, 2)(4, 5) ]

gap> ShowMultiplicationTable(dp);
*
| ()      (4, 5)      (1, 2, 3)      (1, 2, 3)(4, 5) (1, 3, 2)
-----
(1, 3, 2)(4, 5)
-----
()      | ()      (4, 5)      (1, 2, 3)      (1, 2, 3)(4, 5) (1, 3, 2)
(1, 3, 2)(4, 5) | (4, 5)      ()      (1, 2, 3)(4, 5) (1, 2, 3)      (1, 3, 2)(4, 5) (1, 3, 2)
(4, 5)          | (1, 2, 3)      (1, 2, 3)(4, 5) (1, 3, 2)      (1, 3, 2)(4, 5) ()      (4, 5)
(1, 2, 3)       | (1, 2, 3)(4, 5) (1, 3, 2)(4, 5) (1, 3, 2)(4, 5) (1, 3, 2)      (4, 5)      ()
(1, 2, 3)(4, 5) | (1, 3, 2)      (1, 3, 2)(4, 5) ()      (4, 5)      (1, 2, 3)
(1, 3, 2)       | (1, 2, 3)(4, 5) (1, 3, 2)      (4, 5)      ()      (1, 2, 3)(4, 5) (1, 2, 3)
(1, 3, 2)(4, 5) | (1, 3, 2)(4, 5) (1, 3, 2)      (4, 5)      ()      (1, 2, 3)(4, 5) (1, 2, 3)

```

32. How can I create the Quaternion group?

```

gap> a:=(1, 2, 5, 6)*(3, 8, 7, 4);
(1, 2, 5, 6)(3, 8, 7, 4)

gap> b:=(1, 4, 5, 8)*(2, 7, 6, 3);
(1, 4, 5, 8)(2, 7, 6, 3)

gap> q:=Group(a, b);
Group([ (1, 2, 5, 6)(3, 8, 7, 4), (1, 4, 5, 8)(2, 7, 6, 3) ])

gap> Size(q);
8

gap> IsAbelian(q);
false

gap> Elements(q);
[ (), (1, 2, 5, 6)(3, 8, 7, 4), (1, 3, 5, 7)(2, 4, 6, 8), (1, 4, 5, 8)(2, 7, 6, 3),
(1, 5, 3, 7)(2, 6, 4, 8), (1, 6, 5, 2)(3, 4, 7, 8),
(1, 7, 5, 3)(2, 8, 6, 4), (1, 8, 5, 4)(2, 3, 6, 7) ]

gap> q:=QuaternionGroup(IsPermGroup, 8);
Group([ (1, 5, 3, 7)(2, 8, 4, 6), (1, 2, 3, 4)(5, 6, 7, 8) ])

gap> Size(q);
8

gap> IsAbelian(q);
false

gap> Elements(q);
[ (), (1, 2, 3, 4)(5, 6, 7, 8), (1, 3)(2, 4)(5, 7)(6, 8), (1, 4, 3, 2)(5, 8, 7, 6),
(1, 5, 3, 7)(2, 8, 4, 6), (1, 6, 3, 8)(2, 5, 4, 7),
(1, 7, 3, 5)(2, 6, 4, 8), (1, 8, 3, 6)(2, 7, 4, 5) ]

```

33. *How can I find a set of independent generators for a group?*

```
gap> c6:=CyclicGroup(IsPermGroup, 6);  
Group([ (1, 2, 3, 4, 5, 6) ])
```

```
gap> Size(c6);  
6
```

```
gap> GeneratorsOfGroup(c6);  
[ (1, 2, 3, 4, 5, 6) ]
```

```
gap> d4:=DihedralGroup(IsPermGroup, 8);  
Group([ (1, 2, 3, 4), (2, 4) ])
```

```
gap> Size(d4);  
8
```

```
gap> GeneratorsOfGroup(d4);  
[ (1, 2, 3, 4), (2, 4) ]
```

```
gap> s5:=SymmetricGroup(5);  
Sym([ 1 .. 5 ])
```

```
gap> Size(s5);  
120
```

```
gap> GeneratorsOfGroup(s5);  
[ (1, 2, 3, 4, 5), (1, 2) ]
```

```
gap> a5:=AlternatingGroup(5);  
Alt([ 1 .. 5 ])
```

```
gap> Size(a5);  
60
```

```
gap> GeneratorsOfGroup(a5);  
[ (1, 2, 3, 4, 5), (3, 4, 5) ]
```

```
gap> q:=QuaternionGroup(IsPermGroup, 8);  
Group([ (1, 5, 3, 7)(2, 8, 4, 6), (1, 2, 3, 4)(5, 6, 7, 8) ])
```

```
gap> Size(q);  
8
```

```
gap> GeneratorsOfGroup(q);  
[ (1, 5, 3, 7)(2, 8, 4, 6), (1, 2, 3, 4)(5, 6, 7, 8) ]
```

34. *How do I find the conjugate of a permutation in the form $a^b = b^{-1}ab$?*

```
gap> a:=(1, 2, 3, 4, 5);  
(1, 2, 3, 4, 5)
```

```
gap> b:=(2, 4, 5);
(2, 4, 5)
```

```
gap> a^b;
(1, 4, 3, 5, 2)
```

```
gap> b^-1*a*b;
(1, 4, 3, 5, 2)
```

35. *How do I divide up a group into classes of elements that are conjugate to one another? (Note that “conjugacy” is an equivalence relation on our group G . That means that G can be separated into nonintersecting subsets that contain only elements that are conjugate to one another.)*

```
gap> d3:=DihedralGroup(IsPermGroup, 6);
Group([ (1, 2, 3), (2, 3) ])
```

```
gap> Size(d3);
6
```

```
gap> Elements(d3);
[ (), (2, 3), (1, 2), (1, 2, 3), (1, 3, 2), (1, 3) ]
```

```
gap> cc:=ConjugacyClasses(d3);
[ ()^G, (2, 3)^G, (1, 2, 3)^G ]
```

```
gap> Elements(cc[1]);
[ () ]
```

```
gap> Elements(cc[2]);
[ (2, 3), (1, 2), (1, 3) ]
```

```
gap> Elements(cc[3]);
[ (1, 2, 3), (1, 3, 2) ]
```

36. *How do I input a 3x3 matrix in GAP and display in its usual rectangular format?*

```
gap> x:=[[1, 2, 3], [4, 5, 6], [7, 8, 9]];
[ [ 1, 2, 3 ], [ 4, 5, 6 ], [ 7, 8, 9 ] ]
```

```
gap> PrintArray(x);
[ [ 1, 2, 3 ],
  [ 4, 5, 6 ],
  [ 7, 8, 9 ] ]
```

37. How do I do arithmetic with matrices?

```
gap> x:=[[1, 2],[3, 4]];
      [[ 1,  2 ], [ 3,  4 ] ]
```

```
gap> y:=[[5, 6],[7, 8]];
      [[ 5,  6 ], [ 7,  8 ] ]
```

```
gap> PrintArray(x+y);
      [[  6,   8 ],
       [ 10,  12 ] ]
```

```
gap> PrintArray(x-y);
      [[ -4,  -4 ],
       [ -4,  -4 ] ]
```

```
gap> PrintArray(x*y);
      [[ 19,  22 ],
       [ 43,  50 ] ]
```

38. How do I multiply a matrix by a number (scalar)?

```
gap> x:=[[1, 2],[3, 4]];
      [[ 1,  2 ], [ 3,  4 ] ]
```

```
gap> PrintArray(x);
      [[ 1,  2 ],
       [ 3,  4 ] ]
```

```
gap> PrintArray(2*x);
      [[ 2,  4 ],
       [ 6,  8 ] ]
```

```
gap> PrintArray(x/2);
      [[ 1/2,  1 ],
       [ 3/2,  2 ] ]
```

39. How do I find the inverse of a matrix?

```
gap> x:=[[1, 2],[3, 4]];
      [[ 1,  2 ], [ 3,  4 ] ]
```

```
gap> PrintArray(x);
      [[ 1,  2 ],
       [ 3,  4 ] ]
```

```
gap> xi nverse:=x^-1;
      [[ -2,  1 ], [ 3/2, -1/2 ] ]
```

```
gap> PrintArray(xi nverse);  
[ [ -2, 1 ],  
  [ 3/2, -1/2 ] ]
```

```
gap> xi nverse: =1/x;  
[ [ -2, 1 ], [ 3/2, -1/2 ] ]
```

```
gap> PrintArray(xi nverse);  
[ [ -2, 1 ],  
  [ 3/2, -1/2 ] ]
```

```
gap> PrintArray(x*xi nverse);  
[ [ 1, 0 ],  
  [ 0, 1 ] ]
```

40. How do I find the transpose of a matrix?

```
gap> x:=[[1, 2], [3, 4]];  
[ [ 1, 2 ], [ 3, 4 ] ]
```

```
gap> PrintArray(x);  
[ [ 1, 2 ],  
  [ 3, 4 ] ]
```

```
gap> xtranspose: =TransposedMat(x);  
[ [ 1, 3 ], [ 2, 4 ] ]
```

```
gap> PrintArray(xtranspose);  
[ [ 1, 3 ],  
  [ 2, 4 ] ]
```

41. How do I find the determinant of a matrix?

```
gap> x:=[[1, 2], [3, 4]];  
[ [ 1, 2 ], [ 3, 4 ] ]
```

```
gap> PrintArray(x);  
[ [ 1, 2 ],  
  [ 3, 4 ] ]
```

```
gap> DeterminantMat(x);  
-2
```

SUMMARY (PART 7)

We have covered some amazing stuff in part 7! In particular, we've learned about:

- *Matrices*
- *Permutation Matrices*
- *Matrix Groups*
 - *The General Linear Group, $GL_n(\mathbb{R})$*
 - *Upper Triangular Matrices*
 - *Lower Triangular Matrices*
 - *Reflection Matrices*
 - *Diagonal Matrices*
 - *Scalar Matrices*
 - *Rotation Matrices*
 - *Orthogonal Matrices, $O(n)$*
 - *Special Orthogonal Matrices, $SO(n)$*
 - *Unitary Matrices, $U(n)$*
 - *Special Unitary Matrices, $SU(3)$*
- Using GAP to study *matrices*.

PRACTICE (PART 7)

1. Find the sum, difference, and product of the following two matrices.

$$\begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix}, \begin{bmatrix} 2 & 10 \\ -3 & 0 \end{bmatrix}$$

2. Find the product of the scalar and matrix below.

$$3 \cdot \begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix}$$

3. Use GAP to find the determinant and inverse of each of the following matrices. Also, use `PrintArray` to express results in standard format for matrices.

$$\begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix}, \begin{bmatrix} 2 & 10 \\ -3 & 0 \end{bmatrix}$$

4. Use GAP to find the transpose of each matrix below. Also, use `PrintArray` to express results in standard format for matrices.

$$\begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix}, \begin{bmatrix} 2 & 10 \\ -3 & 0 \end{bmatrix}$$

5. Construct 5×5 matrices corresponding to the cycles below. Also, find the size of the permutation group generated by these cycles, and see if you can guess what group you've seen before that this group is isomorphic to.

$(1,2,3), (4,2,5)$

PRACTICE (PART 7) - ANSWERS

1. Find the sum, difference, and product of the following two matrices.

$$\begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix}, \begin{bmatrix} 2 & 10 \\ -3 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix} + \begin{bmatrix} 2 & 10 \\ -3 & 0 \end{bmatrix} = \begin{bmatrix} 10 & 5 \\ -1 & 6 \end{bmatrix}$$

$$\begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix} - \begin{bmatrix} 2 & 10 \\ -3 & 0 \end{bmatrix} = \begin{bmatrix} 6 & -15 \\ 5 & 6 \end{bmatrix}$$

$$\begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix} \cdot \begin{bmatrix} 2 & 10 \\ -3 & 0 \end{bmatrix} = \begin{bmatrix} 31 & 80 \\ -14 & 20 \end{bmatrix}$$

2. Find the product of the scalar and matrix below.

$$3 \cdot \begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix}$$

$$3 \cdot \begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix} = \begin{bmatrix} 24 & -15 \\ 6 & 18 \end{bmatrix}$$

3. Use GAP to find the determinant and inverse of each of the following matrices. Also, use PrintArray to express results in standard format for matrices.

$$\begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix}, \begin{bmatrix} 2 & 10 \\ -3 & 0 \end{bmatrix}$$

```
gap> x:=[[8, -5], [2, 6]];
[[ [ 8, -5 ], [ 2, 6 ] ]]
```

```
gap> PrintArray(x);
[[ [ 8, -5 ],
  [ 2, 6 ] ]]
```

```
gap> DeterminantMat(x);
58
```

```
gap> PrintArray(x^-1);
[[ [ 3/29, 5/58 ],
  [ -1/29, 4/29 ] ]]
```

```
gap> y:=[[2, 10], [-3, 0]];
[[ [ 2, 10 ], [ -3, 0 ] ]]
```

```
gap> PrintArray(y);
[[ [ 2, 10 ],
  [ -3, 0 ] ]]
```

```
gap> DeterminantMat(y);
30
```

```
gap> PrintArray(1/y);
[[ [ 0, -1/3 ],
  [ 1/10, 1/15 ] ]]
```

4. Use GAP to find the transpose of each matrix below. Also, use `PrintArray` to express results in standard format for matrices.

$$\begin{bmatrix} 8 & -5 \\ 2 & 6 \end{bmatrix}, \begin{bmatrix} 2 & 10 \\ -3 & 0 \end{bmatrix}$$

```
gap> x:=[[8, -5], [2, 6]];
[[ [ 8, -5 ], [ 2, 6 ] ]]
```

```
gap> PrintArray(TransposedMat(x));
[[ [ 8, 2 ],
  [ -5, 6 ] ]]
```

```
gap> y:=[[2, 10], [-3, 0]];
[[ [ 2, 10 ], [ -3, 0 ] ]]
```

```
gap> PrintArray(TransposedMat(y));
[[ [ 2, -3 ],
  [ 10, 0 ] ]]
```

5. Construct 5×5 matrices corresponding to the cycles below. Also, find the size of the permutation group generated by these cycles, and see if you can guess what group you've seen before that this group is isomorphic to.

$(1,2,3), (4,2,5)$

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

```
gap> a:=(1, 2, 3);
(1, 2, 3)
```

```
gap> b:=(4, 2, 5);
(2, 5, 4)
```

```
gap> g:=Group(a, b);
Group([ (1, 2, 3), (2, 5, 4) ])
```

```
gap> Size(g);
60
```

According to GAP, the order or size of the *group* generated by the two permutations is 60. Also, notice that $(1,2,3)$ and $(4,2,5)$ are both even permutations. Thus, the *group* generated by these permutations must be *isomorphic* to A_5 , the *alternating group of degree 5* which is essentially the *group* of all even permutations of five objects.



**GROUP THEORY REALLY
IS FOR EVERYONE!**