[] - Activity

[] - Demo

[] - Discussion Question

[] – Note or Optional Material

# Intro / Warm Up

* [Show final demo so they can see the end goal]
* High Level Overview
  + This complex shape is made of a bunch of spinning vectors, like these
  + When we add the vectors together, tip-to-tail, it draws out some shape over time.
  + By tweaking the initial *size* and *angle* of each vector, we can make it draw anything we want
* **Each component, an arrow spinning at constant speed, is super simple. But these simple pieces added together can make something that’s super complex.**

# Series

* The idea of building complex things out of simple pieces is called a **series** in math.
* [Turn to your neighbor and see if you can think of series in the real world]
  + Examples: Legos, Money, Paying with Cash
* [Walking Activity]
  + Problem: Walk from one end of the room to the other, but you have to rest after walking half the distance left to go.
  + Have students draw it out on a piece of paper or draw it on the white board
  + Have a student demonstrate if someone volunteers
  + Can you ever make it to the other side? How close is close enough?
  + Formula:

## Constant Series

* In the walking example, we really didn’t need a series since we know *exactly* how far we need to go (i.e. 1\*distance, we know exactly what 1 is and how to represent it).
* But what if we want a number like
  + is an **irrational number**. The decimal goes on forever and no one knows *exactly* what it is.
* We can use a series to describe pi:
* [show pi series demonstration]
  + The more terms we add, the closer we can get to the exact answer
  + Also, we understand exactly what each piece we’re adding is; we have a formula for each piece. But these pieces are adding up to something that we don’t know exactly.
* If we had an infinite number of terms to work with, then the series would *equal* pi exactly. But **Finite sums will always by approximations**. No matter how many terms we add, we won’t get exactly the right answer, but we can get as close as we want by adding more and more terms.

### Sound as a Series

* Like numbers are the simple building blocks for pi, pressure waves are the building blocks for sound.
* Sound is a series of waves that move from the source to your ear
  + [How sound works video]
    - Good video with creepy animations: <https://www.youtube.com/watch?v=eQEaiZ2j9oc>
    - Presenter with too much coffee: <http://youtube.com/watch?v=3-xKZKxXuu0>
    - Good quick animation: <https://www.youtube.com/watch?v=27a26e2CnuM>
    - Visualizing sound: <https://www.youtube.com/watch?v=px3oVGXr4mo>
* The speed of the wave determines the “pitch” of the sound – how high the note sounds to our ears.
  + [Sound generation while manipulating the frequency of the sound]
* Now things get interesting when we start adding these different sound waves together
  + [Manipulate with different sounds and sinusoids overlapping to form a chord]
  + [Students can manipulate “Sound Builder” demo to see how the different notes add up to make a chord]
* **The waves are the building blocks – or basis – for making the more complex sound**

# Complex Exponentials = Spinning Arrows

* Just like different sound waves can add together to create any sound, spinning vectors can add together to draw any continuous line.
  + Sound is 1-Dimensional (pressure vs time), while the spinning vectors are 2D (x,y vs time).
* In Math, we use something called a “Complex Exponential” to represent our spinning vectors.
  + Normally, is a function that curves up and goes to infinity.
  + But when we add the imaginary number to the exponential, it does something totally different! When plotted in the “Complex Plane,” it wraps around the center of the plot and makes a circle.
  + To draw a vector that is one unit long at an angle to the horizontal line:
    - [Plot for different angles]
  + To make the vector spin, replace with
    - ,
    - [Plot for different speeds]
  + If they ask why f(t) makes a circle, point them to Euler’s Formula: , which comes from the Taylor *series* expansion of .

# Control Knobs

* So far, our basic building blocks are unit vectors that spin at different speeds. How do we combine them to make them draw what we want?
* For each spinner, we control its *length* and *starting angle* by multiplying them with a coefficient:
  + [Demonstrate how multiplying the spinners with different coefficients work]
* We can add these spinners up to make a series, just like we did with numbers and waves
  + [Add random spinners to make cool spirographs]
* So if we have a goal drawing, how do we know where to turn our control knobs length and angle to make it do what we want?
  + [Try randomly tweaking the coefficients, show its really hard to do by hand]
  + [Let students try manually tuning the parameters with demo]
* How do we know where to turn our control knobs? Again, math has the answer:
* It looks super complicated, but this formula essentially takes an *average* of our input drawing times the spinner.
* We then get a for each spinner, multiply the spinner by its , and add all the spinners together!

# Final Demo

* Try to show each step of the process here and [let them experiment with it].

# Conclusion

* We can approximate almost anything if we have enough terms
* The beautiful thing is that we can take simple ideas, like spinning circles, and combine them to make things that are complex and beautiful.
* Really, this is how all of science and technology works
  + Our knowledge about how the world works is built out of a bunch of simple discoveries that have added up to the understanding we have today.
  + Our cars started with the discovery of the wheel…
  + Tie it back to how they can contribute to STEM by following their curiosity and contributing small things.

# Resources (If you’re curious)

## Websites

* <http://jezzamon.com/fourier/>
* <http://alex.miller.im/posts/fourier-series-spinning-circles-visualization/>

## Videos

* <http://youtube.com/watch?v=r6sGWTCMz2k>
* <https://www.youtube.com/watch?v=ds0cmAV-Yek&t=432s>
* <http://youtube.com/watch?v=spUNpyF58BY&t=397s>