

## Daily Log

### Monday October 28

I fixed some nasty merge conflicts and started working on added different types of graphs.

### Wednesday October 30

I added the ability to "graph" 2D vectors. Basically the current value is 100% opacity and then values after that drop off in opacity. At first I the opacity drop off linearly but eventually I decided to make it start with 100% opacity for the most recent vector and then immediately drop to 60% for the second and drop off linearly from there. I thought that being able to see the vectors would be more informative than it is. I thought of graphing the endpoint of the vector over time as a curve, but I decided that it wouldn't be very helpful since it's hard to tell direction and I'm not sure how I could indicate direction without a lot of work. I also optimized the graph drawing backend stuff a bit and made it easier to work with in general.

### Friday November 1

Honestly I skipped school to work on college apps. I did play with trying to improve the collisions again but didn't make any progress.

### Monday/Tuesday November 4/5

I ran a linter and made some changes. Recently compile times (4-5ish seconds on my laptop) have been a pain so I looked into ways to speed it up. I try and implement a few things like removing debugging symbols, but I didn't notice a significant difference unless compiling all the dependencies too. Maybe I should get a threadripper. I also tried to improve collisions again but still no improvement. I'll probably rewrite the collisions with all the stuff I've learned in mind.

### Wednesday November 6

I updated Rust and ran 'pacman -Syyu'. Next I had to completely recompile the project and all the dependencies which took probably 15 minutes. I've never had to compile everything all at once on my laptop before because the dependencies got added incrementally. Theoretically the new version of Rust compiles slightly faster though.

### Friday November 8

I added the ability to record graphs to .csv files. This is probably the first place where I regretted a design decision I made because right now no matter what value is recorded (i.e. speed, x velocity, rotational velocity etc.), the column will have to say the same value. The reason for this is that

the csv writing library I'm using serializes structs and names the columns the same that the field was originally named. Right now I'm mapping the vector of values to a struct that contains the time and the value and then serializing that struct, so I could create different structs for each type of data and just rename the fields, but that seems pretty silly. I could also implement a custom serialize method for the struct, but I'd have to store the type of data in that struct then, and it would have to be duplicated depending on how many rows the data is, wasting a lot of space. I also want to switch from a right click menu to a side panel because there's a lot of buttons now. I could add submenus, but I think submenus would work in a side panel better too. — While writing this I realized that I could implement the custom serialize and just skip the map and just use the `.for_each`, so now the labels are correct.

## Timeline

I took away the new page for this because I'm low on paper

Date	Goal	Met
Today minus 2 weeks	Complete UI and record graph data	Yes
Today minus 1 weeks	Finalize	I added a lot of nice features but I still don't think it's finalized. The biggest thing I have to add is FBDs, but since I've already added vector graphing it shouldn't be that hard.
Today	Simultaneous collisions	Nope, I'll probably try to rewrite it in my free time.
Today plus 1 week	Rewrite gravity sim from last year	I really want to take a break from rigidbody mechanics
Today plus 2 weeks	Finalize mechanics stuff more	FBDs, side panel, maybe more

## Reflection

The last two weeks I've had a lot less time even though there was a lot less school. Collisions haven't made any progress since last week but circles still work fine and most things are teachable with just circles, so I think it's time to make other things my primary focus. I'll rewrite the collision module when I have more time but I don't know when that will be. Vector graphs are pretty cool looking even if they're not as informative as I thought. The work isn't wasted though because it's basically the start of a FBD maker. Graphs are also significantly faster now which is cool. Compile times have been getting annoying because so much of the work is iterative. When I'm trying to get the math to work, it either works or doesn't (other than collisions), but for user side stuff like the GUI and recording graph data I need to tune it to make it easiest. Working with CSV files is pretty interesting because I thought they'd work basically the same as text files, but it's more like JSON. I want to take a break from mechanics and the gravity sim is probably the quickest one I have to do since most of it is already done, so I think I'll work on that next. I read through the code for it but it's from last year and kind of bad so I think I'll just rewrite it. It's also better if I rewrite it so that everything is standardized for the future, especially since the wave lab sim will probably be pretty similar.