PART

April 13, 2021

YEAH WE WERE THOSE GUYS

We made an actual website for a school project, like what, who does that???

Anyways, summary: This was quite a fun project, since we not only saw some cool graphics (looking at you, shaders), but we saw the more interactive side of graphics (animations and simulations, very very juicy). It was cool to see physics and math, two heavy backbones for the raytracing project, in action here, creating relatively realistic cloth movement (with collisions), and then further bringing it to life with shaders.

Also can we get an extra credit point for the website plz lol

A NOTE ON PARTNER COLLABORATION:

Like before, we collaborated by taking turns "piloting," where we discussed approaches, then coded one at a time. However, we didn't always have time to meet together, so sometimes after discussing approaches, one of us would be the ones to actually implement it, without the other person's supervision. We then came back if the other became stuck, trying to debug. For example, one of us noticed the other person had not grouped variables correctly in parenthesis. It went relatively well, and we've been working together for 4 projects so we had almost no trouble dealing with merge conflicts. We learned more about remote software development, debugging each other's code, and writing more understandable code.

A NOTE ON DEBUGGING:

Luckily since there was a lot of interactivity, bugs were a little easier to squash. For example, we had accidentally projected the normal onto a vector rather than the other way around for plane collision testing, which caused the cloth to take a

nosedive off the side, lul. Working with vectors did make it easier to see that our correction vector was bad, since with just checking if we had collided, the cloth stopped right above the plane, which was fine. We did have a few more C++ related bugs, since this project was a little more involved on that front. For example, for auto vs for auto& causing point masses to not update, since it was passing around a copy rather than the actual reference, but overall this wasn't as hard as raytracing to debug (dunno if anything is gonna top that, haha).