

Vision:

The goal of this project remains the same - to create a type system which forces correct transformations between 3D spaces (Such as World, Object, Model) and coordinate systems. Previously I described the idea of using a Tag and Function system. I've transitioned to a more tag focused approach. I am focused only on getting tags to work properly, without transformation between tags. An example would be that a vector tagged as "space, world" and "system, cartesian" can only be added with another vector with the same tags.

Summary of progress:

Previously I fleshed out the tag system idea in more detail and simplified the hw6 base code.

However I found that the lambda calculus implemented in hw6 made implementing my vision more difficult, so I started on a new AST, Type checker, lexer, and parser [based on the code provided in the ocaml textbook].

I then implemented a Vector2 structure (what is primarily used in 2d computer graphics). There were some difficult bugs with menhir and the Lexer/Parser, but they were resolved eventually.

I implemented a basic "TaggedExpression", which consists of a target expression and a string containing the tag information. It was a bit tricky to get the tag working within the lexer, but it currently parses as intended.

Activity breakdown:

I am working alone for this assignment.

Productivity analysis:

I am satisfied with the amount of work I got done this sprint. I feel that while refactoring the base code was time consuming it increased my understanding and will make further progress faster. I also met the main goals I had set for myself in the Alpha sprint, with Vector and basic Tag implementation.

However I wish I had gotten further into the type checking using tags, as it is very naive right now, and just returns the type contained in the tag, ignoring the string.

Grade:

I accomplished the goals for "Satisfactory" which I set for myself last sprint. However I would rate myself as "Good" based on the extra time spent setting up the new lexer and parser. I think I am well on track to have a functional project after the final sprint.

Goals for next phase:

Satisfactory:

- Make the type checker less naive (match each tag individually)
- Expand the test suite to include type checks

Good:

- Implement a more robust Tag structure, of the form of a string tuple list

Excellent:

- Print further details on test suite failures
- Add a tag transformation function