Specification (what do you think the purpose of this milestone is):

I see the main purpose of this milestone to be finally getting us outputting some actual gforth code with our "compiler". It enforces the use of a post order traversal in order to translate the prefix IBTL to postfix Gforth and gives us a bit more experience working with Gforth after Milestone 1. We were assigned only constant operations with this milestone in order to ease the transition of making these switches.

Processing (how did you go about solving the problem):

To me, this milestone had much less of a planning and design compared to the previous ones. I knew that the biggest thing was switching from prefix to postfix and then just adding some minor changes for the gforth semantics that differ from the IBTL. I was presented with the idea of waiting until the very end of the parser to traverse the tree to generate the code, or to do the traversal and code generation as the parser went along. I ended up doing the code generation as the parser was going along and designed my code generator to print out terminal constants whenever they were found and to print out terminal operators or expressions on the way back up the recursive descent.

Testing Requirement (how did you test for correctness):

A lot of testing was done along the way. As soon as the code for a certain production was finished I would run small tests that only included pieces of code that could be generated by the productions I had just coded. Once my code generator was complete I came up with quite a few test input cases that I thought should pass. Through these tests I found some small errors in my code that needed to be fixed, so I worked on fixing all the bugs that I found until all the tests I had properly passed.

Retrospective (what did you learn in this milestone):

I got some more practice with tree traversal, specifically doing postorder in order to properly output gforth code. I also learned more about gforth semantics that helped refresh what I learned doing milestone 1.