

Samuel Leonard

CS461

PA3: Cexpr using Lex and YACC

1. The program I set out to write was to implement a calculator using the lex and yacc programming languages.
2. To implement this, I had to come up with a grammar with regular expressions. I tried to create a parse tree with some of the given inputs to try to create the productions. I also had to check left recursion and try to eliminate it.
3. I used the input file to help debug my program even though I was not fully able to finish the program. I also used several websites with information on lex and yacc to try to better understand how they work.
4. I was not completely able to finish the program because of several complications. First, I spent a lot of time trying to figure out how to get the variables to be recognized by the grammar in yacc. Another issue was that a value keeps printing out after variables are assigned. I spent a lot of time trying to get lex and yacc working properly that I did not get to the dump, overflow and two symbol tokens like ">=" etc. My plus, minus, divide, multiply, modulus and parenthesis work for the most part and they can be multinomial but the assignment token '=' works to a degree but when two variables are to be set equal to a third, all three do not represent the right number. So, in conclusion, I had several issues with this program and will seek more help on the constraints and how to implement them. My calculator turned out to be a more basic calculator than what was intended.