1. Problem Statement:

The problem is to create a program that can analyze the lexical of a source code then generates the tokens and writes the result to an output file

1. How to use the program:

When first load up our program, you will be asked to input a number for your selection of inputting the code to be analyze. It can be from hard input the code or input the name of the file containing the code. If the user hard input the code, then the user doesn’t need to skip a line but can just write all the code as a single line. After choosing which option is best for analyzing the code, the user can just press Enter and the program will analyze the code.

1. Design of the program:

The program is built base on the use of Linked List. The Linked List can store 2 data at the same time. One for the token and one for the word corresponding to the token after going through the lexical analyzer function. The program is designed to be able to accept the code that the user input from the keyboard or from an input file. The program takes 1 line and starts analyzing it one char at a time. First, the char will be input to the testWord variable. If the variable is a block comment (‘!’), the integer variable blockComment + 1, and the program will start ignoring all other char until it finds another block comment. Second, if the testWord is a space, an operator, a separator then the string that contains everything before it will be put into the lexical analyzer function(lexer()). When the string got put into lexer(), it will keep going and compare if it is an operator, a keyword or a separator, but if its none of those then it’s an identifier. However, the string will go into one more checking to see if it is a number or not. If all the character in the string is a digit, it is an integer. If there’s a dot then it is a real number.

1. Limitation: None
2. Shortcoming: None