**GITHUB:**

**102008 ESTHER WAMBUI NJOROGE ICS 4A**

**For the C++ Code**

***In this section, I will briefly explain the code while giving example of what to expect.***

The parser breaks down phrase into token

Operators:

\\+|\\-|\\\*|\\>|

Key word

If, char, case, do, while,

Identifiers

[a-z]

Separators

\\;|\\(|\\)|\\{|\\}

Numbers

[0-9]

Suppose I insert the phrase: **a+ b=3,** the analyzer is to break the tokens down into their respective types.

**a - Identifier**

**+ - operator**

**b - Identifier**

**= - operator**

**3 – number**

***For the Flex folder:***

The symbols.h file was created for the parser to recognize tokens such as; colons, numbers and identifiers

In the lex.l file:

Keywords:

db\_type, db\_name

Characters

[a-z A-Z]

Integers

[1-9] [0-9] \*

White space is set to be ignored.

An error shows up if the input doesn’t match what has been specified.

***On the C file***

External declarations are defined.

On writing the parse,

An example of the appropriate syntax would be;

**:a2**

If the symbols are place differently, it would be considered to be a syntax error.