**Project Proposal**

This project aims to introduce lexical analysis for the complex functions of a programming language. The lexical analysis is the phase of compiler development where the grammar for the language is developed along with tokens and parsing rules. Following are my ideas for this project:

* Analysis of complex statements of C++
* The parsing of the advanced functions
* Keywords, Parsing and detailed tokenization
* The identifiers for the C++ advanced lines
* Exceptional work other than simple if-else, loop, etc.

Following are some examples of my grammar:

<condition> -> condition<cond><expression\_list>

<cond> -> (<fact> = (.|&|\*))

<loop> -> loop<cond><expression\_list>

<cond> -> (<fact> = (integer|&++)

<expression\_list> -> { <product> | <addition> | <division>}

<addition> -> (<fact>{,<fact>})

In this way, the project aims to develop parser and analyzer for complex statements. I can also try other things as per my understanding during the implementation but an idea as whole is to work and understand the compiler construction.