**101193 Cosmas Nyairo ICS 4B**

**Description of Lab 1**

**Link for Lab 1:** [**https://github.com/cosmasnyairo/101193\_CCProjects/tree/main/lab1**](https://github.com/cosmasnyairo/101193_CCProjects/tree/main/lab1)

**Preview of lab 1 :** [**https://github.com/cosmasnyairo/101193\_CCProjects/blob/main/readMe.md**](https://github.com/cosmasnyairo/101193_CCProjects/blob/main/readMe.md)

For our simple lexical analyzer broke down our phrase into different tokens:

* Keywords

if|else|do|while|switch|case|break|continue|default|for|auto|const|int|float|double|string|void|char|char\*|bool|return|cin|cout|endl|main"

* Identifiers

[a-z]+

* Operators

"\\+|\\-|\\\*|\\/|\\>|\\<|\\=|\\++|\\==|\\--|\\&&|\\!=|\\+="

* Separators

"\\;|\\(|\\)|\\{|\\}|\\<|\\>|\\[|\\]|\\&|\\:"

* Numbers

[0-9]+

The input phrase is passed through the program which breaks down it into tokens and matches it with the role it qualifies

For the phrase, c=b\*2; our program would break it down and return the following output:

|  |  |
| --- | --- |
| **Token** | **Type** |
| c | identifier |
| = | operator |
| b | identifier |
| \* | operator |
| 2 | Number |