



1- Identifiers:

- ## 2- Integer constants

- ### 3- Operators

- #### 4- Brackets

- ## 5- String constants

- 6- Keywords:

- Keywords are:  
break, case, char, const, continue, do, else, enum, float, for, goto, if, int, long, record, return, static, while
- Ceng++ language is not case sensitive and all the keywords are standardized as lower case. You can write the same word as "while" OR "While" OR "WHILE" and they all generate the
- Example Token: Keyword(while)

7- End of line: ;

- Example Token: EndOfLine

8- Comments: Anything between (\* and \*) is a comment.

- If a comment cannot terminate before the file end, there should be a lexical error issued.
- Comments are just like blank space and they provide no tokens.

-----

**Project Definition:** The Program should accept a source file called `code.Ceng` and produce a text file named as `code.lex` that contains all the tokens of the `code.lex` listed one after the other.

Example:

if code.Ceng contains:

```
var:=var_1+18;(*additonn*)
```

```
var_1++; (*increment*)
```

code.lex would be:

```
Identifier(var)
```

```
Operator(:=)
```

```
Identifier(var_1)
```

```
Operator(+)
```

```
IntConst(18)
```

```
EndOfLine
```

```
Identifier(var_1)
```

```
Operator(++)
```

```
EndOfLine
```

-----