# Compiler-2018 Project1

## 1. Ability of scanner

**The scanner will parse the input file**

- Tokens passed to the parser
    - Delimiters

      `,` `;` `()` `[]` `{}`
    - Arithmetic, Relational, and Logical Operators

      `+` `-` `*` `/` `%` `=` `<` `<=` `!=` `>=` `>` `==` `&&` `||` `!`
    - Keywords

      `while` `do` `if` `else` `true` `false` `for` `int` `print` `const` `read` `boolean` `bool` `void` `float` `double` `string` `continue` `break` `return`
    - Identifiers

      `string of letters and digits beginning with a letter`
    - Integer, Float, Scientific
    - Strings
- Tokens discarded
    - white space, tabs, newlines
    - comments

      `// c++style`

      `/* c-style */`

**scanner will print tokens, lines, and frequency of the occurrence**

**of each identifier, based on Source, Token, Statistic options**

*use pragma directive*

- `#pragma source on/off`:
  - listen and print source code / stop listening
- `#pragma token on/off`:
  - listen and print each token type / stop listening
- `#pragma statistic on/off`:
  - print the frequency of the occurrence of each idenifier / not printing freqency

**Error message should be printed**

When an error occurs, print error message:

```
fprint(stderr, "Error at line %d: %s\n", linenum, yytext);
exit(1);
```

# 2. Platform to run scanner

Use **lex** to implement scanner

build and execute in Linux/Unix system

*Take Ubuntu as example*

- Install Flex/Lex

```
% sudo apt-get install flex
```

# 3. How to run my code?

- To run my scanner, type

```
% make
% ./scanner [inputfile]
```