CS 3323 - Principles of Programming Languages
Assignment 1

Download the files pl-scanner.yy, driver.c, tokens.h and Makefile from the assignment directory space. Then, perform the necessary modifications to the rule's file to recognize the below strings and tokens. The token codes can be found in the tokens.h header file.

The assignment is due on February 10th, 2020, 11:59pm. Files must be uploaded by then. Late policy deduction applies.

1. (1pt) Add the necessary rules to recognize the arithmetic operators: $+, -, *, /, <=, >=, ==, ! =, <, >$. See the file tokens.h to determine the constants to be used.

2. (2pt) Modify the rule returning the T_ID token to recognize all identifiers matching the following:

   - identifiers can start with an underscore ('_') or a letter, both lower or uppercase
   - all identifiers must have a minimum length of 2 characters
   - all identifiers must have at least one letter
   - identifiers can consist of letters, the underscore character or digits

   Examples of valid identifiers are: it, ii, a1, _a, b_, b1, _counter_, _a_b_c_2_, __a3. Examples of invalid identifiers are: __, _2, a, 2_.

3. (1pt) Create a new rule to recognize floating point numbers. The rule should return the token T_FLOAT:

   - Can start with a digit, the '+' or a '-'
   - Both the integer and fractional part should consist of at least one digit
   - The integer and fractional part should be separated by a '.'

   Examples of acceptable floating point numbers are: ”10.0”, ”+1.5”, ”-10.50000”, ”0.9”. Examples of string that should be rejected are: ”0.”, ”.01”

4. (1pt) Create a rule to recognize the following keywords: integer, float, foreach, begin, end, repeat, until, while, declare, if, then.

For convenience, a Makefile is provided, but you are not required to use it. Run:

```
make
```

to rebuild the scanner generator (lex.yy.c), and to recompile the driver.

Several online resources can be found in the web, for instance:

- `http://alumni.cs.ucr.edu/~lgao/teaching/flex.html`,
- `http://web.mit.edu/gnu/doc/html/flex_1.html`,
- `https://westes.github.io/flex/manual/`.

More resources can be found by searching for the key terms: C scanner generator.

Do not change the driver file nor the tokens.h file since the actual integer values will be used for grading. Do not print anything to the output.

Every student should **upload a single file named: ABCDEFGH.yy**, where ABCDEFGH is the 8-digit code identifying the student (not the 4+4).