Assignment 1

- Design regular expressions for the following tokens:
  - arithmetic const: any sequence of digits, that possibly begins with "-"
  - arithmetic comparison: <, >, <=, >=, =
  - arithmetic operations: +, -, *, "div", "mod"
  - function definition keyword: "define-fun"
  - predefined functions "get-int" and "get-bool"
  - boolean constants "true" and "false"
  - boolean operators: "and", "or", "not"
  - conditional operator: "if"
  - local variable declaration: "let"
  - variable/function types: "int", "bool"
  - program entry point: "print"
  - variable/function name: any sequence of digits and Latin letters which cannot begin with a digit
  - parentheses

- Lexemes should be separated: by whitespaces, tabs, line breaks, or if concatenated with parentheses (e.g., string ”(not a)” gives four lexemes)
- Comments begin with “;” and end with the line breaks: they should be ignored

Assignment 1 (cont.)

Implement a scanner (recognizer of a token stream) in lex:

- For each input file that has a sequence of lexemes, identify their tokens and line numbers
- Make sure keywords are recognized first and not confused with variables/functions
- Ignore the comments
- In the case no token can be recognized, abort the scanner and print error message
- For each successfully recognized token stream, print the following:

```
line <#>: <token description>:     <lexeme>
```

Example:

- Input file:

```
(print
  (+ 5 bla))
```

- Output:

```
line 1: left parenthesis:      (
line 1: program entry:         print
line 2: left parenthesis:      (
line 2: arithmetic operation:  +
line 2: integer constant:      5
line 2: variable/function:     bla
line 2: right parenthesis:     )
line 2: right parenthesis:     )
```

Important

- Your code should be committed to your GitHub repository
- Commit your test cases (i.e., small programs with successfully recognized token streams or detected errors) and compilation/running steps
- All group members should commit
- When submitting to Canvas, write a short personal description of how each of you contributed to group work
- Late submission policy:
  - 10% off if submitted with a delay of up to 24 hours
  - 20% off if submitted with a delay between 24 and 48 hours
  - 0 points otherwise.