

## Project 1: Scanner and Parser

### COP4020: Programming Languages

Trent Wells

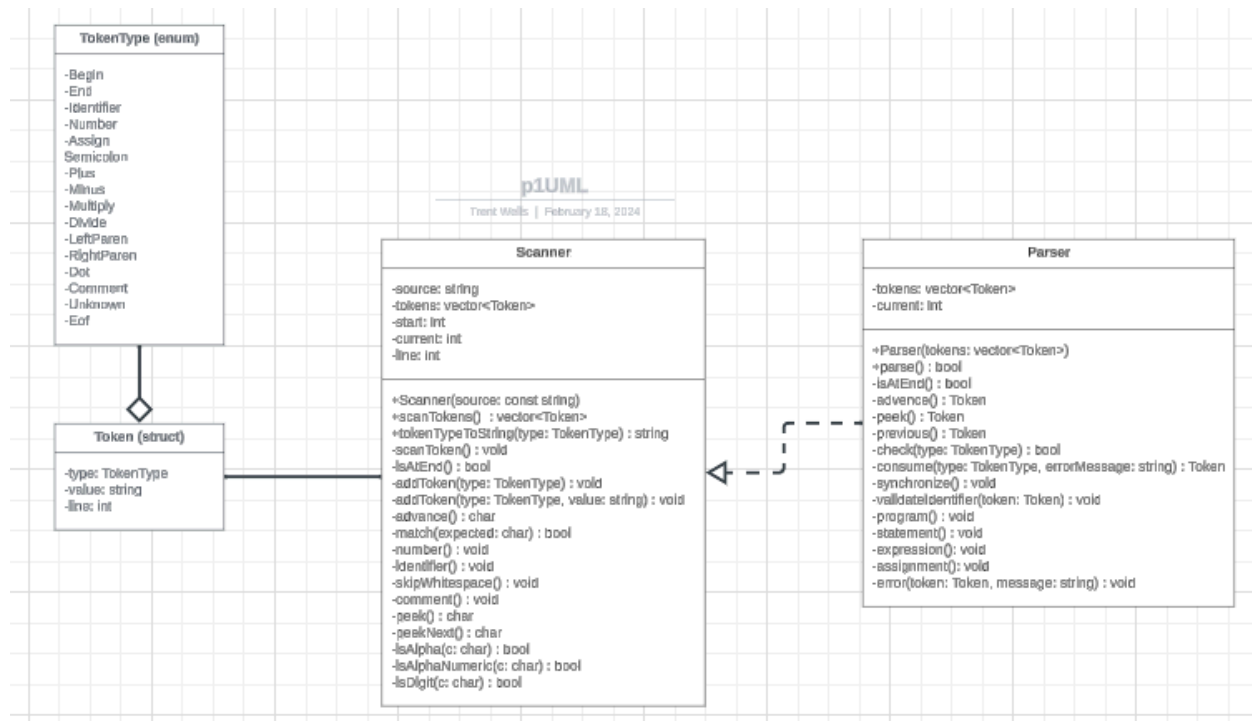
#### Lexeme Structure (found in p1Lexemes.xlsx):

	A	B	C	D	E
1	Lexeme description (RE or verbal)	Examples	Meaning	Token	Attribute 1
2	Begin	begin	beginning of code	Begin	N/A
3	End	end	end of code	End	N/A
4	[a-z], [A-Z]	a, b_xy	identifier	Identifier	name
5	\d+	99, 25	numeric literal	Number	numeric value
6	=	=	assignment operator	Assign	N/A
7	;	;	semicolon	Semicolon	N/A
8	+	+	addition operator	Plus	N/A
9	-	-	subtraction operator	Minus	N/A
10	*	*	multiple operator	Multiply	N/A
11	/	/	divide operator	Divide	N/A
12	(	(	left parenthesis	LeftParen	N/A
13	)	)	right parenthesis	RightParen	N/A
14	.	.	defines end of code along with "end"	Dot	N/A
15	~	~	comment denotation	Comment	N/A
16	N/A	N/A	unknown token	Unknown	N/A
17	eof	eof	end of file	Eof	N/A
18					

#### EBNF (found in p1Lexemes.xlsx):

	A	B
1	<u>Nonterminal</u>	<u>RHS</u>
2	program	"begin" {assignment} "end" " "
3	assignment	Identifier "=" expression ","
4	expression	Term {"+"   "-"} term
5	term	Factor {"*"   "/" } factor
6	factor	Identifier   number   "(" expression ")"
7	identifier	letter {letter   digit} [letter   digit]
8	letter	"a"   "b"   ...   "z"   "A"   "B"   ...   "Z"
9	digit	"0"   "1"   ..   "9"
10	number	digit {digit}
11	comment	"~" {all_characters} newline
12	newline	newline char
13	all_characters	any character

## UML (found in p1UML.png):



## Running:

To run this program in Linux follow these steps-

1. Open a terminal and navigate to the directory holding the project
2. Run "make" in the terminal
3. Use the command "./main all" to run all files (a1-a8)
4. Use the command "./main inputFilesP1/a1" to run a specific file (to run a specific file, a directory must be specified)

## Error Messages:

1. "Expected 'begin' at the start of the program" - This happens when a 'begin' token isn't detected at the start of the code.
2. "Expected 'end' after statements" - This happens when a 'end' token isn't detected at the end of the code.
3. "Expected '.' after 'end' - This happens when '.' isn't found directly after the 'end' token.
4. "Expected statement" - This happens when a statement is expected.
5. "Expected identifier" - This happens when an identifier is expected.
6. "Expected '=' after identifier" - This happens when an '=' isn't found after an identifier.

7. "Expected ';' after expression" - This happens when a ';' isn't found after an expression.
8. "Identifier cannot end with an underscore" - This happens when an identifier ends with an underscore.
9. "Identifier cannot contain consecutive underscores" - This happens when an identifier contains consecutive underscores.
10. "Unmatched parenthesis in expression" - This happens when an opening parenthesis doesn't have a matching closing parenthesis.