HW1

F74046365 李昕樺

執行:

lex compiler_hw1.l

cc lex.yy.c –ll –std=c99

./a.out <input file>

Basic Feature

1. Print the recognized token

(1) +	Add
(2) -	Sub
(3) *	Multiply
(4) /	Divide
(5) %	Remainder
(6) ^	Exponent
(7) =	Assign
(8) !	Not
(9) ,	Comma
(10) ;	Semicolon
(11) (	LB
(12))	RB
(13) {	LCB
(14) }	RCB
(15) “	Quotation
(16) If	IF FINCTION
(17) Else	ELSE FUNCTION
(18) For	FOR FUNCTION
(19) While	WHILE FUNCTION
(20) Print	PRINT FUNCTION
(21) Println	PRINTLN FUNCTION
(22) ++	Increment
(23) --	Decrement
(24) &&	And_Operator
(25) ||	Or_Operator
(26) <,>	Relational
(27) <=,>=,==,!=	Relational
(28) +=,-=,*=,/=,%=	Assignment

(29) number	Number
(30) number.number	FloatNumber
(31) “this is a string”	this is a string STRING
(32) //	C++ COMMENT
(33) /*……*/	C COMMENT

2. Count the lines of code in the given program
3. Implement the symbol table functions.
   - create_symbol() : Print “Create a symbol table”, when scanning the first id token
   - insert_symbol() : 當 id 不在 symbol table 裡面時，print ”Insert a symbol: [ID]”
   - lookup_symbol() :找是否有相同名稱的 id，並回傳 index
   - dump_symbol() :print 整個 symbol table，包括其 Index、name、type

Advanced Feature

1. Discard C and C++ type comment.

//this is a comment	C++ comment
/*this is a comment*/	C comment

2. Count the comment lines.

```
/* Hello World */
        C COMMENT
// Hello World
         C++ COMMENT
/*
 * Hello World
 */
        C COMMENT

Parse over, the line number is 6
comment: 5 lines
```

3. Undeclared variables
   當一個 id 沒有宣告時，他的 type 會被定為 undefined
4. Redefined variables
   當一個 variable 已經被宣告過，又再次被宣告時，會被記錄為 redefined，且不會再次被加入 symbol table

Code:

```
1 var x int
2 var y int = 5
3 int x
4 z
```

執行結果:

```
Create a symbol table
x        int TYPE VAR
Insert a symbol: x
y        int TYPE VAR
Insert a symbol: y
=        Assign
5        Number
x        Redefined
z        Undefined
Insert a symbol: z

Parse over, the line number is 4
comment: 0 lines

!!!Dump symbol table!!!
Index   Name    Type
0       x       int TYPE VAR
1       y       int TYPE VAR
2       z       Undefined
```