Compiler-2018 Project3

1. Changes to my previous scanner

- Add `#pragma symbol on/off` to lex.l
- Add `yylval` in *lex.l* to get return attribute information
- Add `%union` in yacc.y to define yylval type
- Associating union member names in yacc.y:
    - With terminals: `%token <union_member_name> TERMINAL`
    - With nonterminals: `%type <union_member_name> NONTERMINAL`
- Add actions (using functions defined in SymbolTable.c, SymbolTable.h)

2. Ability of parser

The parser will check syntax correctness and output syntax correctness.
Output symbol table information

- Name: The name of the symbol.
    - Each symbol have the length between 1 to 32.
- Kind: The name type of the symbol. There are four kinds of symbols:
    - function
    - parameter

    - variable
    - constant.
- Level: The scope level of the symbol.
    - 0 represents global scope, local scope starts from 1, 2, 3, …
- Type: The type of the symbol.
    - Each symbol is of types int, float, double, bool, string, or the signature of an array.
    - (Note that this field can be used for the return type of a function)
- Attribute: Other attributes of the symbol.
    - Such as the value of a constant, list of the types of the formal parameters of a function, etc.

The output format will be like:

```
printf("==========================================================\n");
// Name [29 blanks] Kind [7 blanks] Level [7 blank] Type [15 blanks] Attribute [15 blanks]
printf("Name Kind Level Type Attribute \n");
printf("----------------------------------------------------------\n");
printf{"a function 0(global) int int[2],float\n");
// ....
// Format of Attribute: type,type,type,...
printf("==========================================================\n");
```

3. Platform to run scanner/parser

Use **lex** and **Yacc** to implement scanner/parser

build and execute in Linux/Unix system

Take Ubuntu as example

- Install Flex/Lex and Bison/Yacc

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

3. How to run my code?

- To run my scanner/parser, type

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

- To delete files except `lex.l` `yacc.y` `Makefile` `SymbolTable.c` `SymbolTable.h`, type

```
% make clean
```