

1. lex 版本: flex 2.6.4
2. 作業平台: ubuntu 22.04
3. 執行方式: ./a.out < testfile.pas
- 4.

(1) 保留字是 case-sensitive，可以建一個這個表:

a	[Aa]
b	[Bb]
c	[Cc]
d	[Dd]
e	[Ee]
f	[Ff]
g	[Gg]
h	[Hh]
i	[Ii]
j	[Jj]
k	[Kk]
l	[Ll]
m	[Mm]
n	[Nn]
o	[Oo]
p	[Pp]
q	[Qq]
r	[Rr]
s	[Ss]
t	[Tt]
u	[Uu]
v	[Vv]
w	[Ww]
x	[Xx]
y	[Yy]
z	[Zz]

往後遇到字母都可以用{a}等方式同時表示大小寫。

(2) 識別字最長到 15 字元，可以簡單用 if-else 處理。

(3) 字串 30 過長的部分也可以用 if-else 處理，另外像: 'You'll see' 可以像這樣寫個小段程式來去掉多餘的單引號:

```
int i;
for(i=1; i<=yytext[i]; i++){
    if(yytext[i]=='\ ' && yytext[i+1]=='\ ' && i+1!=yytext[i]){
        printf("%c", yytext[i]);
        i++;
        continue;
    }
    printf("%c", yytext[i]);
}
```

(4) (*ab***)我還是沒辦法很完美的解決，如果(*ab***)出現之前有一個合法的註解，就會沒辦法正確辨認。目前我的 `comment` 和 `wrong comment` 的 `regular expression` 是這樣的：

```
wrong_string  (\'([^\r\n;]|\\")+)|{letter}+\'
comment  \\/[^\r\n]*|\"{\"[^\"]*\"}|\"([^\"]|\\\"+)[^\"]*\"
```

5. 遇到的問題：

我覺得最難的部分是讓 `scanner` 除了辨認正確的 `token`，還能辨認錯誤的 `token`，因為不合法 `token` 的可能性很多。我覺得這要對 `pascal` 有一定的熟悉，且思考周到才有辦法處理大部分錯誤。

第二個讓我產生問題的東西是 `regular expression`，延續第一個問題，由於有太多可能性要考慮，且又要將想法轉換成 `regular expression` 來表示，這個部分造成了我最多問題與停頓。`Regular expression` 需要不少時間熟悉，才能用的比較靈活。

第三個我覺得難的部分是 `lex`，`lex` 有許多功能，因此我覺得跟 `regular expression` 一樣，熟悉其功能及原理需要一些時間，在寫作業時，有時候也會因為不熟悉 `lex` 的規則而產生不必要的錯誤。

6. 執行結果：

-----Scanner start-----

```
Line: 1, starts at char. position: 1, program      "reserved word"
Line: 1, starts at char. position: 9, test        "identifier"
Line: 1, starts at char. position: 13, ;          "punctuation symbol"
Line: 2, starts at char. position: 1, var          "reserved word"
Line: 3, starts at char. position: 3, i            "identifier"
Line: 3, starts at char. position: 5, :            "punctuation symbol"
Line: 3, starts at char. position: 7, integer      "reserved word"
Line: 3, starts at char. position: 14, ;           "punctuation symbol"
Line: 4, starts at char. position: 1, begin        "reserved word"
Line: 5, starts at char. position: 3, read         "reserved word"
Line: 5, starts at char. position: 7, (            "punctuation symbol"
Line: 5, starts at char. position: 8, i            "identifier"
Line: 5, starts at char. position: 9, )            "punctuation symbol"
Line: 5, starts at char. position: 10, ;           "punctuation symbol"
Line: 6, starts at char. position: 1, end          "reserved word"
Line: 6, starts at char. position: 4, ;           "punctuation symbol"
```

Identifier Table:

test
i

-----Scanner end-----

-----Scanner start-----

```
Line: 1, starts at char. position: 1, program      "reserved word"
Line: 1, starts at char. position: 9, test        "identifier"
Line: 1, starts at char. position: 13, ;          "punctuation symbol"
Line: 2, starts at char. position: 1, var          "reserved word"
Line: 3, starts at char. position: 3, 3i          "wrong identifier"
Line: 3, starts at char. position: 6, :            "punctuation symbol"
Line: 3, starts at char. position: 8, string       "reserved word"
Line: 3, starts at char. position: 14, ;           "punctuation symbol"
Line: 4, starts at char. position: 1, begin        "reserved word"
Line: 5, starts at char. position: 3, 3i          "wrong identifier"
Line: 5, starts at char. position: 6, :=          "operator"
Line: 5, starts at char. position: 9, 'ab         "wrong character string"
Line: 5, starts at char. position: 12, ;           "punctuation symbol"
Line: 6, starts at char. position: 1, end          "reserved word"
Line: 6, starts at char. position: 4, ;           "punctuation symbol"
```

Identifier Table:

test

-----Scanner end-----

-----Scanner start-----

```
Line: 1, starts at char. position: 1,
(* comment 1
   comment 2 *)
"comment"
Line: 3, starts at char. position: 1, program      "reserved word"
Line: 3, starts at char. position: 9, test        "identifier"
Line: 3, starts at char. position: 13, ;          "punctuation symbol"
Line: 4, starts at char. position: 1, var          "reserved word"
Line: 5, starts at char. position: 3, i            "identifier"
Line: 5, starts at char. position: 5, :            "punctuation symbol"
Line: 5, starts at char. position: 7, integer      "reserved word"
Line: 5, starts at char. position: 14, ;           "punctuation symbol"
Line: 6, starts at char. position: 1, begin        "reserved word"
Line: 7, starts at char. position: 3, read         "reserved word"
Line: 7, starts at char. position: 7, (            "punctuation symbol"
Line: 7, starts at char. position: 8, i            "identifier"
Line: 7, starts at char. position: 9, )            "punctuation symbol"
Line: 7, starts at char. position: 10, ;           "punctuation symbol"
Line: 8, starts at char. position: 1, end          "reserved word"
Line: 8, starts at char. position: 4, ;           "punctuation symbol"
```

Identifier Table:

test
i

-----Scanner end-----

```

-----Scanner start-----
Line: 1, starts at char. position: 1, program "reserved word"
Line: 1, starts at char. position: 9, test "identifier"
Line: 1, starts at char. position: 13, ; "punctuation symbol"
Line: 2, starts at char. position: 1, var "reserved word"
Line: 3, starts at char. position: 3, f "identifier"
Line: 3, starts at char. position: 5, : "punctuation symbol"
Line: 3, starts at char. position: 7, float "reserved word"
Line: 3, starts at char. position: 12, ; "punctuation symbol"
Line: 4, starts at char. position: 1, begin "reserved word"
Line: 5, starts at char. position: 3, f "identifier"
Line: 5, starts at char. position: 5, := "operator"
Line: 5, starts at char. position: 8, 12.25e+6 "float"
Line: 5, starts at char. position: 16, ; "punctuation symbol"
Line: 6, starts at char. position: 1, end "reserved word"
Line: 6, starts at char. position: 4, ; "punctuation symbol"

Identifier Table:
test
f
-----Scanner end-----

-----Scanner start-----
Line: 1, starts at char. position: 1,
(* a**b)*)
"comment"
Line: 2, starts at char. position: 1, program "reserved word"
Line: 2, starts at char. position: 9, test "identifier"
Line: 2, starts at char. position: 13, ; "punctuation symbol"
Line: 3, starts at char. position: 1, var "reserved word"
Line: 4, starts at char. position: 3, i "identifier"
Line: 4, starts at char. position: 5, : "punctuation symbol"
Line: 4, starts at char. position: 7, integer "reserved word"
Line: 4, starts at char. position: 14, ; "punctuation symbol"
Line: 5, starts at char. position: 3, _s "identifier"
Line: 5, starts at char. position: 5, , "punctuation symbol"
Line: 5, starts at char. position: 7, _s2 "identifier"
Line: 5, starts at char. position: 10, , "punctuation symbol"
Line: 5, starts at char. position: 12, _s3 "identifier"
Line: 5, starts at char. position: 15, , "punctuation symbol"
Line: 5, starts at char. position: 17, _s4 "identifier"
Line: 5, starts at char. position: 20, , "punctuation symbol"
Line: 5, starts at char. position: 22, _s5 "identifier"
Line: 5, starts at char. position: 26, : "punctuation symbol"
Line: 5, starts at char. position: 28, string "reserved word"
Line: 5, starts at char. position: 34, ; "punctuation symbol"
Line: 6, starts at char. position: 1, begin "reserved word"
Line: 7, starts at char. position: 3, i "identifier"
Line: 7, starts at char. position: 5, := "operator"
Line: 7, starts at char. position: 8, -100 "integer"
Line: 7, starts at char. position: 12, ; "punctuation symbol"
Line: 8, starts at char. position: 3, _s "identifier"
Line: 8, starts at char. position: 6, := "operator"
Line: 8, starts at char. position: 9, 'db lab' "character string"
Line: 8, starts at char. position: 17, ; "punctuation symbol"
Line: 9, starts at char. position: 3, _s2 "identifier"
Line: 9, starts at char. position: 7, := "operator"
Line: 9, starts at char. position: 10, 'You'll see' "character string"
Line: 9, starts at char. position: 23, ; "punctuation symbol"
Line: 10, starts at char. position: 3, _s3 "identifier"
Line: 10, starts at char. position: 7, := "operator"
Line: 10, starts at char. position: 10, '' "character string"
Line: 10, starts at char. position: 12, ; "punctuation symbol"
Line: 11, starts at char. position: 3, _s4 "identifier"
Line: 11, starts at char. position: 7, := "operator"
Line: 11, starts at char. position: 10, ''' "character string"
Line: 11, starts at char. position: 14, ; "punctuation symbol"
Line: 12, starts at char. position: 3, _s5 "identifier"
Line: 12, starts at char. position: 7, := "operator"
Line: 12, starts at char. position: 10, ' ' "character string"
Line: 12, starts at char. position: 13, ; "punctuation symbol"
Line: 13, starts at char. position: 1, end "reserved word"
Line: 13, starts at char. position: 4, ; "punctuation symbol"

Identifier Table:
test
i
_s
_s2
_s3
_s4
_s5
-----Scanner end-----

```

```

-----Scanner start-----
Line: 1, starts at char. position: 1, ProGram      "reserved word"
Line: 1, starts at char. position: 9, test "identifier"
Line: 1, starts at char. position: 13, ; "punctuation symbol"
Line: 2, starts at char. position: 1, var "reserved word"
Line: 3, starts at char. position: 3, #db "wrong identifier"
Line: 3, starts at char. position: 7, : "punctuation symbol"
Line: 3, starts at char. position: 9, float "reserved word"
Line: 3, starts at char. position: 14, ; "punctuation symbol"
Line: 4, starts at char. position: 3, _f2 "identifier"
Line: 4, starts at char. position: 7, : "punctuation symbol"
Line: 4, starts at char. position: 9, float "reserved word"
Line: 4, starts at char. position: 14, ; "punctuation symbol"
Line: 5, starts at char. position: 1, begin "reserved word"
Line: 6, starts at char. position: 3, #db "wrong identifier"
Line: 6, starts at char. position: 7, := "operator"
Line: 6, starts at char. position: 10, .1 "wrong float"
Line: 6, starts at char. position: 12, ; "punctuation symbol"
Line: 7, starts at char. position: 3, _f2 "identifier"
Line: 7, starts at char. position: 7, := "operator"
Line: 7, starts at char. position: 10, 12.100 "float"
Line: 7, starts at char. position: 16, ; "punctuation symbol"
Line: 8, starts at char. position: 1, end "reserved word"
Line: 8, starts at char. position: 4, ; "punctuation symbol"

```

Identifier Table:

test

_f2

-----Scanner end-----

```

-----Scanner start-----
Line: 1, starts at char. position: 1,
(* This line is a comment. *)
"comment"
Line: 2, starts at char. position: 1, program      "reserved word"
Line: 2, starts at char. position: 9, test "identifier"
Line: 2, starts at char. position: 13, ; "punctuation symbol"
Line: 3, starts at char. position: 1, var "reserved word"
Line: 4, starts at char. position: 3, i "identifier"
Line: 4, starts at char. position: 5, : "punctuation symbol"
Line: 4, starts at char. position: 7, integer      "reserved word"
Line: 4, starts at char. position: 14, ; "punctuation symbol"
Line: 5, starts at char. position: 1, begin "reserved word"
Line: 6, starts at char. position: 3, i "identifier"
Line: 6, starts at char. position: 5, := "operator"
Line: 6, starts at char. position: 8, 1 "integer"
Line: 6, starts at char. position: 9, + "operator"
Line: 6, starts at char. position: 10, 2 "integer"
Line: 6, starts at char. position: 11, ; "punctuation symbol"
Line: 7, starts at char. position: 1, end "reserved word"
Line: 7, starts at char. position: 4, ; "punctuation symbol"

```

Identifier Table:

test

i

-----Scanner end-----

