

Project 1 Report / Scanner

0416081 趙賀笙

Overview

這次的Project利用lex實現P Language的scanner，主要掃描的Token有Delimiters，運算符號，關鍵字，identifier，數字，字串，註解，Pseudocomments。

Implementation

- **Delimiters、運算符號、關鍵字：**

單純掃描對應的文字。

- **Identifiers：**

掃描字母開頭，英數字結尾的字串。

- **數字：**

- 整數：分成兩種Token，0開頭為八進位整數，1-9開頭為十進位整數。
- 浮點數：掃描任意數字串加小數點再加任意數字串
- 科學記號：整數或浮點數開頭，接大寫或小寫E，加正負號，最後在整數結尾。

- **字串：**

當找到任意一個雙引號，進入字串模式，進入字串模式後，根據掃描入字的不同有四種反應：

1. 單一雙引號：結束字串模式，該token為合法字串
2. 連續兩個雙引號：視為一個雙引號放入字串內容
3. 換行符號：違法字元，字串未正確結束，程式終止。
4. 其他字元：視為字串內容放入

- **註解：**

根據註解的開頭不同 `//` 與 `/*`，進入不同的Condition，兩個不同的Condition，一個遇到 `\n` 就結束註解模式，一個遇到 `*/` 結束註解模式，中間的字元不做任何處理。

- **Pseudocomments：**

掃描 `//&[A-Z][±]` 這樣的Token，利用lex掃描的優先順序可以比普通註解先找到。再根據英文字母的不同，開關不一樣的控制變數。

How to compile from source?

使用作業系統 Ubuntu 16.04，將 `lextemplate.l` 和 `makefile` 兩個檔案放在相同資料夾。cd 到此資料夾執行**make**指令即可編譯完成。輸入 `./scanner 檔案名稱` 即會顯示掃描結果。