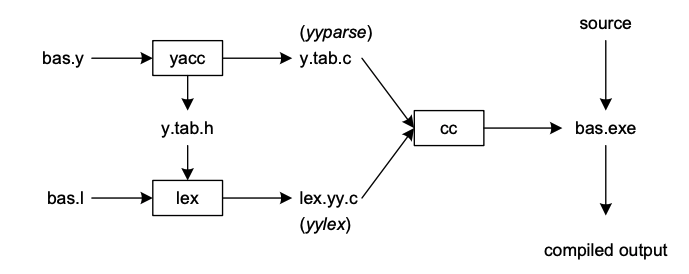
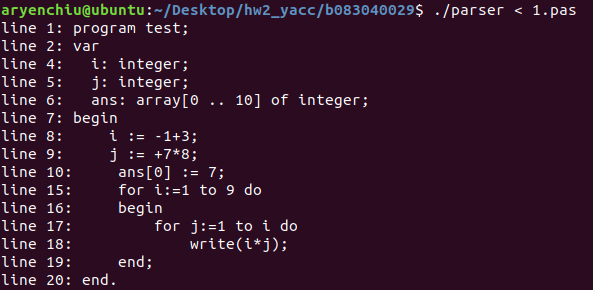
Programming Assignment 2: Pascal Parser

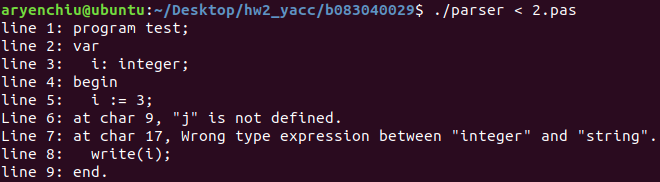
B083040029邱品諺

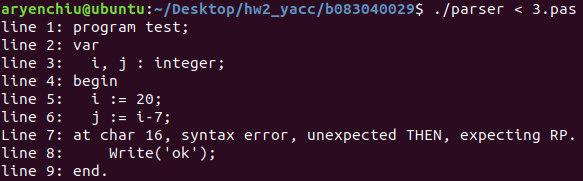
* Environment
  + Operating System
    - macOS 11.4
    - Ubuntu Linux 18.04.6
  + Lex
    - flex 2.6.4
  + Yacc
    - bison (GNU Bison) 3.0.4
* 執行方式
  + 本次作業的目的為撰寫一個Pascal的Parser，簡單來說就是撰寫一個可以判斷Pascal語法是否正確的程式，但在判斷語法正確與否之前，需要有其他工具可以幫忙把程式中的每個token切出來，而這就是Scanner的工作，其負責每個token的定義方式及名稱，使得Parser能透過每個token去定義符合該程式語言的語法，最後得以判斷指定程式的語法是否正確。而本次作業執行了Compiler的Lexical Analysis以及Syntax Analysis的部分。
  + 我對於本作業的撰寫方式主要為：當Parser開始依據撰寫的文法parsing，透過Scanner讀取token時，便利用strcat函式將yytext串接至自行定義的字串中，每當遇到換行符號便印出該行，如果有文法錯誤時，Parser便會停止向下處理，因此需定義error recovery的方式使得parser繼續執行。此外，除了文法錯誤外，還需要在Parser中可能會出現錯誤訊息的文法中自行定義、撰寫判斷錯誤的方式。
* Error Handling
  + 文法錯誤，如：缺少括弧、定義變數型態時利用 ’=’ 而非 ‘:’、缺少必要的符號，像是最後end沒加Period等等。
    - 這些錯誤其實只要在Parser的程式最上面define以下圖片中的程式碼，Parser在parsing時就會回傳詳細的錯誤訊息，再自行加上行號及parsing至的字元數即為一則完整的錯誤訊息。

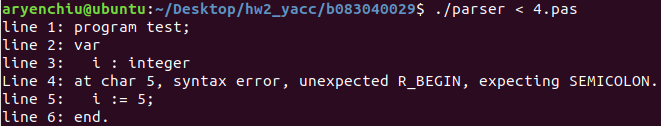


* + 使用未定義的變數、賦值型態與宣告型態不同、不同型態的變數相加
    - 需使用Symbol Table紀錄每個identifier的變數名稱以及宣告型態，並自行定義函式提供查找功能以判斷當前遇到的變數是否有事先宣告過，並自行定義錯誤訊息。
    - 本程式中有5個function來執行Symbol Table：
      * void insert\_name (char \*x);
        + 將變數名稱，亦為在宣告變數時冒號左邊的identifier加入Symbol Table。
      * void insert\_type (int x);
        + 記錄變數的宣告型態。
      * int lookup (char \*x);
        + 用以查看使用的變數是否有先宣告過，如果沒有便產生錯誤訊息。
      * void chktype (int type);
        + 用以檢查變數型態和賦值型態是否相同，如果不同便產生錯誤訊息。
      * void idctype (int x);
        + 用以標示該行當前變數的型態。利用chktype function和assign (:=) 右邊的變數進行比較，檢查是否有型態不同的expression。
* 撰寫時遇到的問題
  + Lex 和Yacc的執行方式
    - 由於本程式是設計在lex印出每一行的輸出，且利用換行符號作為印出的標示，因此最後一行需要在Yacc印出，也表示Lex和Yacc需要共用一個用以記錄輸出訊息的變數，而在Yacc宣告變數為extern即可，由於透過Yacc會先編譯出y.tab.h，而Lex只需include這個header，即可共同使用一個變數。
  + reduce-reduce conflict
    - 由於PDF提供的文法為簡易版的Pascal BNF Syntax，所以需要再基於該文件自行增加一些文法，而增加的過程中容易遇到conflict，因此需要想清楚在Parser時是如何執行的。
* Output

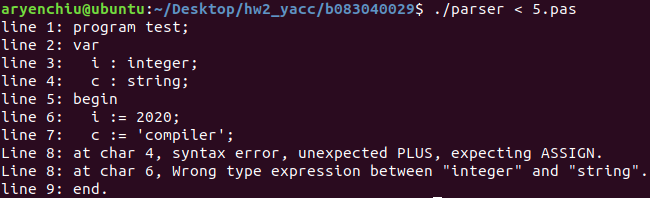


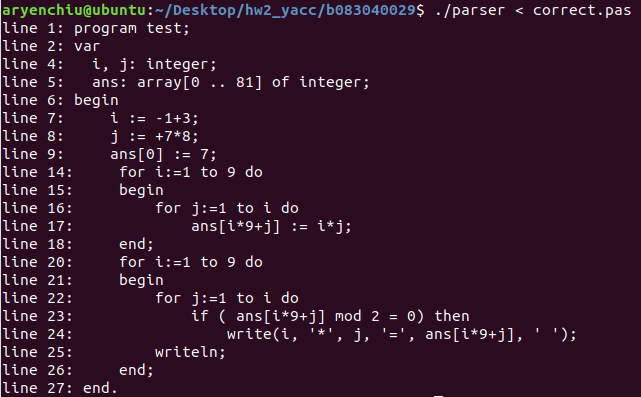


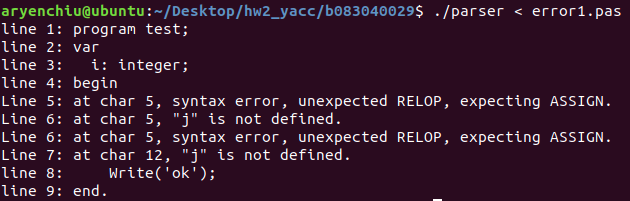
RP為Right Parenthesis、右括弧、’ ) ’

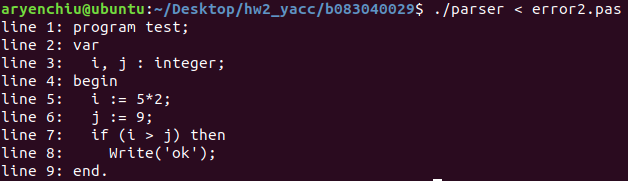
R\_BEGIN為保留字begin，SEMICOLON為分號、’ ; ’

PLUS為加號、’ + ’，ASSIGN為 ’ := ’

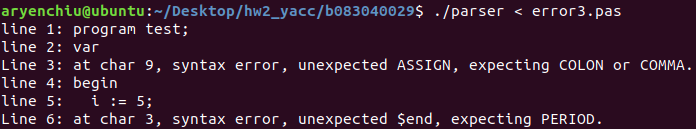




RELOP為Relation Operator，本pascal檔為 ’ = ’，ASSIGN為 ’ := ’



COLON為冒號、’ : ’，COMMA為逗號、’ , ‘，PERIOD為句點、’ . ‘



RELOP為Relation Operator，本pascal檔為 ’ = ’，ASSIGN為 ’ := ’

