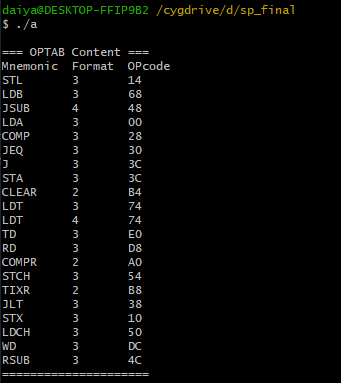
1. The architecture of the implemented assembler

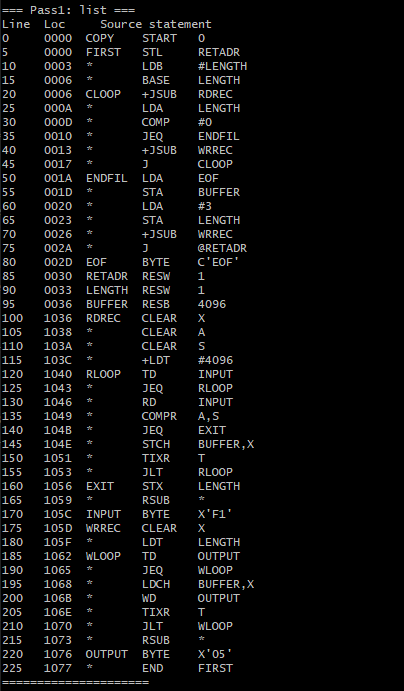
* OPTAB：儲存有支援的mnemonic及其格式與opcode，供Pass1用。
* Pass1 階段(定義符號)：
  + 計算程式中directive跟mnemonic的LOC。
  + 處理好directive，像是BYTE或RESW用多少LOC。
  + Label 不能跟directive或mnemonic一樣。
  + 建立SYMTAB，記錄程式中所有Label的LOC，供Pass2用。
* Pass2 階段(翻譯程式並產生object code)：
  + 翻譯程式（根據不同格式找到其位址）。
  + 處理BYTE等directive的值到object code。
  + 輸出object code到record。

1. Learned and experienced during the implementation

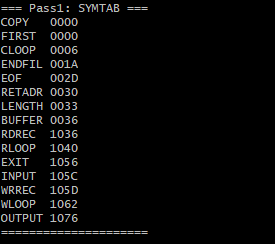
* 12/21 實現Pass1階段：
  + OPTAB：儲存mnemonic、format、opcode。
    - ASMTAB：用來記錄每一行程式的資訊。每一行分割token後分別儲存到此表格的loc、label、opcode、operand，同時更新LOCCTR。
    - SYMTAB：每當處理過程遇到一個label時，LOCCTR的現行值就是該symbol的位址，新增到SYMTAB。



OPTAB 內容

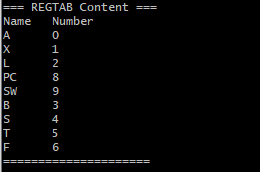


經過 Pass1後ASMTAB內容

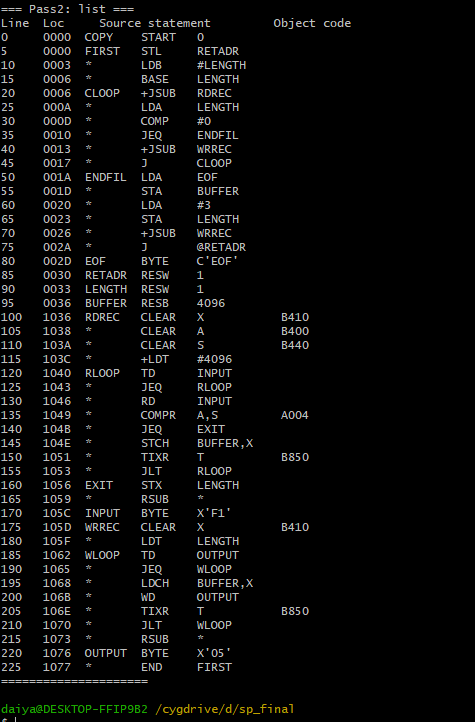


SYMTAB內容

* 12/22 實現部分Pass2需要的功能：
  + REGTAB：存SIC/XE有支援的register及其對應的號碼，處理format 2查表用。
    - 函式format2Process：看operand是一個還是兩個register加上指令的opcode生成object code。

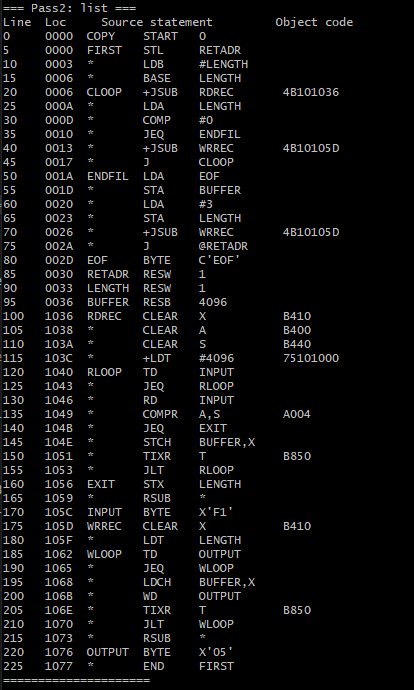


REGTAB 內容



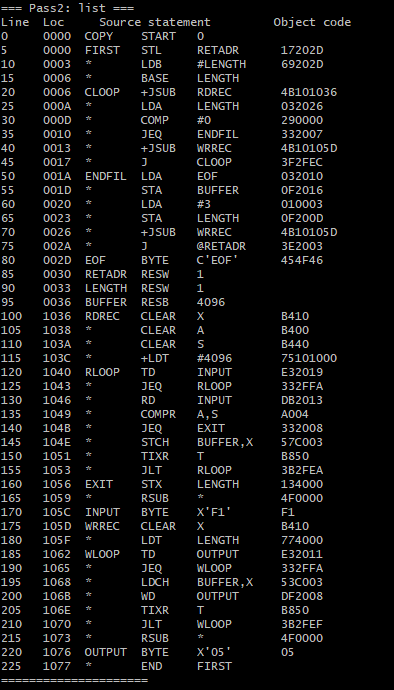
只處理format 2的list內容

* 12/23（白天） 實現部分Pass2需要的功能：
  + - 函式findsymbol：查詢label，給format 4處理address用。
    - 函式format4Process：根據mnemonic & operand設置opcode、flag、address。



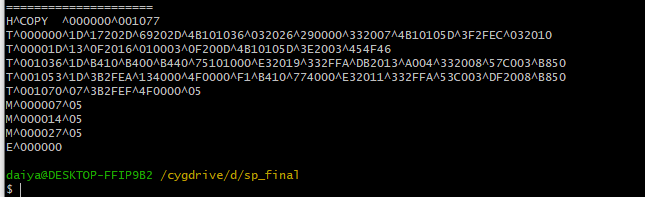
只處理format 2 & 4的list內容

* 12/23（晚上）實現Pass2階段：
* 函式format3Process：根據mnemonic & operand設置opcode & n、i、x、e flag，然後在計算offset算disp，同時間設置p、b flag。



處理format 2 & 3 & 4後的list內容

* + Record：根據課本定義格式輸出objectProgram.txt



1. implement more than the required specification

沒有。

1. Do you make the implementation yourself

是，但我有使用ChatGPT o1。

1. Any thing you would like to let G.H.Hwang know

謝謝老師。