

# 系統程式 期末專題 Report

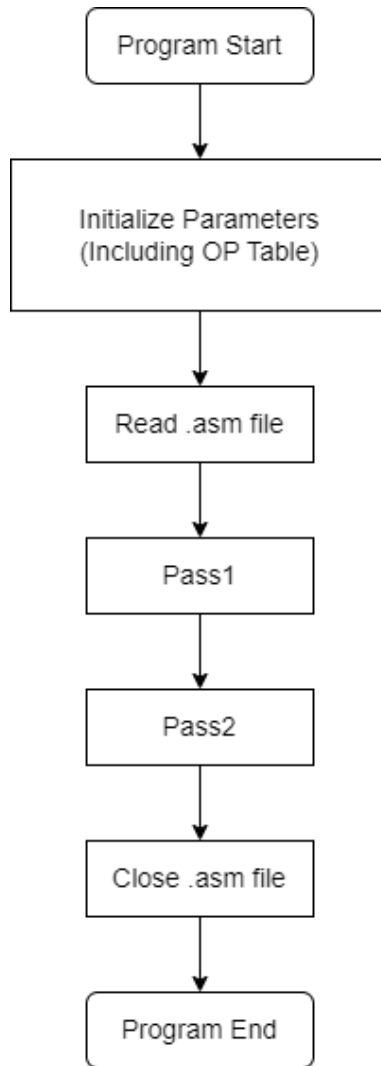
姓名：陳柏瑜

系級：資工系 114 級

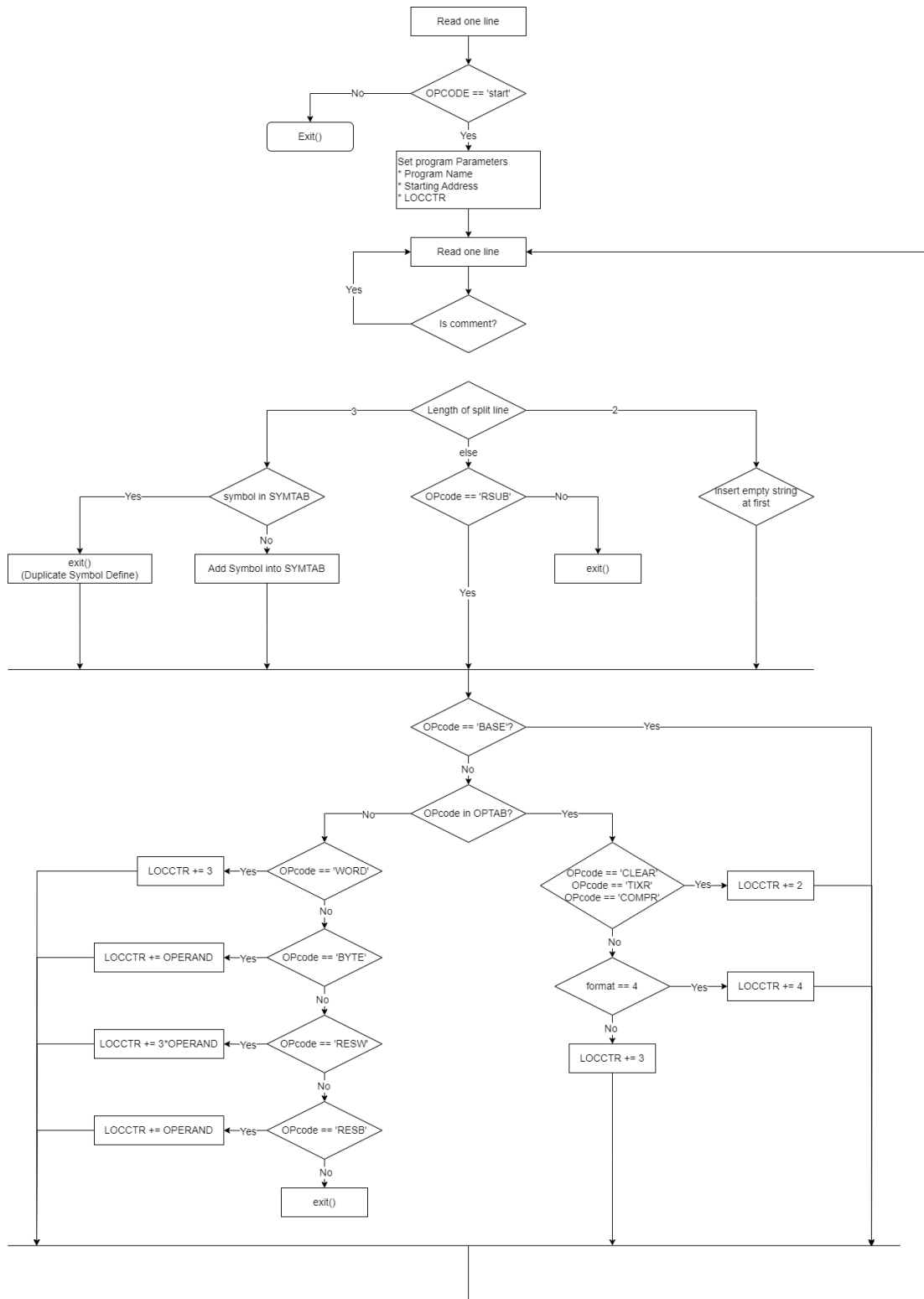
學號：41047054S

## Assembler Architecture

**main()**



## PASS1()



## Lessons and Experiences

這次因為要做比較多字串處理，所以採用 Python 開發，我個人對於 C 比較熟悉，但在衡量開發時間後，比起做字串處理比較麻煩的 C 和很不熟悉的 C++，我選擇用以前有稍微接觸過也還可以勝任的 Python。不過這次主要也從中學到兩件事情

### Python class

這次因為要開發 OP Table 和 SYMBOL Table，想說用 class 會比較好做，管理起來也比較方便，但在開發時才發現不知道要怎麼確認已存在於 class 的元素，舉個例子如下：

我的 class 是這樣定的

```
class OPCODE
{
    def __init__( self, name, opcode ):
        self.name = name
        self.opcode = opcode
}
```

而如果我確定輸入 asm 檔的 OPcode 是否存在，就不知道如何轉寫程式碼達到此效果，最後發現用 dict 寫或許會有辦法，但也礙於不太會寫 dict，所以最後改成用 list 去寫，並使用 `if OPCODE in OPTAB` 的方式達成

### Python Format String

另一個學到的內容是如何製造或印出格式化的字串，尤其這次專題有很多十六進位的數值交替，如果不會轉換是很麻煩的，但除十進位跟十六進位轉換以外，還要同時考慮現在處理的變數型別是整數還是字串，一旦一個地方弄錯其實都會花很多時間處理

## Copyright Claim

此組譯器皆為我自己開發，雖然目前只開發到 PASS1，但所有參考資料都只有課本圖 2.5，圖 2.6，圖 2.8，圖 2.4 的演算法，以及課本最後 Appendix A 的 OPcode 而已

## Comments to this Course

我個人很喜歡這門課，因為每一次上課都能學到一些以前似懂非懂的觀念，經過這門課把這些東西補齊了（尤其是 Unix System Calls 那部分），就課程內容來說我覺得很紮實也很豐富，不過給老師一些小建議，因為本系生在上這門課之前理論上應該上過計算機結構與組合語言，加上計算機結構是必修，其實大家對於組語一定不陌生，老師前半段的課程可以稍微快一些，這樣可以增加課程的豐富度，像是 Unix System Calls 那部分有些東西想聽但沒聽到就覺得有點可惜，例如 IO multiplication，`select()` 或 `epoll()` 的內容最近剛好碰到覺得很重要，但很可惜就沒機會上到

如果就上課方式的話，我覺得維持現況很好，不需要特別調整

最後，想要趁這個機會大大表揚助教，真的覺得助教很認真很用心，每一堂課都會跟課，幫同學解決問題，作業批改上也很有效率，還會一一給回饋，的是不可多得的好助教！