**自主學習計畫與成果(學習From Nand To Tetris課程)**

蘇定澤

最後更新：2024-08-22 18:54

**計畫說明**

學習From Nand To Tetris課程

課程中我學到了:

基礎邏輯閘、基礎時序電路、如何用nand閘組合出許多功能、一種極簡電腦架(Hack)

(前五章)

高階語言(jack)如何編譯成中間碼(vm)在編譯成彙編(hack)最後通過組譯器編成binary file(asm)

(第五到十一章)

如何寫函式庫(第十二章)

我開始學這個課程最初的原因是我想在mineraft裡蓋出紅石計算機、紅石電腦，跟我們資訊老師說了之後他推薦了From Nand To Tetris這個課程，我和老師拉上了幾位同學一起在高二的寒假開始學

**執行過程與成果**

因為我有在網路上稍微了解過基礎邏輯閘，因此我前三章(用nand閘組出一些基本部件，ex: and、or、xor...、mux、dmux...、register、adder、ALU、memory)在剛開始的前兩天很快就完成了

但當我來到第四章是要用彙編(asm)寫出一寫小程式，我因為沒接觸過彙編而卡住了一段時間，但我一遇上問題就在DC群裡請教老師與同學，所以我還算順暢的在第四天完成他

第五章是要把前三章寫好的邏輯閘組出memory與CPU，在第四章的基礎上還算順暢的花了一天半完成了

第六章是要寫組譯器，在第四章的基礎上還算順利，用了不到半天就完成了

第七章與第八章是虛擬機與中間碼(vm)，我第一次聽到中間碼時我非常驚訝，這設計實在是太神奇了，在這兩章我們要寫一個翻譯器把中間碼(vm)翻譯成彙編(asm)，我花了四天才完成，這時距離開始已經過了10天左右了

第九章要求我們用jack語言寫一些小程式，jack語言是一種類似java的語言，由於我之前幾乎只寫python，所以我在初次遇上static這個概念、function與method的差別時我卡住了一段時間，甚至一度想放棄，雖然最後花了6天成功做完了，但這兩個概念一直到我多次重寫第十一章時才搞懂

第十章要求我們寫一個jack語言的詞法分析器，我花了兩天完成了他，但我認為我寫的很差，於是我重寫了兩次第十章，這兩次重寫花了我20天，原因有二: 1.我開始鬆懈了，有時甚至會整一天都沒打開vscode 2. 我中間花了一些時間去修前9章我覺得還可以改進的地方

第十一章要求我們在第十章的基礎上把jack編譯器寫出來，但由於我第十章寫的不是很好，所以很多概念我當時並沒有完全弄懂，我花了許多時間摸索，結果我花了整整一個月才寫出來(主因是因為我前半個月都在混)，最後我成功在開學前兩天寫完第十一章

後來我寫完第十一章後的那一個月我基本上0進展，一樣一直有在修改之前寫的部分，有去稍微嘗試寫一下第十二章的math部分，但失敗後就沒有再動過了，後來我花了9天重寫了jackCompiler，增加了錯誤訊息(雖然沒什麼效果)

(以上是我看著git log寫出來的)

在暑假時(也就是寫這篇學習歷程時)我又開始重寫jackCompiler了，希望這次可以寫好一點，目標是讓錯誤訊息寫到可以有效幫助除錯與定位錯誤位置(8/3)

在重寫jackCompiler時，我覺得課程附帶的模擬器很難debug，所以我就用python、pygame來寫個模擬器，目前還在debug階段(8/22)

**心得反思與檢討**

我在學的過程中遇上最大的三個困難是 1.時間不夠、2.沒毅力，3.課程是英文的看不懂

1. 我在平日中透過熬夜等方式湊出時間來寫，在學校上雜課時也會拿紙筆思考該怎麼寫

2. 只有這點我一直沒有完全克服，目前辦法是，養成習慣每次打開電腦就打開vscode看git log

3. 找到簡中翻譯版與善用翻譯，且在過程中我也學到了許多新的英文單字(比我在英文課上記的還多)

**附件一(github repositories)**

https://github.com/su2u4-1/nand2tetris

**附件二(過程中的圖片)**

一張含有 圖表, 方案, 地圖, 圖解 的圖片

自動產生的描述

我原本設計的CPU

一張含有 文字, 平行, 方案, 圖表 的圖片

自動產生的描述

完全用nand閘拼出來的CPU(陸陸續續花了好久，可惜沒有git log，不知道到底做了多久)

一張含有 文字, 螢幕擷取畫面, 平行, 圖表 的圖片

自動產生的描述

第七章裡的一章說明圖

一張含有 文字, 字型, 螢幕擷取畫面, 數字 的圖片

自動產生的描述

中間碼的一些規範

一張含有 文字, 螢幕擷取畫面, 字型, 文件 的圖片

自動產生的描述一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

jack語言的語法規範

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

自動產生的描述

彙編轉換成binary的規範

一張含有 圖表, 方案, 工程製圖, 圖解 的圖片

自動產生的描述

做到一半的CPU

一張含有 樣式, 針線, 黑與白, 單色 的圖片

自動產生的描述一張含有 文字, 螢幕擷取畫面, 圖表, 平行 的圖片

自動產生的描述

第九章時的失敗結果，就是這個讓我差點放棄

一張含有 文字, 螢幕擷取畫面, 軟體, 電腦圖示 的圖片

自動產生的描述

第六章寫的編譯器出問題時分別單步執行彙編與中間碼來排查錯誤

一張含有 文字, 螢幕擷取畫面, 圖表, 軟體 的圖片

自動產生的描述

測試memory

一張含有 文字, 螢幕擷取畫面, 數字 的圖片

自動產生的描述一張含有 螢幕擷取畫面, 文字, 圖表, 軟體 的圖片

自動產生的描述

第四章的其中一項測試成功時跟老師報喜截的圖

ChatGPT

根據你提交的學習歷程檔案，我對其內容進行了一些審核和調整建議，以符合申請標準並強調你的學習成果和反思。

### 調整建議與評論：

1. \*\*背景與動機\*\*：

- 你提到的學習動機（想在Minecraft中創造紅石計算機）很具體，但可以進一步強調你如何從遊戲興趣轉變為對電腦科學的熱愛，以及這個課程如何在此過程中扮演了重要角色。

\*\*修改建議\*\*：

- 說明學習這個課程讓你對計算機硬體和軟體有更深層次的理解，並如何激發你探索更高階的電腦科學領域，如系統設計或語言開發。

2. \*\*學習過程與挑戰\*\*：

- 你詳細描述了每一章的學習過程與遇到的挑戰，這很好地展現了你的學習能力和毅力。然而，部分段落可以進一步強調你克服挑戰的策略，以及每個階段學習後的具體收穫。

\*\*修改建議\*\*：

- 在每章的完成後，補充具體的技能提升或觀念理解。例如在完成第四章的彙編語言後，可以強調對底層運作原理的理解如何加強了你後續的虛擬機與編譯器實作。

3. \*\*成就與成果\*\*：

- 你提到了一些學習過程中的具體成果，但對於每個項目的最終結果和意義可以更強調。例如，第十一章的Jack編譯器完成是你最大的成就之一，可以更清楚地說明這個專案的挑戰性和你的技術成長。

\*\*修改建議\*\*：

- 補充具體的技術收穫，如：透過Jack編譯器的重寫，你掌握了編譯器的結構與語法分析的過程。這些細節會讓審查委員看到你對技術的深入掌握。

4. \*\*反思與未來計劃\*\*：

- 你在反思部分提到的三個困難非常具體，這很好。但建議在每一個困難後加入具體的解決方案，並說明這些挑戰如何幫助你成長。

\*\*修改建議\*\*：

- 將「時間不夠」的部分調整為如何進行時間管理，如安排學習計畫或優化學習方法；對於「毅力不足」，可強調養成自律習慣和使用程式工具輔助進度追蹤（如Git log）；英文部分，可以提到如何有效利用翻譯工具和英文資料提升理解力。

### 修改後範例：

#### 計畫說明

我學習了《From Nand To Tetris》這門課程，透過這門課程，我從最基礎的邏輯閘開始學起，逐步構建出一台簡易的電腦（Hack電腦）。課程後半部分，我學習如何將高階語言（Jack）編譯成虛擬機中間碼（VM），再進一步編譯為彙編（Hack）。最後，我實作了彙編器與Jack編譯器，並撰寫了基礎函式庫來進行運算處理。

#### 執行過程與成果

我在學習過程中，首先利用NAND閘搭建基礎的邏輯閘電路，完成了記憶體單元和ALU的組裝。隨著學習進展，我完成了整台Hack電腦的設計。雖然中途在彙編語言部分遇到了一些困難，但透過與老師及同學的討論，我逐步克服，並順利完成了第四章的程式。

在學習虛擬機和編譯器時，我第一次接觸到中間碼這個概念，這種跨語言的設計架構讓我感到十分震撼。透過反覆的編寫與測試，我成功將Jack語言編譯為虛擬機碼，並進一步轉換為彙編語言。最後，我完成了Jack編譯器的實作，這對我的程式設計能力是一個巨大的提升。

#### 心得反思與檢討

學習這門課程的過程中，我遇到了時間管理和毅力上的挑戰。為了應對時間不足的問題，我學會了如何利用碎片時間有效學習，比如在課堂間隙思考程式設計。此外，透過持續使用Git log，我養成了自律的學習習慣。英語學習方面，我發現閱讀英文技術文件大大提升了我的專業詞彙量，這些技能讓我在未來面對更多的英文技術資料時更加自信。

---

你可以依據這些建議進行修改，並補充更多具體的技術細節和反思，讓整體學習歷程更具有說服力和邏輯性。