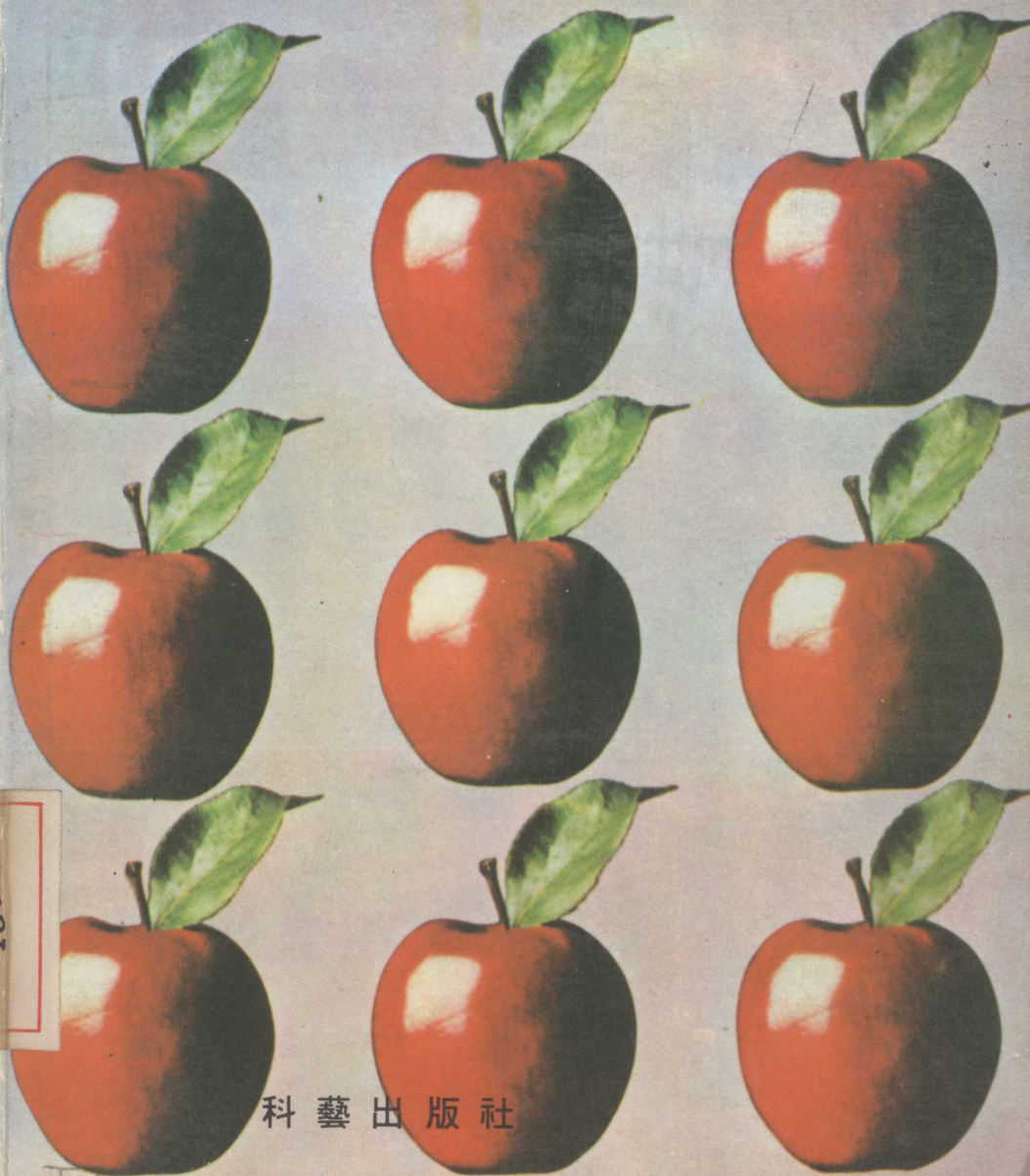


APPLE INTEGER BASIC 程式語言

蔡家顯 譯



科藝出版社

515



APPLE INTEGER BASIC 程式語言

蔡家顯 譯



科藝出版社

APPLE INTEGER BASIC 程式語言

編譯者：蔡家顯

出版者：科藝出版社

發行者：科藝出版社

地址：九龍西洋菜街102號三樓

印刷者：達華印刷廠

地址：香港柴灣工廠大廈10樓

定價：港幣 \$ 18.00

目 錄

介 紹.....	1
第 1 課 清除 VDM 螢幕.....	5
第 2 課 載入並存放程式.....	11
字 彙.....	11
程式範例.....	12
存放一個程式.....	12
核對程式.....	14
音量與音調.....	15
磁 帶.....	15
第 3 課 程式撰寫與 PRINT 規則.....	17
字 彙.....	17
程式範例.....	18
習 題.....	20
第 4 課 運 算.....	21
字 彙.....	21
數學運算.....	22

	程式範例·····	23
	習 題·····	27
第 5 課	捨棄與整數·····	29
	字 彙·····	29
	程式範例·····	29
	習 題·····	31
第 6 課	模擬的實數·····	33
	字 彙·····	33
	程式範例·····	34
	習 題·····	40
第 7 課	總 合·····	41
	字 彙·····	41
	習 題·····	49
第 8 課	流程圖·····	51
	字 彙·····	51
	問 題·····	52
	程 式·····	53
	流程圖·····	54
	流程圖練習·····	57
	習 題·····	62
第 9 課	迴路與計數變數·····	63

	程式範例·····	64
	習 題·····	68
第10課	約定規則與判斷敘述·····	71
	字 彙·····	71
	判斷敘述·····	71
	習 題·····	75
第11課	程式的一般概括·····	77
	字 彙·····	77
	練習問題·····	78
	習 題·····	82
第12課	使用計算機·····	83
	字 彙·····	83
	練習程式·····	84
	TRACE 函數·····	85
	NOTRACE 函數·····	86
	計算機運轉·····	86
	習 題·····	87
第13課	下標的變數·····	89
	字 彙·····	89
	程式範例·····	90
	使用註標變數之程式·····	93
	習 題·····	97

第14課 MIN-MAX與SORT 99

字彙..... 99

問題..... 99

MIN-MAX 程式..... 101

MIN 程式..... 103

SORT 程式..... 106

習題..... 109

第15課 字串與GOSUB 111

字彙..... 111

字串的例子..... 112

習題..... 129

第16課 函數 131

字彙..... 131

程式範例..... 131

習題..... 135

第17課 有效率的程式撰寫 137

程式範例..... 137

習題..... 144

第18課 作圖 145

字彙..... 145

程式範例..... 146

	習 題.....	157
第19課	Menu與旗.....	159
	字 彙.....	159
	程式範例.....	159
	習 題.....	164
第20課	遊 戲.....	165
	程式範例.....	165
	習 題.....	170
附 錄		171
	SORT 程式的進一步討論.....	171
	流程圖資料補充.....	174
	更多的作圖顯示.....	178

介 紹

本書的主要目的，是為一些沒有程式設計的基礎而卻很想學習的人所寫的。當我在打字時，我發覺許多的資料是一再的重覆，所差別的，僅僅是一些觀念的不同而已，但是，語言的了解是因人而異的，因此，若你覺得這些一再重覆的觀念很無聊，我只得向您致無限的歉意，但若是因為這一再重覆的資料對於你學習和了解有很大幫助的話，那本書的目的就算是完成了。

程式設計是詳細的，吃力的，令人思考的非常難熬的工作，別想一夜的工夫就能變成一個程式設計的專家，就像一些有價值的事情一樣，它是須要花費你的一些精神去了解這些新的東西，常常一些令人迷惑的廣告使得大家以為任何人都可以設計程式，有一個廣告說一位 13 歲的孩童甚至連程式設計的手冊都尚未看完就設計了一個太空戰爭遊戲的程式，你可不要相信它。

在程式設計的範圍內，每個符號，字元、逗點、分號、冒號，在機器上都有特別的意思，而你就必須知道這些符號在機器上的明確意義。這不是一本有趣的書，在書上記載的每一件事都表明了一些用處，你的程式設計能力和

本書中的話有很大關聯，所以請務必詳細的閱讀，並且將它消化。

程式設計是「建築在基礎上」的最真實形態，一個石頭搭建在另外一個石頭之上，所以，假如你並不了解第一課，那麼不要進入第二課。每個規則不僅它本身的意思要了解，更重要的是與它相關的一些規則也須透徹的知道，在微電腦上，必須告知這機器做什麼且依何種次序去做，微電腦是不知道什麼事情的，所以你必須比機器先設定你所要做的工作，再由機器重做你的事情。每一課你必須深切的了解、練習，並在進入下一課之前將它記憶清楚，有一句口頭禪是這樣說的：「硬體是簡易的，但軟體却是困難的」(Harnware in Soft, but Software in haid)

當你在學習這些規則時，你可以在一個程式中改變一些規則，然後去體驗這當中所引起的變化，擴大你學習的經驗對於你是最好的。

但是千萬不要強迫自己去故意犯錯以獲得經驗，如此一來，只是增加挫折感和減低學習的能力而已。假如你白天的學習能力高，那麼將你學習的時間設定在早上，假如你晚上工作情緒高，那麼將學習的時間設定在晚上，可能的話，在你感覺最敏銳的時刻和環境學習。

若你學習了新的規則，你必須不斷的練習，練習又練習，否則你將會忘記它。

除非是硬體上的錯誤，否則電腦是不會錯的，你可能聽到這樣的事：若帳單有錯，則有人會說一電腦出錯一，別相信這種話，這如果不是程式設計有問題，便是輸入資

料時有誤，假使一個錯誤是來自電腦，它可能是人爲的錯誤，你最好叫人來改正這個錯誤。

電腦在我們的日常生活和未來佔了相當大的一部分，你對微電腦的操作和程式設計了解的愈多，你對未來的了解也愈深。

第 1 課

清除 VDM 螢幕

在開始運作之前，必須先將 Apple II 計算機 (computer) 的開關打開。此一開關位於計算機的左後方。接下來將 TV 開關也打開 (設到 “ ON ” 的一邊)，此一 TV 電路又稱為影視顯示模組 (video display module 簡稱 VDM) 或稱為螢幕 (screen)。鍵盤 (keyboard) 上的燈會亮起來，螢幕也會亮起來。Apple II 的鍵盤顯示在圖 1.1 中，詳細的描述與說明可以在 Apple II Basic Programming Manual 中第 6 頁到第 16 頁找到。

接下來，依照下列的順序步驟，你就可以將螢幕清乾淨：

1. 按下 RESET 鍵，然後放開手。
2. 同時按下控制 (CTRL) 鍵以及 B 鍵，然後手放開。
3. 按下 RETURN 鍵，然後手放開。

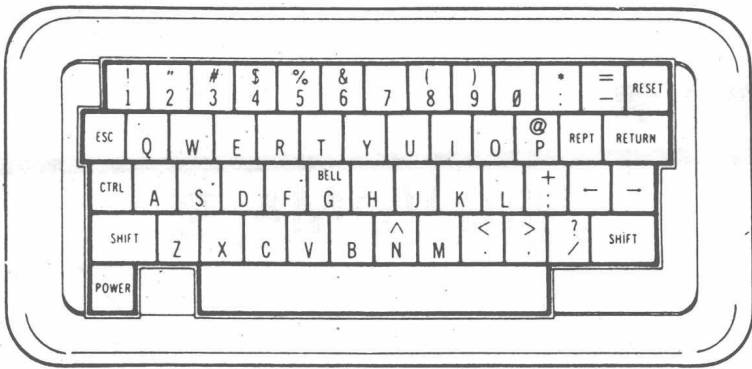


圖 1.1 Apple 鍵盤

4. 按下 ESC (escape) 鍵，然後手放開。
5. 同時按下 SHIFT 以及 P 鍵，然後手放開。
6. 按下 RETURN 鍵，然後手放開。

這些步驟作完之後影視顯示模組 (VDM) 會被清除，只剩下螢幕的左上角有個閃動的游標 (cursor)。此時 VDM 已準備好要從鍵盤接受輸入了。有一個規則要注意，每次叫計算機作什麼動作時，按完了操作鍵最後必須按 RETURN 鍵，否則計算將不會開始工作。

游標可由向右位移 (shift left) 或向右位 (shift right) 的箭頭鍵，來調整位置。敲進 1, 2, 以及 3。若要刪除某個 (些) 數，可以用向左位移鍵將游標移向左，直到游標對上了某個數為止。假如將游標移到 3 的位置 (將 3 蓋住)，然後按下 RETURN 鍵，則 3 會被清除，而螢幕上只留下 1 與 2 兩個字。游標再往下位移一行，出現了一行空行。然後敲進 ASD，再將游標移回 D 字的上面

，按下RETURN鍵。則D被清掉，但是計算機却會發出叫聲，並印出AS以及***SYNTAX ERR。等幾個字。螢幕會變為：

```
> 12          (12 稱為數字 (numeric characters))
> AS          (AS 稱為文字 (alpha characters))
*** SYNTAX ERR
> ■
```

(12 AS 稱為文數字串 (alphanumeric character string))

文法的錯誤 (syntax error) 表示計算機 (computer) 不接受這種文字資料。在撰寫程式的時候，每一個程式敘述 (program statement) 都必須以一個行號 (line number) 開始。行號的大小由0到32767，每次增加10。下面的例子顯示為什麼32767是所能使用的最大行號。將此程式打入：

```
10    LET  A = 2    (按RETURN)
20    LET  B = 3    (按RETURN)
32768  END          (按RETURN)
*** > 32767  ERR
```

因為32768不為微計算機 (microcomputer) 所接受。任何數目假如超過-32767或+32767，都會使計算機印出*** > 32767 ERR，因為Apple II 微計算機使用的是Integer BASIC，在設計之初就規定將數目的大小限制在-32767與+32767之內。

8 INTEGER BASIC 程式語言

現在敲 LIST 並按下 RETURN 鍵。LIST 會使得已經敲進去的所有程式敘述。被顯示在螢幕上。

其實在微計算機以及 VDM 都打開（電源）之後，有更簡便的步驟使得使用者可以更快地將程式寫入：

1. 按 RESET 鍵後放開手。
2. 同時按下控制 (CTRL) 鍵與 B 鍵，然後手放開。
3. 按下 RETURN 鍵，然後手放開。

不過這一系列步驟，並不會將螢幕清乾淨。

假如在計算機運算過程中，RESET 鍵被突然按下，則一個機器語言碼 (machine language code) 會顯示於螢幕之上。爲了得到微計算機的重新控制，並使之回到 Integer BASIC 狀態，必須同時按下控制 (CTRL) 鍵以及 C 鍵。（此後，CTRL 鍵爲了清楚起見均寫爲 CONTROL。而且，當兩個鍵同時按下時，我們在它們中間以三個短線代表，如：CONTROL....C。）此時 C 字不能顯示在螢幕上，假如 C 鍵顯現於螢幕之上，表示 C 鍵按下的時間早於 CONTROL 鍵，這樣整個動作就無效，必須 CONTROL 鍵先按下，手不放開；緊接著又按下 C 鍵。注意，CONTROL....C 並不會將計算機的記憶體 (memory) 清除，CONTROL....B 才會。另一種將記憶體清掉的方法是，敲 NEW 然後再按 RETURN 鍵。核對記憶體是否被清掉的方法，是敲 LIST 再按 RETURN 鍵。假如螢幕上什麼都沒有顯現出來，表示記憶體已被清除。

所以清除記憶體的方法有兩種：

1. 按 CONTROL....B 並按 RETURN 。

2. 敲NEW並按RETURN

要核對記憶體 (memory) 是否清除 (在NEW—RETURN 之後)

方法爲：

1. 敲LIST 並按RETURN 。 (如果螢幕上沒有顯現東西，表示記憶體已清除) 。

敲進剛剛的第10行號與第20行號之程式，然後LIST (列印) 出程式。現在按CONTROL....C, 再按RETURN, 則浮標 (cursor) 會跑回螢幕上，敲LIST 並按RETURN, 則程式又回到螢幕上。現在按RESET鍵，再按CONTROL...B, 放開手後再按RETURN。然後敲進LIST 並按RETURN, 則螢幕就不再顯現出程式了。