

APPLESOFT

程式語言

林聖修
楊愛能

編譯



協群科技出版社

73.876

L62

APPLESOFT

程式語言

林聖修
楊愛能

編譯



協群科技出版社

APPLESOFT 程式語言

編譯者：林聖修 · 楊愛能
出版：協群科技出版社
發行：協群科技出版社
印刷者：廣源印務局
香港中環卑利街684號二樓
青山道875號工廠大廈

定價：H.K.\$ 30.00

前言

程式設計是一項詳細、精確，而又富思考性的刺激活動。別寄望於一個晚上就成為專家。也許一些誇大的廣告會使得你以為程式設計也沒什麼。這並不真是如此。程式設計要花相當大的努力，而這是許多人不願，或是無法，付出的。更甚者，由於一部計算機中的程式很可能無法在另一部計算機中執行，因而除了程式設計之外，你還必須要瞭解到有關計算機的設計問題。要知道，若干程式語言的細節部份只在一種機器上確實有效。

在程式設計中，每一個字母，每一項目，每一方程式，及每一格式，都代表著計算機上一些特殊的意義。要讓計算機正確的動作，就要規畫出精確的步驟。任何不正確的程式設計，或將一些步驟任意放入程式之中，都會使程式執行失敗。這失敗，可能是一部份的，可能是整體性的，但程式中都不會產生希望的正確結果。背誦條文式的記憶法多多少少可以幫忙設計程式，但增強明瞭其中一些規則和關係的理解方式，更可產生較有效率的程式。

程式設計是建立基石的最真實的方法。要開始另一處理程序之前，其立足點必須為一基固的基石。計算機本身是部沒有生命的個體，只能按照被設計的一些規格來行動。計算機本身

APPLESOFT 程式語言

並不懂任何東西，也不對任何東西做任何假設。計算機只會精確地照著你要它執行的方式去做。程式設計師必須要瞭解機器本身的能力，以及應用適切的程式設計規則，以產生正確的結果。

本書中的各課是爲了表達出程式設計的邏輯規則，詳細步驟，以及一些進步的方法，因而設計而成，故涵蓋由初學者至高水準不同的層次。本書企圖建立起一個參考點。若你的程式設計有問題，你永遠可以回到上一參考點，以嚐試去了解正確的程序。當正確的程式設計程序都能了解之後，你就要實際練習了。實際練習可增強你學習的經驗。當你練習得很熟練時，你就會得到莫大的快感。

不要勉強追求學習經驗。多數的學習效果是由正常方法而得。同樣的環境以及處所可幫助學習。有些人早上讀書最好，有些人卻是晚上較好。試著找找你自己的時間和地點，以享受最大的樂趣。

除非計算機的硬體有問題，否則它不會出錯。我們常常聽說“這都是計算機的錯”，你可千萬別信以爲真。若計算機出來結果有錯，則此錯誤必由人爲因素所造成。此種錯誤可予修改過來。

計算機在今日已是今日我們生活的一大部份。你對計算機系統操作，程式設計知道的愈多，你對未來就愈容易瞭解。未來的世界大部份都將和計算機作業有關。

喬治 H · 黑木

簡 介

在 1978 年十一月，我和柏安買了一部 Apple 計算機。我對它抱以極大的期望，希望可以用它來解決我過去所無法解決的一些難題。這些難題包括：怎麼賺上一百萬，怎麼在存貨市場上戰勝，以及如何解決能源問題。在使用了計算機幾個月之後，我發現我碰見了更大的難題。我竟然無法寫出一個能解決很容易的問題的程式。所以，我必須去學著如何來寫計算機程式。我本身並沒有數學方面的背景，所以“實數”和“整數”對我而言，並沒有什麼分別。我除了學著寫程式之外，還要學一些和計算機有關的術語。在買了 Apple 兩年之後的今天，我已經會寫一些很簡單的程式了。若沒有柏安的話（他是計算機工程師），我根本沒法子學到怎麼去寫程式。

並不是每個人都有機會，讓一個計算機程式回答他有關於計算機以及程式設計方面的問題，可能一個人根本就無法很正式的來學程式設計。對於那些想學計算機，卻又沒有人能在旁教導他，也沒有正式學習機會的人而言，本書是最好的指南。本書之目的，乃在於爲了對程式設計了解不多，種人幫助的人而設計。本書是提供給對程式設計有興趣的人一節程式設計手冊，不需要基本背景或幫助。

APPLESOFT 程式語言

若你能夠明瞭，辨認出計算機程式的最基本水準，則此書將給你莫大幫助。若你已看過了 Kunth 的 *Fundamental Algorithms*，則你顯然已不需要本書了。不必再浪費你的金錢或時間。

本書針對初學者，由最簡單的程式設計開始，循序漸進到高等程式技巧，以用到設計藝術的最高層。為簡明起見，我們也對十六進位系統、組合語言、以及機械語言列出一些參考資料。這已夠讓一個人很精通於 BASIC 語言了。

我們把本書分為三節，Applesoft 語言，程式設計，以及補充資料。

第一節是講 Applesoft 語言。假設你有一部 Apple 計算機，把開關打開，鍵盤上的小燈就會亮。第一課就由這裡開始，介紹你詳細的程式設計工作及原則。這解釋可能會很沉悶，同時一直在重複，但程式設計本身是精確的，任何改變都足以影響程式的結果。解釋的方法避免掉了一些計算機含混的專用名詞，而採用日常用語。計算機界有句老話，最足以說明此一狀況。“若你不能使它炫耀發光的話，就用賬單毀了它吧”。

第二節是程式設計。這和邏輯、方程式、設計師之間的思想交流，程式本身，以及計算機都有關。本節的內容將各位由手工時代帶入思維時代。本節可激發腦袋裡較少用到的細胞開始活動。你可以感覺到裡面的鏽在脫落，神經也會較緊張而專注。

第三節是補充資料。提供一些模擬系統，電玩遊戲，以及繪圖的方式。

附錄 A 中，列出了繪圖的一些常用技巧。

附錄 B 中，是 BASIC 程式的一些簡介。

本書中所討論之對象，為使用 microsoft (Applesoft)
語言的 Apple 計算機。我們並沒有直接和 Applesoft 公司，
Bell 或 Howell. 有任何關係。

目 錄

第一篇 APPLESOFT 程式語言

第一課	程式之載入和儲存	1- 1
第二課	印字的規則	2- 1
第三課	用 HTAB, TAB, 及 VTAB 描述 控制輸出格式	3- 1
第四課	變 數	4- 1
第五課	運算的優先次序	5- 1
第六課	迴圈程式	6- 1
第七課	相關與邏輯運算子	7- 1
第八課	解決問題和流程圖	8- 1
第九課	設計高效率程式之規則	9- 1
第十課	總計，計數，和旗標	10- 1
第十一課	單下標變數	11- 1
第十二課	雙下標變數	12- 1
第十三課	字串矩陣	13- 1
第十四課	函 數	14- 1
第十五課	程式列印和編輯	15- 1

第十六課	玩計算機	16-1
第十七課	保留字	17-1
第十八課	選擇與編寫式子	18-1
第十九課	程式概要	19-1
第二十課	清除	20-1

第二篇 程式設計

第二十一課	接近問題的重心	21-1
第二十二課	程式的伸縮性	22-1
第二十三課	圓列、疊堆和指標	23-1
第二十四課	分類、尋找及刪除	24-1
第二十五課	公式解	25-1
第二十六課	雙下標的數列	26-1

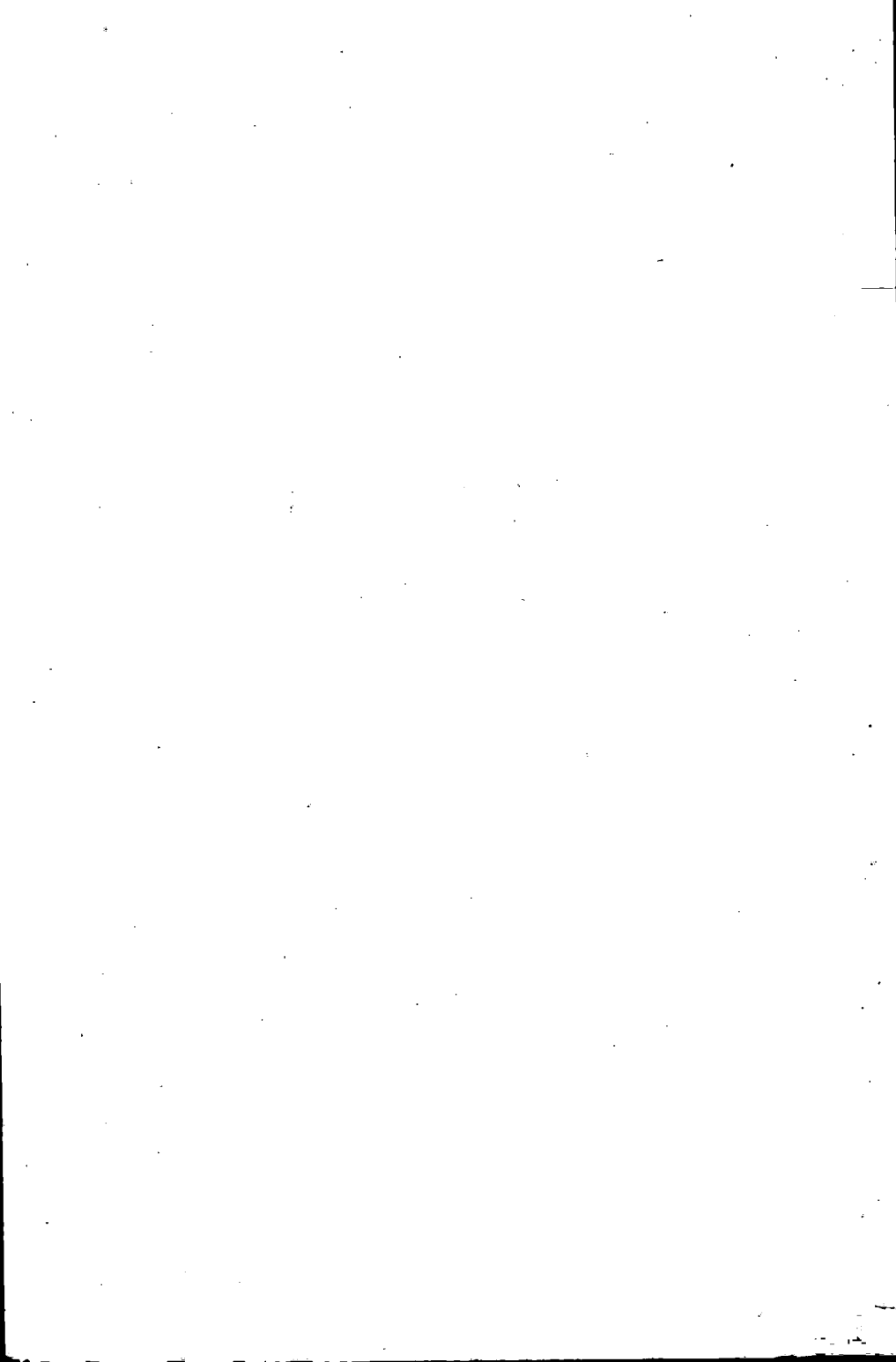
第三篇 補充資料

第二十七	繪圖法	27-1
第二十八課	高解析度的繪圖	28-1
附錄A	計算機的繪圖	附A-1

第一篇

APPLESOFT

程式語言



第一課

程式之載入和儲存

在讀完本課之後，你將能夠：

1. 由鍵盤打入程式並將程式存於記憶體內。
2. 將程式存於卡式磁帶內。
3. 將程式卡式磁帶讀出存於記憶體內。
4. 將程式存於磁碟。
5. 將程式由磁碟讀出存於記憶體內。

常用術語

CRT —陰極射線管(cathode ray tube.)之縮寫。也就是電視之映像管。在此CRT用來顯示由 Apple 電腦輸出和由鍵盤打入輸入之資料文字等，顯示於螢光幕上。(或稱終端機)。

游標 (Cursor) — CRT 在 NORMAL (正常) 狀態時，此為在螢光幕上顯現之黑底白色之閃爍方塊。而在 INVERSE 狀態下，此則為白底黑色的閃爍方塊。

輸入 (Input) —此為由週邊裝置傳送資料或程式指令至記憶

體之過程。也可以指資料本身為輸入。有時亦可指一信號送入一電路或裝置例如脈波信號等為“輸入”。

行號 (*Line Number*) — 此為在程式中每一指述 (*statement*) 開頭的一正指述代號。

LIST 命令 — 此命令將整個程式顯示於 CRT 螢光幕上。如 *LIST 0,100* 則將程式自 0 至 100 之指述全部列於螢光幕上。

LOAD 命令 — 此命令由卡式磁帶將程式讀出並存於記憶體內。而 *LOAD USA* 則將一叫“USA”之程式由磁碟讀出放於記憶體內。

NEW 命令 — 此命令將清除電腦記憶體中現有之程式及所有的變數。

程式 (*Program*) — 為一套指述或指令組合而成，用來指揮電腦如何去解決一問題或完成一些工作。

程式指述 (*Program Statement*) — 此為存於記憶體中指揮電腦之執行的獨立一行的指令，每一指述均以一正整數為起始。

RUN 命令 — 此命令先清除所有變數，指標 (*pointers*) 及堆疊 (*stacks*)，並由指定之行數開始執行。一般 **RUN** 多由最低之行數開始執行，執行整個程式。

SAVE 命令 — 將程式存於卡式磁帶。而 *SAVE USA* 則將一叫“USA”之程式存於“磁碟”上。(程式設計者可以指定此 **SAVE** 動作為存入磁帶或存入磁碟內)。

VDM — 為影像顯示模組 (*video display module*) 之簡寫。即為顯現資料之電子螢幕。

討 論

在研讀本課之前題為 Applesoft 須先存於 Apple II 電腦之適當位置。當此 Applesoft 由 ROM (唯讀記憶體)、磁碟或磁帶讀入電腦後，在螢幕之左方將出現指示字 (prompt) (`>`) 及游標 (cursor) (`█`)。

在舊型的 Apple II 中以整數 BASIC (Integer BASIC) 為標準裝置。而 Applesoft 則為存於 ROM 或磁帶上附加供應。但現在之 Apple II Plus 則以 Applesoft 為標準 ROM 卡，整數 BASIC 則為附加之語言卡，可分開選購。

由鍵盤打入之程式將被存於記憶體內。而重要的程式可用 SAVE 將之儲存在磁帶或磁碟上。由磁帶或磁碟所存之程式庫 (program library) 吾人亦可以將程式讀入記憶體內。而這些程式即可視為軟體 (software)。

螢幕被清除掉後，打入下列程式：

```

10 PRINT "THIS IS THE USA"
20 PRINT
30 PRINT "THIS IS THE"
40 PRINT "UNITED STATES"
50 PRINT "OF AMERICA"
60 PRINT
70 PRINT "THIS IS THE",
80 PRINT "USA"
999 END

```

接著打入 RUN 並按下 RETURN 鍵後，程式的輸出即顯示現在終端機之螢幕上。若欲查看輸出與程式之關係，可打入 LIST 按下 RETURN 鍵，則程式將整個列於螢光幕上。若要

將程式存於磁帶可依下列步驟進行：

1. 打入 SAVE 命令—“暫時不要按 RETURN 鍵”。
2. 將磁帶放入錄音機內並倒轉至磁帶開始處。
3. 按下錄音機“停止”或“暫停”(stop-pause)鍵。
4. 將錄音機的計數器歸零。
5. 讓錄音帶前進至固定圈數(如,計數器(counter)轉至 005)為錄存之起始計數。
6. 調整錄音機音量至“6”處,而音調(tone)於中間位置。
7. 按下“錄音”(record)跟“前進”(play 或 Forward)鍵,開始錄音。
8. 此時才按下電腦鍵盤之 RETURN 鍵。按下後,游標(cursor)將會消失且電腦會發出“嗶”聲,表示程式開始錄存於磁帶內了。
9. 當程式錄完後,電腦會再發出“嗶”聲,而游標也將回至螢幕上。
10. 此時按下錄音機的“停止”鍵,即結束錄存的動作了。

程式錄完後,最好依下列方式在磁帶盒子或外壳上標明此程式,以便下次使用時,易於找尋此程式之位置。標明:

1. 此程式之名字及其功能說明。
2. 此程式錄於磁帶時,錄音機之開始計數(如上述此計數為 005)。
3. 錄完後,錄音機上的終結計數。
4. 錄時錄音機的“音量”及“音調”(tone)的大小。(一般如用 Panasonic 錄音機,音量多為 6,音調調至中間即可適於 Apple II 之 SAVE 和 LOAD 程式的使用)。

爲了幫助在 LOAD 程式時易於找出所要之程式，最好能將磁帶上程式的各項記錄好好保存起來。爲了能保護磁帶，在 15 分鐘之磁帶內最好只錄四個程式，而程式只錄於磁帶之單面。對於較重要的程式必須再複製拷貝於另一捲磁帶上並置於較安全的地方。

接着你可以把剛剛錄存於磁帶上的程式讀出置於記憶體內，以便檢查所錄程式有無錯誤。其 LOAD 方法如下：

1. 先打入 NEW 命令，把記憶體清除掉。
2. 打入 LIST 命令，檢查記憶體是否被清除了。
3. 再打入 LOAD — “暫時不要按 RETURN 鍵”。
4. 將磁帶倒轉至開始處。
5. 把錄音機計數器歸零。
6. 讓磁帶前進至 004。（因上例程式由 005 開始錄存）
7. 停止錄音機運轉。
8. 把錄音機上插於 “monitor” 之插頭拔下以便待會能聽到錄音機發出的聲音。
9. 按下錄音機 “play” 之按鍵，使錄音機開始轉動。
10. 如聽到刺耳尖銳的聲音表錄音機正常運轉，再將 “monitor” 之插頭插回 “monitor” 上。以便載入程式
11. 此時才按下電腦的 “RETURN” 鍵。
12. 此時游標（Cursor）會離開螢幕，電腦並發出 “嗶” 聲，表示程式開始被讀入記憶體內。
13. 程式讀完後，電腦會再發出一聲 “嗶” 聲，而且游標也會再出現在螢光幕上。
14. 按下錄音機 “停止” 鍵，停止錄音機之運轉；在電腦上可打入 RUN 或 LIST 之命令，程式將開始執行或被列於