

結構化COBOL程式設計

李惠明 編著

萬人出版社 出版

結構化COBOL程式設計

版權所有

請勿翻印

萬館卡電字第 81-1003 號

編著者：李 惠 明

出版者：萬 人 出 版 社 有 限 公 司

發行人：呂 芳 烈

電 話：3144895 · 3819425

地 址：台北市重慶南路一段41號

郵 撥：119410 萬人出版社帳戶

出版證：局版台業字第1882號

印刷者：啟盟印刷廠股份有限公司

台北市長泰街121巷10弄2號

中華民國七十年九月初版

中華民國七十一年九月增訂二版

中華民國七十二年九月增訂三版

中華民國七十三年八月增訂四版

定價 320 元

內政部著作權執照

著 人	著 物	著 作	著 人
李 志 明	李 志 明	李 志 明	李 志 明
萬 人 出 版 社 有 限 公 司	萬 人 出 版 社 有 限 公 司	萬 人 出 版 社 有 限 公 司	萬 人 出 版 社 有 限 公 司
台 內 著 字 第 壹 陸 捌 陸 肆 號	台 內 著 字 第 壹 陸 捌 陸 肆 號	台 內 著 字 第 壹 陸 捌 陸 肆 號	台 內 著 字 第 壹 陸 捌 陸 肆 號
有 效 年 限	有 效 年 限	有 效 年 限	有 效 年 限

中華民國

月

四

鄧永鄧創煥



教育部獎狀

李惠明先生參加七十二學
年度大學校院自然及應用科
學教學資料改進獎勵競賽，
作品經評審結果列為講義類
乙等獎，除發給獎金外，特
頒獎狀，用資激勵。

部長 朱 滄 白



中華民國

三月廿八日

目 錄

第一章	電子計算機系統與資料處理	1
1-1	電子計算機系統	1
1-2	數字系統	9
1-3	電子計算機作業概述	14
1-4	資料處理的意義	15
1-5	電子資料處理系統之作業程序	17
1-6	資料處理的工作	20
	習 題	22
第二章	結構化程式設計法則	23
2-1	結構化程式設計法則的目標	23
2-2	模組化程式設計	25
2-3	結構化程式設計	28
2-4	程式設計的步驟	32
	習 題	43
第三章	COBOL 程式語言基本認識	45
3-1	程式語言的沿革	45
3-2	COBOL 程式語言的優點與特性	46

3-3	COBOL程式語言可用的字元.....	47
3-4	COBOL使用字.....	48
3-5	COBOL 程式的組成.....	54
3-6	COBOL 寫碼紙.....	56
3-7	COBOL 程式的標點規則.....	59
3-8	COBOL 述句格式的表示法.....	60
3-9	COBOL 程式範例.....	60
	習 題.....	72
第四章	識別部與設備部.....	73
4-1	識別部.....	73
4-2	設備部.....	76
	習 題.....	86
第五章	資料的組織.....	87
5-1	資料的單位—檔、錄、欄.....	87
5-2	資料的型式.....	95
	習 題.....	97
第六章	資料部.....	99
6-1	檔節 FILE SECTION	99
6-2	工作儲存節WORKING-STORAGE SECTION	113
	習 題.....	117
第七章	程序部.....	119

7-1	OPEN敘述	121
7-2	READ敘述	124
7-3	MOVE 敘述	127
7-4	PERFORM 敘述	137
7-5	WRITE 敘述	139
7-6	CLOSE 敘述	142
7-7	STOP 敘述	143
7-8	實例研討	144
	習題	152
第八章	報表資料之編輯	153
8-1	編輯符號 Z	153
8-2	編輯符號 .	155
8-3	編輯符號 \$	158
8-4	編輯符號 ,	159
8-5	編輯符號 *	161
8-6	編輯符號 +	163
8-7	編輯符號 -	166
8-8	編輯符號 DB 與 CR	168
8-9	編輯符號 B 與 0	169
8-10	BLANK WHEN ZERO	171
	習題	172
第九章	算術運算	175
9-1	ADD 敘述	175

9-2	SUBTRACT 敘述	184
9-3	MULTIPLY 敘述	190
9-4	DIVIDE 敘述	194
9-5	COMPUTE 敘述	197
9-6	實例研討	201
	習題	222
第十章	IF 敘述	229
10-1	關係條件	230
10-2	巢狀 IF 敘述	234
10-3	正負條件句	242
10-4	類別條件句	243
10-5	複合條件句	245
10-6	條件名稱條件	249
10-7	實例研討	252
	習題	302
第十一章	PERFORM 敘述	309
11-1	PERFORM 敘述	309
11-2	實例研討	327
	習題	331
第十二章	表格處理	331
12-1	REDEFINES 子句	334
12-2	OCCURS 子句	336

12-3	存固定值的表格處理.....	345
12-4	指標.....	349
12-5	指標的設定.....	350
12-6	表格尋找.....	354
12-7	實例研討.....	363
	習題.....	392
第十三章	DISPLAY 與ACCEPT 敘述.....	401
13-1	DISPLAY 敘述.....	401
13-2	ACCEPT 敘述.....	404
13-3	ACCEPT 敘述與DISPLAY 敘述的聯合運用.....	407
	習題.....	409
第十四章	其他控制轉移敘述.....	411
14-1	GO TO 敘述.....	411
14-2	EXIT 敘述.....	414
14-3	GO TO.....DEPENDING ON.....敘述.....	417
	習題.....	421
附錄A	DATAPONINT 電腦系統顯像端末機及公用程式操作說明.....	423
附錄B	完整的ANSI COBOL 語言格式.....	431

附錄 C	TAB501 型打驗卡機操作說明	469
C - 1		488
C - 2		491
附錄 D	電腦系統的字元碼	492
附錄 E	CDC 電腦系統顯像端末機操作程序	495
附錄 F	PRIME 電腦系統顯像端末機操作程序	503
附錄 G	VAX 電腦系統顯像端末機操作程序	508

第一章 電子計算機系統與資料處理

1-1 電子計算機系統

一部電子計算機系統係由硬體與軟體所組成。硬體係指看得見摸得著的設備，亦即電子計算機系統本身的機器設備。軟體則包括指揮命令電子計算機設備運轉、執行作業的各種指令、程式。

(一) 硬體與軟體

組成一部電子計算機的實體裝置，稱為電子計算機的**硬體**。它可分成五個部門，即**輸入部門**、**儲存（記憶）部門**、**算術及邏輯部門**、**控制部門**以及**輸出部門**。

1. 輸入部門

輸入部門（Input Unit）為將資料（Data）及指令（Instruction）輸入電子計算機處理之單位，通常這些資料或指令可經由卡片（Card）、磁帶（Magnetic Tape）、磁碟（Disk）、紙帶、控制台上的打字機等媒體送入電子計算機。

2. 儲存（記憶）部門

儲存（記憶）部門（Storage or Memory Unit），主要的功用是儲存資料、指令與程式及運算結果之單位。

3. 算術及邏輯部門

算術及邏輯部門(Arithmetic / Logical Unit)，為執行各種算術運算(包括加、減、乘、除等運算) 以及邏輯運算(包括比較、合併、選擇以及邏輯決定) 之單位。

4. 控制部門

控制部門(Control Unit) 為電子計算機的中樞，凡程式指令之排列次序，以及上述各部門與輸出部門之任何作業之督導與推動，均由本部門控制。

5. 輸出部門

輸出部門(Output Unit)，為電子計算機供給運算或處理後的結果輸出之單位。

此五部門相互關係之結構方塊圖如圖 1 - 1 所示。

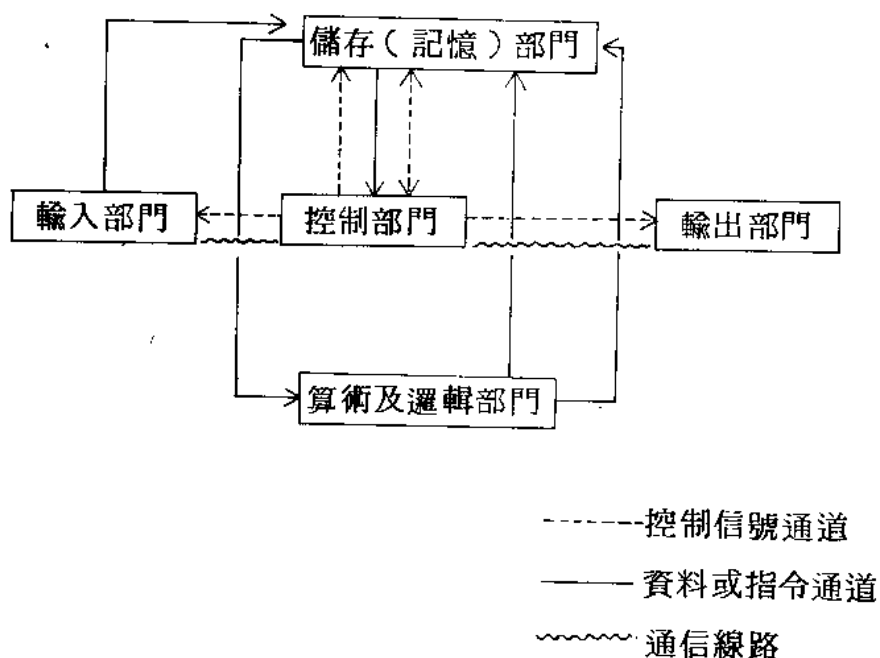


圖 1 - 1 電子計算機之結構方塊圖

其中控制部門、算術邏輯部門、儲存記憶部門合稱為**中央處理部門**(Central Processing Unit , 簡寫為CPU) 或稱**中央處理機**。

電子計算機系統之硬體示如圖 1 - 2 , 圖 1 - 3 。

在圖 1 - 3 裏的電子計算機系統有輸入部門(如讀卡機) , 中央處理機(CPU) 以及輸出部門(如列表機) , 還包括外部(輔助) 儲存部門如磁帶機或磁碟機。在一個典型的資料處理工作, 資料係經由輸入部門(或自外部儲存部門) 輸入中央處理機以執行計算或其他處理。在處理完畢之後的結果則經由輸出部門(如印表機) 輸送出來, 或者存入外部儲存部門供爾後的處理或輸出。

電子計算機硬體能執行讀、寫、計算等功能, 但是這些功能執行的順序以及輸出入設備的使用等是由儲存在電子計算機記憶體裏的一組指令來指揮的。這些電子計算機處理指令一般就叫做軟體。這些指令組成許多組叫做常規(Routines) 及程式(Programs) 。一個常規係指一組指令執行某一特定的任務, 例如計算一個數值的平方根, 或是當輸入資料錯誤時產生一項錯誤訊息。一個程式則包含一個或數個常規用以解決一個完整的問題。

軟體有數種類型, 與COBOL的學習特別有關的軟體有三種: 應用程式、編譯程式, 和作業系統。

1. 應用程式 :

電子計算機使用人為了利用電子計算機解決某一問題所撰寫的程式叫做應用程式。例如某公司為處理員工薪資財務會計、庫存盤點等, 以及本書的一些範例程式, 都是應用程式。應用程式可以由電子計算機中心的人員自己撰寫, 或向軟體公司購買。軟體公司常能提供一一些常用的應用程式組。

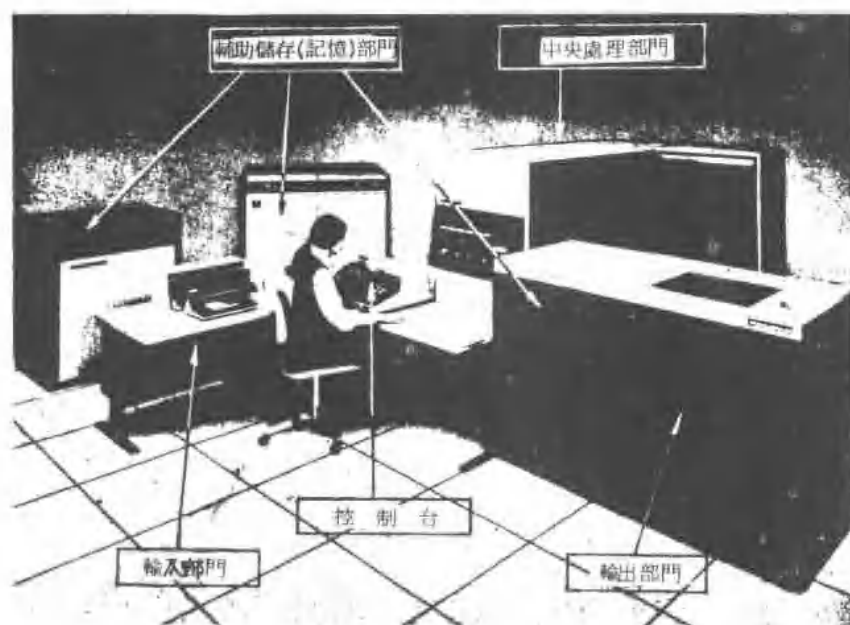


圖 1 - 2

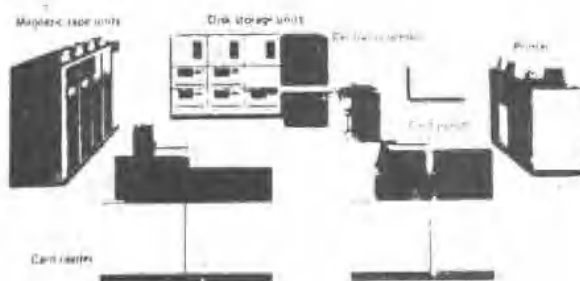


圖 1 - 3

2. 編譯程式：

利用高階程式語言所撰寫的程式，例如COBOL程式，必先利用編譯程式把它編譯成機器語言之後，電子計算機才能夠據以執行。這種編譯程式大都由電子計算機製造廠商所提供。

3. 作業系統：

指示和管理一部電子計算機系統作業的程式組叫做作業系統。應用程式就是在作業系統的指揮和支援之下作業的。例如有一個應用程式要自讀卡機讀入一張卡片，而讀卡機無法作業時，作業系統會送一個適當的訊息給電子計算機操作員。作業系統通常是由電子計算機製造廠家所提供的。

正在執行的程式是存放在內部儲存體(Internal storage)或稱為記憶體(Memory)內，不是正在執行的程式則存放在外部(輔助)儲存體裏。作業系統亦是存放在外部儲存體裏，而其中最常用的控制程式則常駐在內部儲存體內。這些控制程式根據工作控制陳述的要求把所需要的常規和程式自外部儲存體讀進來並安排他們的執行。這些執行的程式可能是編譯程式、應用程式、或其他軟體。

(二) 程式語言種類

電子計算機程式語言有很多種類，例如：由一些特殊數碼及數字所組成，是電子計算機系統的機器裝置直接認識之語言——**機器語言**(Machine Language)；以記號記述的**組合語言**(Assembly Language)又稱**符號語言**(Symbolic Language)；和我們日常使用的語言或計算式的方式來表示，以編寫程式，為一種很方便的程式語言，這種語言稱為**高階語言**(High Level Language)或稱**編譯語言**(Compiler Language)等三種。

1. 機器語言

電子計算機直接能理解的程式語言，即是機器語言。這種語言形式上是由一串串的數字符號所組成的，是電子計算機內部執行程式對實際所使用的語言。數字符號的個數，完全視電子計算機所用的數字系統而定。在二進位制是為“0”及“1”兩個符號；八進位制是“0”至“7”八個符號。每一串的組合代表一個指示電子計算機作一項最基本工作的指令。

由於機器語言係由一大堆吾人無法了解的意義與文字所組成，因此，對人類而言是一種既煩且雜而很難處理的語言。同時，機器語言很難記憶，在編寫程式時，需化費很多時間與勞力。電子計算機剛發明出來的時候，程式設計人員必須用機器語言來寫程式，但是以機器語言寫成的程式，不易做部份的刪除或追加，處理也很複雜，又檢查程式的內容也不容易，因此現在甚少使用。

2. 組合語言

由於機器語言完全是由一串串數字符號所組成，對電子計算機來說固然是方便極了，但是對使用者而言，却是很麻煩的事，不但要記住代表指令的數字符號，而且在寫程式時，指令和資料所在的位址也需要程式設計人員來安排，實在是一件很煩雜而且易錯的工作，因此，機器語言僅在第一代電子計算機中被使用。到了1950年代第二代電子計算機產生後，以英文字母來代表作業碼，位址可由撰寫程式的人選用適當的符號來代替的組合語言（Assembly Language）乃應運而生。利用此種組合語言寫程式較機器語言為容易。

但是用組合語言寫成的程式，在讀入電子計算機後，電子計

算機並不能直接認識它，是要靠預先儲存在機器內的組合格式（Assembler），將它編譯成機器語言，並自動分配整個程式中用到的指令和資料的位址，以及預留一些位址以儲存運算結果。使用組合語言寫程式，資料儲存位置的長度，也可以由程式設計人員自行擬定。

由於機器之不同，組合語言也不同，因此不易普遍使用。即是組合語言是隨著電子計算機之廠牌不同而異，像CDC CYBER機型的組合語言叫COMPASS；DATAPOINT機型，IBM 360及370機型就叫ASSEMBLER，因此不易普遍使用。

3. 高階語言

機器語言及組合語言與電子計算機內部之構造有直接關係，這兩種語言之表示方法與編寫方式是隨著電子計算機的不同而不同。使用者欲學好這兩種語言，必須對機器之內部組織要有充分之了解，對使用者而言極其不便。為了使電子計算機能被廣泛的使用，需要更方便且容易學習的程式語言提供運用。這種語言既可以完全採取與我們日常使用的語言與計算方式相似的方式來表示，又可以讓使用電子計算機系統的人無需十分了解內部之動作原理亦能運用自如的程式語言，這種語言系統通常稱為高階語言（High Level Language），或稱編譯語言（Compiler Language）。

一種高階程式語言是問題解法導向或處理程序導向的，而不是某一種電子計算機的機器層面指令。高階語言的指令敘述所使用的字元與符號和我們習用的解法描述或處理程序相類似。高階語言指令和符號組合指令的另外一項主要區別是一個高階語言陳述編譯成好幾個機器語言指令。