

计算机百科

[日] 土居範久 笈捷彦编
邱建设 译

操作系统 35

- 命令 255
- 任务 292
- 信号量 409
- 虚拟存储器 411
- 文件 391
 - 数据库 330
 - 存取 93
 - 系统程序 402
 - 实用程序 317
 - 编辑程序 11
 - 分类, 合并 131
 - 应用程序 427
 - 程序库 69
- MS-DOS 258
- CP/M 43
- UNIX 364
- TRON 354
- 语言加工程序 444
 - 编译程序 16
 - 波兰表示法 23
 - 汇编程序 157
 - 目标码 272

人工智能 289

- 模式识别 267
- 知识工程 460
- 知识表示 456
- 专家系统 472
- 机器翻译 164

数学

- 算法 343
- 代数 100
- 排错 276
- 逻辑 239
- 数据结构 321
- 无限 398
- 数据类型 334
- 计算机科学 201
- 图灵机 360
- 软件工程 302
- 程序设计方法学 70
- 程序范例 62

程序语言 78

- 文法 387
- 程序 51
- 程序单位 54
 - 自变量 483
 - 递归 111
- LISP 237
- MODULA-2 263
- Pascal 280
- Prolog 282
- Smalltalk 315
- 简易语言 223
- Ada 1
- APL 3
- BASIC 9
- C 25
- COBOL 41
- dBASE II, III 115
- FORTRAN 142

应用领域

- 公式处理 149
- 数值计算 339
- 模拟 265
- 随机数 345
- 计算机图形学 203
- CAD, CAM 27
- 排字计算机化 278
- 医学图象技术 423
- 医疗信息
- 机器
- 自动化
- CAI
- 家庭自动

前 言

“我也想学计算机，不过……”，后面想说什么呢，追根刨底地寻问，才回答：“因为不懂专业用语……”。可以说，这些话不是没有道理的。只要一打开书学习，首先碰到的，就是计算机，后面呢，哎呀，又是什么操作系统啦，什么命令啦，什么编辑啦，什么窗口啦，各种专业用语简直多如牛毛。对于本来就有些担心，又加上没有经验的人来说，无疑，是在忍受着一个又一个的盘问。

说专业用语太多，以至不能继续学下去，无论如何，也是不过分的。如果有一本书，心平气和地读下去，越读越觉得“啊，原来如此！”，那该多好啊，本“百科”就是出于这样的目的，而编写的。

所以，一开始学习计算机的时候，就要准备从收集耳闻目睹的用语作起。已经有个人计算机的同志，大概也会碰到这样的用语，也要收集一下这些用语。

如何说明这些用语呢？这倒是个难题。即使把这许多用语一条一条地象辞典一样排列起来，仍然不能很好地给以说明。反倒是通过这些用语的互相参照，从它们之间的互相联系中，来学习，效果要好得多。所以，本书采用了，精选一些词条，对其中有关的加以参照说明的方式。

都列举有哪些词条呢，由于本书设计了一幅关连图，如果，一边看此图，一边从所发现的不明白的或感兴趣的地方读起，倒是一个好办法。

当然，也可象辞典那样使用，书后附有索引，所以，查到了用语，再看它是属于哪个词条，由此读去，这也是一种使用方法。

现在，计算机已渗透到各个领域。出现了许多与周围领域密切相关的用语。编写这样的百科，当然要求助于本百科的各位执笔者，请了各个领域的许多有关专家，分别担任有关词条的编写任务。

对各位专家写好的文章，编者又以读者的身份重新阅读，尽管篇幅有限，仍提出了不少要求：这里再强调一下，那里再加一点……，等等，有些地方又请专家们改换了写法。本来是请各位专家从自己的专

业角度来写的，却又提出了不少苛刻要求，现在回想起来，深感歉意。

尽管有不少无理的要求，执笔者各位，仍很快一一答应下来，这里再次表示万分感谢。另外，对阅读了一部分原稿的高泽嘉光先生，以及提供了各种照片的各贵公司，深表谢意。谢谢各位的支持和帮助。

如果这本百科，既没有偏离各位执笔者的本意，又通俗易懂，我们将感到无尚欣慰。最后，希望各位读者，对这本百科，多提宝贵意见。

1987年10月

土居 范久

笈 捷彦

译序

译
序

本书是1987年12月在日本首次出版的最新畅销书，日本称它是一本划时代的计算机事典。它是由日本24位知名的专家、教授集体纂写而成的，囊括几乎计算机的所有领域的基础知识，取材广泛，图文并茂，立意新颖，构思巧妙，概念准确，深入浅出，富有哲理。特别是作者打破了对辞书孤立地罗列一大堆词条的方法，别具新裁地设计了一幅词条关连图，构画出了凌乱的各词条的内部联系，起到画龙点睛，纲举目张的效果。在具体阐述各词条时，采取各种手段，始终贯彻将各词条互相参照、联系起来学习的辩证思想，使本书兼有计算机事典和获取系统的基础知识的学习参考书两重作用，既可作为初学者的入门书，又利于专业工作者借鉴。

由于计算机在各个领域的应用日趋广泛，本书从中精选了最基本、重要的135条词条，给以阐述。概念的变革推动计算机的发展。有些词条是国内少见的，象“TRON”、“工作站”、“生物计算机”，有的老词条加进了新的研究内容，反映了国际上计算机的新动态。读者可按词条关连图，有选择地学习生疏的词条，可系统地得到计算机的基础知识。通过书后的索引，又可将本书作为计算机的最新辞典来使用。

随着我国计算机事业的蓬勃发展，既要求人们迅速掌握大量的新知识、新理论、新成就和新应用；同时也要求有关人员在从事专题研究的过程中，十分重视综合性的研究和学习。但愿本书对读者有所裨益。

最后感谢武汉大学软件工程研究所领导的支持，感谢为本书的出版而付出辛勤劳动的所有同志的热情支持和帮助。

由于时间仓促，限于译者水平，加之题材广泛，不妥之处在所难免，敬请读者不吝批评指正。

译者

1989年1月，于武汉大学

执笔者一览表

(五十音顺序)

飯沼 武	放射线医学综合研究所临床研究部
伊藤 崧	都立乌山工业专科学校
石原 聪	光产业技术振兴学会
上田和纪	新一代计算机技术开发机构第一研究室
大野义夫	庆应义塾大学情报科学研究所
寛 捷彦	早稻田大学情报科学研究教育中心
神沼二真	东京临床医学综合研究所生命信息工程研究室
上林弥彦	九州大学工程系信息工程专业
坂村 健	东京大学理学系信息科学专业
佐佐木建昭	理化学研究所信息科学研究室
白井良明	电子技术综合研究所控制部
竹内寿一郎	庆应义塾大学理学系管理工程专业
辻井润一	京都大学工学系第二电工教研室
寺田 实	东京大学工学系计算工程专业
土居范久	庆应义塾大学信息科学研究所
中岛秀之	电子技术综合研究所人机系统研究室
萩谷昌巳	京都大学数理分析研究所
前田英明	文教大学信息系经营信息专业
松原 仁	电子技术综合研究中心信息系系统研究室
松本 元	电子技术综合研究所计算机部模拟信息研究室
村井 纯	东京大学大型计算机中心
山崎利治	日本通用有限公司技术研究部
山本喜一	庆应义塾大学信息科学研究所
吉村 启	庆应义塾基础部

本百科的组成和使用方法

词条选法

本书选择了计算机领域中经常使用的基本用语。并且，为了便于理解计算机在各个领域的应用，也包括了与应用、实用有关的用语，总共有135条词条。

词条排列

词条，按汉语拼音音序排列。

词条目录

词条目录，有按汉语拼音的中文目录一覽，和词条关连图2种。词条关连图，对词条之间的联系，一目了然，具有以下特点。

(1) 词条关连图，画在左右两页合在一起的版面上，左页是与硬件有关的词条，右页是与软件有关的词条，所以从词条的位置，便容易了解它的意义。

(2) 特别重要的12个词条，用较大的楷体来表示。这里，收集了便于初学者容易理解计算机的基本轮廓，所必须具备的最起码的知识。在中央靠上面的“计算机”，是最基础的词条。

(3) 从“计算机”这条词条出发，按照箭头，寻找解说的展开次序。依着箭头继续阅读各个词条，就会系统的获得计算机的基础知识，对于没有带箭头的词条，若读了用黑体字表示的重要用语，也能充分理解。

词条标题

词条标题形式如下：

词条名……	软件
对应的英语名……	software
最好是先读这一词条……	←计算机
关系密切的词条……	↔硬件

词条解说

解说，首先，是揭示这条词语的定义。由于定义有些简练，可能初学者立即理解有困难。这种情况，由于接着就是通俗、具体的解说，请读者继续读下去。

文中，凡是右上角带有星号^{*}标记的用语，为135条词条之一，请继续参照详细的解说。

索 引

另外，象“……光笔（→定标器）……”之类的说明中，括号中有箭头的时候，表示光笔的解说，在“定标器”这一词条中也有说明。

书后附有汉语拼音顺序的重要专业用语的索引。135条基本词条里没有的用语，请查一下索引。英语、略语是按abc顺序排列的，也附有索引表。

目 录

目
录

前 言.....	I
译 序.....	III
汉语拼音音序词条一览.....	VIII
词条关连图.....	X
本百科的组成和使用方法.....	XII
百科词条.....	1
索 引.....	495

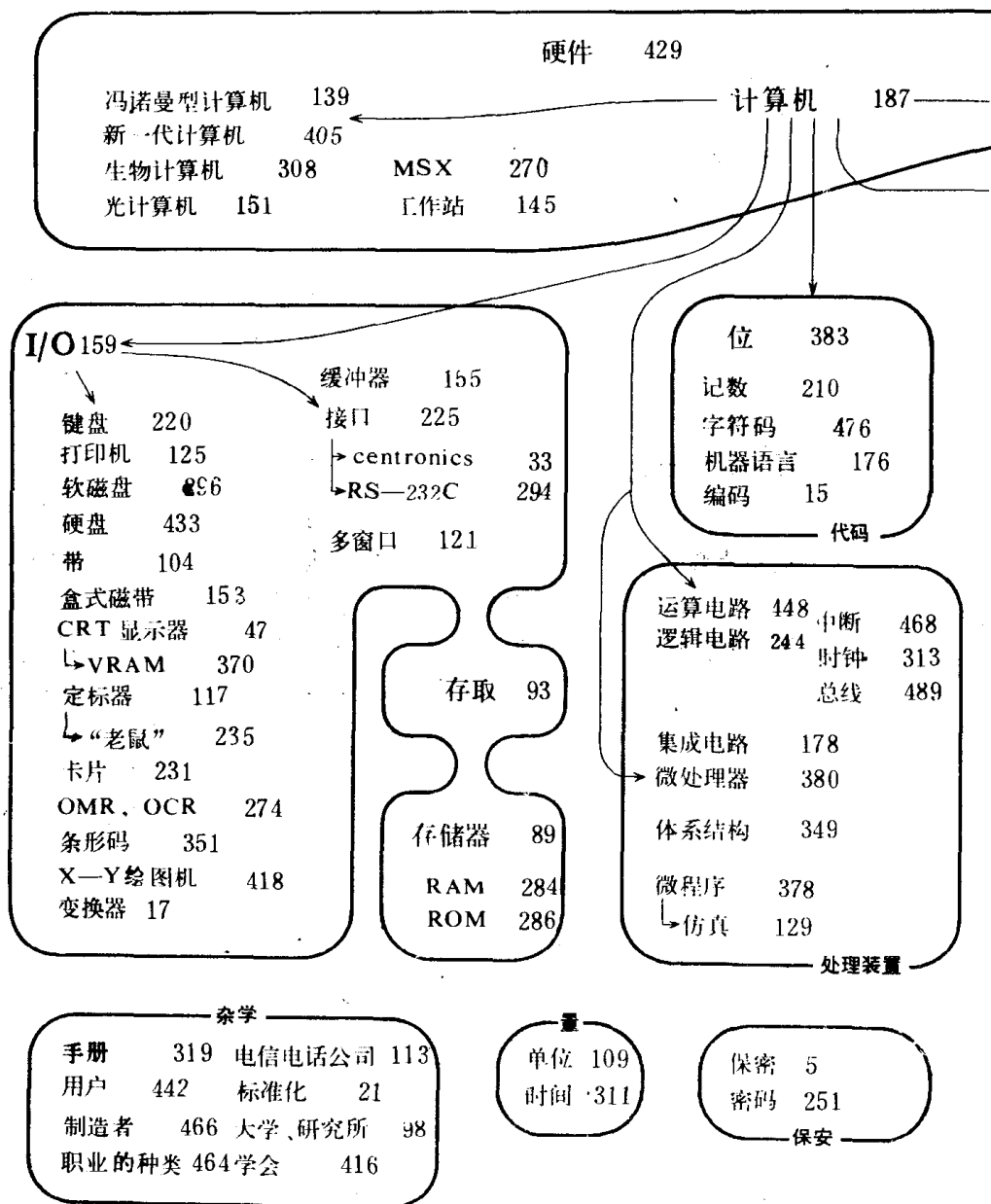
汉语拼音音序词条一览

A	Ada	1	存取	93
	APL	3		
B	保密	5	D 打印机	125
	BASIC	9	大学, 研究所	98
	编辑程序	11	代数	100
	编码	15	带	104
	编译程序	16	单位	109
	变换器	17	递归	111
	标准化	21	电信电话公司	113
	波兰表示法	23	dBAS II、III	115
C	C	25	定标器	117
	CAD、CAM	27	多窗口	121
	CAI	31	F 仿真	129
	centronics	33	分类, 合并	131
	操作系统	35	冯·诺曼型计算机	139
	COBOL	41	FORTAN	142
	CP/M	43	G 工作站	145
	CRT显示器	47	公式处理	149
	程序	51	光计算机	151
	程序单位	54	H 盒式磁带	153
	程序范例	62	缓冲器	155
	程序库	69	汇编程序	157
	程序设计方法学	70	I I/O	159
	程序语言	78		
	处理方式	82		
	存储器	89		

J	机器翻译	164	P	排错	276
	机器人	172		排字计算机化	278
	机器语言	176		Pascal	280
	集成电路	178		Prolog	282
	计算机	187	R	RAM	284
	计算机科学	201		ROM	286
	计算机图形学	203		人工智能	289
	记数	210		任务	292
	家庭自动化	218		RS-232C	294
	键盘	220		软磁盘	296
	简易语言	223		软件	299
	接口	225		软件工程	302
K	卡片	231	S	生物计算机	308
L	“老鼠”	235		时间	311
	LISP	237		时钟	313
	逻辑	239		Smalltalk	315
	逻辑电路	244		实用程序	317
M	密码	251		手册	319
	命令	255		数据结构	321
	MOS-DOS	258		数据库	330
	MODULA-2	263		数据类型	334
	模拟	265		数值计算	339
	模式识别	267		算法	343
	MSX	270		随机数	345
	目标码	272	T	体系结构	349
O	OMR, OCR	274		条形码	351
				TRON	354
				图灵机	360

U	UNIX	364	Y	医疗信息处理	420
				医学图象技术	423
V	VRAM	370		应用程序	427
				硬件	429
W	网络	372		硬盘	433
	微程序	378		用户	442
	微处理器	380		语言加工程序	444
	位	383		运算电路	448
	文法	387			
	文件	391	Z	知识表示	456
	无限	398		知识工程	460
				职业的种类	464
X	系统程序	402		制造者	466
	新一代计算机	405		中断	468
	信号量	409		专家系统	472
	虚拟存储器	411		字符码	476
	学会	416		自变量	483
	X-Y绘图机	418		自动化	487
				总线	489

词 条 关 连 图



软件 299

处理方式 82

网络 372

人工智能 289

→ 模式识别 267
→ 知识工程 460
→ 知识表示 456
→ 专家系统 472
→ 机器翻译 164

操作系统 35

→ 命令 255
→ 任务 252
→ 信号量 409
→ 虚拟存储器 411
→ 文件 1391
 → 数据库 330
 → 存取 93
→ 系统程序 402
 → 实用程序 317
 → 编辑程序 11
 → 分类, 合并 131
→ 应用程序 427
→ 程序库 69

MS-DOS 258
CP/M 43
UNIX 364
TRON 354

语言加工程序 444
 → 编译程序 16
 → 波兰表示法 23
 → 汇编程序 157
 → 目标码 272

背景

算法 343
排错 276
数据结构 321
数据类型 334
计算机科学 201
图灵机 360
软件工程 302
 → 程序设计方法学 70
程序范例 62

代数 100
逻辑 239
无限 398

程序语言 78

→ 文法 387
→ 程序 51
→ 程序单位 54
 → 自变量 483
 → 递归 111
LISP 237
MODULA-2 263
Pascal 280
Prolog 282
Smalltalk 315
简易语言 223

→ Ada 1
→ APL 3
BASIC 9
C 25
COBOL 41
dBASE II, III 115
FORTRAN 142

应用领域

公式处理 149
数值计算 339
模拟 265
随机数 345
计算机图形学 203
CAD, CAM 27
排字计算机化 278
医学图象技术 423
医疗信息处理 420
机器人 172
自动化 487
CAI 31
家庭自动化 218

Ada

← 程序语言

美国国防部，花了近十年时间设计的程序语言，用于国防系统计算机的软件开发。

国防部开始调查国防部范围内必需的语言功能，希望能满足这种确定的设计说明书的功能，把此种语言作为共同的语言。但现存的语言都不能满足这一设计说明书。因此，让很多公司同时设计这样的语言，从中选定一种最适合的语言。后来确定以尹其比亚 (J. Ichblah) 的小组为中心来进行改进。1982年即确定了现在的Ada语言。

这不仅是美国国内的标准，而且已变为国际上的标准了。

它不仅具有Pascal语言的程序结构、数据结构的各种优点，而且引入了叫**程序包 (Package)**的程序单位，相互有关连的数据、过程、函数都一起放在某个程序包，这种程序单位中，经常使用的程序部分，预先放在这样的程序中，然后再把它们组合起来，这样，有利于设计大型程序。而且，并不需要将它们分成好几个数据类型，Ada中，有一叫generic的宏 (→语言加工程序) 样板机能。

对不同类型的数据也能使用同样运算符或过程名或函数名，利用已定义的类型来简单设计新的其他的数据类型。这些都比Pascal的灵活性大。

设计系统时，除了通常的处理之外，例外的事情出现时也应有的相应的处理。Ada有明确地描述这种**例外处理**的机能。

另外，也具有与并行处理或与硬件相关连的细微部分的控制功能。

Ada是PL/I (→程序语言) 之后，目前的大型程序语言。特别是和Ada语言同时设计的名叫APSE的专用程序环境也是很少见的。

Ada[®]是美国国防部的登录商标。

```
-- seiretu
with TEXT_IO; use TEXT_IO;
procedure SEIRETU is
  A : array(0..100) of FLOAT;
  N : INTEGER;
  package F is
    new FLOAT_IO.FLOAT;
  package G is
    new INTEGER_IO.INTEGER;
  use F,G;
begin
  get(N);
  for I in 1..N loop
    get(A(I));
  end loop;
  for I in 2..N loop
    block
      X : FLOAT := A(I);
      J : INTEGER := I-1;
      begin
        A(0) := X;
        while X > A(J) loop
          A(J+1) := A(J);
          J := J-1;
        end loop;
        A(J+1) := X;
      end;
    end loop;
  for I in 1..N loop
    put(A(I));
  end loop;
end;
```

图1 由Ada写的单纯插入法的程序

← 程序语言

由美国的爱巴颂 (K. E. Iverson) 提出的程序语言^{*}。当初，目的是为严密而简洁地表示数学的逻辑和运算，所以并不是开发出来作为计算机用的语言。1962年，第“爱巴颂语言”之大成的名著“A Programming Language”出版了，之后，就以这本书名的每个字的第一个字母，取名叫APL。从此，开始尝试把APL作为程序语言来使用，1966年底，在IBM公司使用，1968年，正式的APL/360逐渐开始实用化。1973年，能够在文件空间用户区，它们之间传递信息，完成了APL/VS，这就是APL的前景。即使在IBM公司以外，APL也相继被开发出来，已能在很多主机（→制造者）上使用。另一方面，自从1974年加拿大的MCM公司，在个人计算机上移植成功APL以来，今天，各种个人计算机上，都已能使用。甚至最近，还有数组处理机的APL机器，及编译程序^{*}等也都发表了，处理速度大大提高。

APL在工作时，提供一个方便的做法叫做工作区的概念，它是类似于笔记本用纸的作用。一项工作的数据和程序（叫用户定义区^{*}），全部放到一个工作区里，实行分工作区管理。因此，对几个工作中中断或又开始的情况，也能开展工作。这样的管理比较容易，但是，在必须有大的工作区的时候，只这个问题就令使用者负担加重，若同时使用好几个，反应就很慢，这是它的缺点。把用户区的数据，变换成一个数据，进行处理，或反之把读进来的数据，作为用户定义函数登录，具有这样方便的功能，也是它的一大特长。近年来，也开发出了处理对象为具有复杂结构的矩阵的APL，这种矩阵或向量允许抽象数组作为矩阵的一个元素。

APL语言的语法非常简单，语句全都用函数组成。运算，和别的语言不同，是从右往左进行的。例如，正和负交替出现的式子 $\sum_{i=0}^n (-1)^i$

1) $\sum_{i=1}^N 1/i!$ ，写成下式

$$1 - (\div! 1) - (\div! 2) - (\div! 3) - \dots - (\div! N)$$

把它整理后，即使写成下式也可以。

$$- / \div! - 1 + 1/N + 1$$

到N的质数，可用下式求出，

$$(\sim I \in I_0. \times I) / I \leftarrow 1 \downarrow 1/N$$

$N \times N$ (N 为奇数)的魔方阵为

$$R \theta (R \leftarrow (1/N) - (N+1) \div 2) \phi (N, N) \rho 1/N^* 2$$

或使用把次数并列起来的矢量 I ，

$$[*] [1 + (\phi I [/ Y) 0. \leq Y]$$

用上式能做出直方图。这样，不使用循环，只用一行就能表达。

由于在APL中有很多特殊的符号，通常的键盘并不能与APL对应，程序难读，并且I/O文件和图形函数也没有标准化，所以，并未获得广泛使用。然而，随着个人计算机的普及和发展，会使APL能随时自由地使用，如果克服了这些缺点，因为它有简洁，强有力的机能，相信一定会有更多的人使用它。