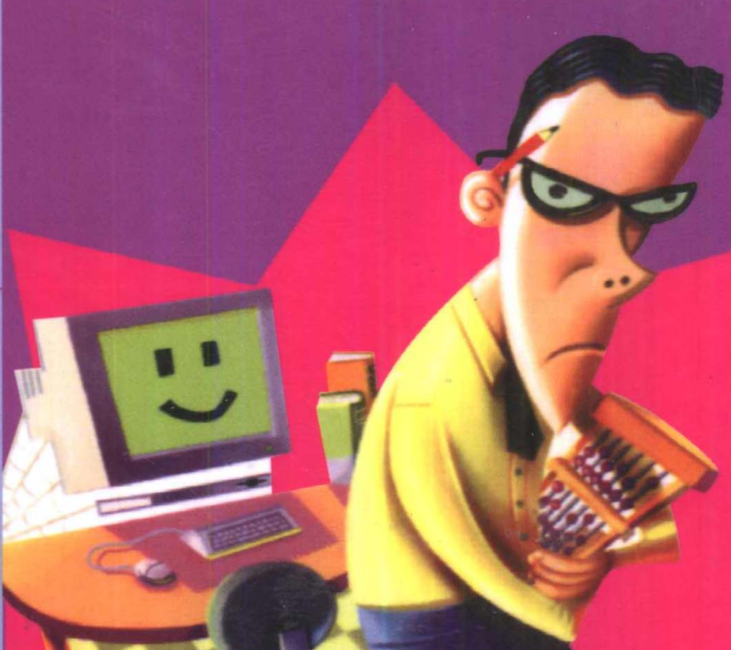
ENGLISH FOR COMPUTER SCIENCE & TECHNOLOGY

计算机科技英语

选读



Reading 英文系列

外文出版社

150

English for Computer Science & Technology

计算机科技英语选读

何兆枢　编著

Foreign Languages Press
外文出版社

图书在版编目(CIP)数据

计算机科技英语选读/何兆枢编著 .—北京:外文出版社,2001.8

ISBN 7-119-02243-1

Ⅰ.计… Ⅱ.何… Ⅲ.电子计算机-英语-对照读物-英、汉 Ⅳ.H319.4:TP

中国版本图书馆 CIP 数据核字(2001)第 059617 号

外文出版社网址:
http://www.flp.com.cn
外文出版社电子信箱:
info@flp.com.cn
sales@flp.com.cn

计算机科技英语选读

作　　者	何兆枢		
责任编辑	蔡　箐　李　湲		
封面设计	吴　涛		
印刷监制	冯　浩		
出版发行	外文出版社		
社　　址	北京市百万庄大街 24 号	邮政编码	100037
电　　话	(010)68996075/68995883(编辑部) (010)68329514/68327211(推广发行部)		
印　　刷	北京顺义振华印刷厂		
经　　销	新华书店/外文书店		
开　　本	大 32 开	字　　数	110 千字
印　　数	0001—5000 册	印　　张	10.75
版　　次	2002 年第 1 版第 1 次印刷		
装　　别	平		
书　　号	ISBN 7-119-02243-1/H·1206(外)		
定　　价	15.00 元		

前　　言

在多年从事计算机专业英语的教学工作中，我深刻地认识到计算机科技发展是如此迅速，以致计算机科技英语的教材需要不断地更新，补充一些新内容，才能满足大专院校的学生以及广大工程技术人员学习计算机专业的需要。本书就是在这种思想指导下编写的计算机科技英语课程的适用教材，内容包括计算机科技、应用、产品、市场、网络等最新的信息，反映20世纪90年代后期和21世纪初计算机技术最新发展的题材。它也包括计算机科技的“经典”题材。在某种意义上说，它是用英语和汉语写成的计算机专业基础课的浓缩教科书。就专业知识而言，它可以作为大专院校学生、优秀高中学生、计算机爱好者和“网迷”们学习计算机科技知识和英语的教材，以拓宽知识面、提高阅读计算机专业英语科技文献的能力和水平。由于内容丰富、题材新颖、理论结合实际、深入浅出、英汉对照等优点，此书将适合广泛的读者群的需要。在信息化时代，各类专业科技人员、企业管理人员、市场营销人员和决策者也需要阅读和学习这类书籍，以提高英语阅读能力，才可以利用环球媒体(包括商业网络、广告)提供的知识和商情及时地把握国际上信息产业发展的时代脉搏，“夺人先机”，适应21世纪工作的要求。

本书一些课文题材参考北京理工大学计算机系教授龚元明等所编著的“计算机科学与技术”，清华大学研究生黄辉撰写第13、14课课文(中文)。

编著者 **何兆枢**

北京理工大学计算机系

CONTENTS
目　　录

Lesson 1

An Introduction of Digital Computers

The von Neumann Architecture

During World War II, in the United States, the need for fast calculating machines for producing projectile table for every type of guns and cannons and for simulation required in the design of the first atomic bomb called for automatic computing machines—digital computers. John Mauchly and J. Presper Eckert developed the first real digital computer, known as the ENIAC. Their work was expanded upon and refined by John von Neumann of the Princeton University.

ENIAC, made in a laboratory of the Northwest University in Evanston, Ill., in 1942, was composed of more than 30 thousand electronic tubes and nearly one million pieces of other components such as resistors, condensers and switches. It occupied three large rooms and consumed electricity that would otherwise be enough to power a small city with 30 thousand residents. To dissipate the heat emanated from inside the machine, powerful ventilators were used. The operating speed was very low, about 2000 operations per second.

Yet, this computer ushered in the Computer Age. It was an unprecedented machine capable of doing work involving human mental process, for example, simple logic inference, basic arithmetic calculations and data processing. Particularly, its structure known as the von Neumann architecture and its working principle known as stored program concept proved to be universally accepted for digital computers built for performing actually all types of computations. Many great achievements have since been made in computer hardware, but even today's computers, ranging from the largest mainframes to the smallest personal computers, use some variations of that idea. Experts predict that the concept is still valid and it will continue to be used for years to come.

All von Neumann type digital computers are consisted of four basic components-the memory, the arithmetic-logic unit (ALU), the control unit and the input/output (I/O) unit. The ALU and control unit are usually considered to be an integral component and it is generally referred to as the central processing unit (CPU). Most microprocessors are nothing more than a single chip of integrated circuit CPU.

Digital computers operate on the von Neumann's stored program concept. A program is consisted of a series of instructions written in a specific code, for example, binary numbers—0 and 1, assembly code, high-level computer programming languages, and arranged in an order to tell the computer specific operations, for example, moving binary numbers into and out of the memory and between various registers and the CPU, and a variety of arithmetic and logic operations. The sequence of instructions arranged in a step-by-step way constitutes a procedure called an algorithm. A program will comprise a series of algorithms for solving some specific problems. The program is stored in the digital memory. Each instruction is given an address code in the memory. Usually, they are arranged in an incremental order of address. An instruction counter is used in the control unit to record the address code of the instruction currently under examination. By making an increment on the address code in the address counter, the control unit turns to the next instruction to be fetched. After the instruction is fetched, its operation code is decoded to produce a signal, which is then sent to a proper unit, and the operation can begin. The operands can be found in an address called the "initial address" . The result of this operation is sent to the cells indicated by the "destination address".

Special instructions called the transfer instructions are used when the control needs to leave the current loop of operations and turn to another set of instructions. Among them, the unconditional transfer instruction always causes a modification done to the instruction counter and a transfer to the next address. The conditional transfer instruction may cause different types of modifications depending on the result of some previous calculations specified in the conditional transfer instruction.

This description tells why the von Neumann type digital computers can work automatically to perform various arithmetic and logical computations and operations, following exactly what a stored program intends to do.

Development of Computer Hardware

The hardware of digital computers has undergone a series of revolutionary changes. The gain in the working speed and function has been impressive. The first transistor was invented in the Bell laboratory in 1948. Computer manufacturers soon used this new contraption to improve their computers with outstanding merits-small volume, light weight, little power consumption, long service hours and high working speed. With transistors replacing electronic tubes, functional circuits could be made on the print circuit boards which can be assembled on sockets, greatly conveniencing manufacturing and maintenance. Coupled with transistors, other components were miniaturized.

The first product of Integrated Circuit (IC) was developed by Intel, an American electronic device company, in the early 1960s. The underlying idea of IC was making transistors with high density on the limited area of a silicon chip and connecting them to form certain functional circuits. Besides the merits shown in transistors, one of their other merits was high reliability. With IC products, computers can be made more portable in volume but more versatile in function than all previously made computers. By the end of the 1960s, IC products had taken the place of discrete transistors and other components to build computers.

But scientists and engineers were not content with the existing density of integration. With the development of micro-electronics and laser technique, the precision of photo-processing in manufacturing IC products could ensure the stripes of every component made on the surface of a silicon chip as narrow as 10 micrometers. This figure meant more transistors made within limited area on a silicon chip and much higher working speed. The product was called the Large Scale Integrated Circuit (LSIC). With LSIC, the computer could be made as small as the personal computer, the desktop computer and the laptop computer. Soon, families of LSIC products dominated manufacture of computers.

Development of Computer Software

In software, the progress lies in the advent of operating system and various applied software packages oriented to various practical uses. In the early computers, there was no software at all. The programmer had to operate his program directly on the operator's console-loading his program into memory, from either paper tape or cards or switches on the front panel, monitoring its execution by watching display lights on the console, and debugging it if any error occurred. The working efficiency was surprisingly low, and the computer frequently sat idle. To solve such problems, the operating system emerged and got mature. The operating system was software exclusive for control of the computer itself, ranging from CPU allocation, queuing service for multiple users, allocation of other computer resourceslike the bus, input/output equipment, memory space, moni toring execution of user programs, to controlling routine operations. With an operating system equipped, the computer became user-friendly. Users did not bother themselves about all operations and manipulations that were essentially responsibility of the computer system itself.

Developed in the mid-1970s, UNIX was the most famous operating system for medium- and large-scale computers, designed mainly for multiprogramming and time-sharing systems. Another form of UNIX operating system was the real-time system often used as a control device in a dedicated application. Sensors were used in the operating system, which conducted and adjusted control to modify the controlled process. A real-time operating system had a well-defined fixed time constraint. Processings must be done within the defined constraint or the system might fail.

DOS and Windows are two of the operating systems developed for microcomputers or personal computers. (Refer to the text of the lesson about personal computers for details). Compared with UNIX, these operating systems are much simpler but none the less essential to the routine operations.

The applied software refers to those that convenience user-created programs for solving practical problems in specific domains. Every user program must be written in a specific computer programming language, such as assembly, COBOL, FORTRAN, BASIC, PASCAL, C, C++, LISP, JAVA. Each language is for specific use, for example, COBOL is convenient for programs in finance and accounting, FORTRAN is good for scientific calculations, PASCAL is widely used in engineering designs, C is suited for automatic control engineering, LISP is exclusive for artificial intelligence research, BASIC is usually used in simple programs, doing basic arithmetic and logical computation and printing. Assembly is a machine-oriented language. Programs written in an assembly language can work faster than programs written in high-level programming languages——FORTRAN, PASCAL, LISP, etc.

High-level programming languages are, to some extent, similar to natural language in sentence structure and in vocabulary, so that it is easier to write programs in high-level programming languages. But programs written in high-level programming languages are beyond the capability of any computer-they simply ignore them. To bridge the gap, compilers are needed to translate programs written in high-level programming languages into machine code that the computer can execute. Each high-level programming language has its own compiler-software designed to do the translation. All computers are equipped with certain number of compilers to meet the need of users of various professional domains.

介绍数字式计算机

冯·诺伊曼体系结构

在二战期间，美国需要快速的计算机器，用以制作各类枪炮射弹的飞行轨迹图表，同时为了对第一颗原子弹的设计作模拟，这就促使自动的计算机器——数字式计算机的诞生。John Mauchly 和 J. Presper Eckert 开发了被称为 ENIAC 的第一台真正的数字式计算机。他们的工作由普林斯顿大学的冯·诺伊曼加以发展和改进。

ENIAC 是 1942 年在伊利诺伊州 Evanston 市的西北大学的一

所实验室里制成的。它由三万只以上的电子管和近一百万只其他的元部件,例如电阻器、电容器和开关组成。它占用了三间大房间,而它消耗的电力足够为拥有三万居民的小城市供电。为了散发从机器内部发出的热量，使用了强力排风机。它的运算速度是很低的,大约每秒 2000 次操作。

但是,这台计算机开创了计算机时代。它是全新的机器,它能做涉及人类思维过程的工作，例如，简单的逻辑推理、基本的算术运算和数据处理。特别是它的被称作“冯·诺尔曼体系结构”的结构和它的被称作“储存程序概念”的工作原理被证明是普遍适用于一切为完成事实上所有类型的计算而建造的数字式计算机。自此之后，在计算机硬件方面取得了许多伟大的成就，但是，即使是今天的计算机——从最大的大型计算机到最小的个人计算机——都采用了该概念的某些变种。专家们预测在今后许多年里这一概念仍将有效且被采用。

所有的冯·诺伊曼类型的数字式计算机都由四个基本部分组成——存储器、算术—逻辑单元(ALU)、控制单元和输入—输出单元(I/O)。ALU 和控制单元通常被认为是一只集成的部件,称作中央处理器（CPU)。大多数的微处理器其实只是一块集成电路芯片 CPU。

数字式计算机按照冯·诺伊曼的储存程序概念运行。程序由一系列的、用一种特定的码写成的指令组成,例如,二进制数—0 和 1、汇编码、高级编程语言。它们按照一定的规律编排,指示计算机进行特定的操作,例如,把二进制数移进或移出存储器和在各种寄存器与 CPU 之间移动，及各种计算和逻辑运算。指令按照步进的方式编排成一序列,构成被称作算法的子程序。计算机程序由一系列用于解决某些特定问题的算法组成。这些程序存放在数字式存储器内。每一指令在存储器中都有一个地址码。通常地址码都按地址的增量编排。在控制单元里有一只地址码计数器,用以记录当前处理的指令的地址码。使地址码计数器产生增量,控制单元就可以转向下一次要取来的指令；指令取来以后，它的操作码就被译码,

产生一个信号；该信号被送到一个适当的单元，于是操作就可以开始。操作数可以在一个被称作“起始地址”的地址找到。操作的结果被送到“目的地址”指定的存储单元里。

当控制需要离开当前的操作循环而转向另一组指令时，使用“转移指令”。这种转移指令中的“无条件转移指令”，总是对指令地址计数器作修改，并转向另一指令。“条件转移指令”会对指令地址计数器产生不同的修改，决定于条件转移指令中预先设定的某种计算的结果。

这一叙述告诉我们为什么冯·诺伊曼型计算机可以自动地运行，精确地按照存储程序的要求完成各种算术和逻辑运算。

计算机硬件的发展

数字式计算机的硬件经历了一系列的革命性的变革。它在工作速度和功能上取得的进步给人们留下深刻的印象。第一只晶体管是1948年由贝尔实验室发明。计算机制造厂家很快就把这一新发明用于改进他们的计算机，使它们具有突出的优点——体积小、重量轻、功耗小、寿命长、工作速度高。用晶体管取代电子管后，功能电路可以制作在印刷电路板上，装配在插座上，大大地方便了制造与维修。与晶体管相配的其它元件也小型化了。

第一只集成电路（IC）产品由美国的一家电子器件公司因特尔在60年代初开发成功。IC的基本思想就是在有限的硅片面积上制造高密度晶体管并且把它们联接成为功能电路。除了晶体管的优点以外，它的其他优点之一就是高可靠性。利用IC产品制成的计算机与以前制造的一切计算机比较体积更小，功能更多。到了60年代末，IC产品已取代了分立的晶体管和其他元件，用于制造计算机。

但是，科学家和工程师们并不满足于现有的集成密度。由于微电子学和激光技术的发展，在制造IC产品中光处理的精度已保证硅芯片表面上的条纹宽度可以窄到10微米。这一数字意味着在硅芯片的有限面积上可以制造出更多的晶体管，而且工作速度高得

多。这种产品称为大规模集成电路(LSIC)。有了 LSIC,计算机可以制成小到个人计算机、桌面式计算机、膝上式计算机。很快,LSIC 就支配了计算机制造业。

计算机软件的发展

在软件方面，进步就在于操作系统和面向各种实际应用的应用软件的出现。在早期的计算机中，根本没有软件；程序编制人员不得不在操作员的控制台面板上直接运作他们的程序，利用纸带、卡片或面板上的开关把程序加载到存储器里，盯着看面板上的指示灯，监控它的运行；如果有什么错误发生，还要排除程序上的错误。工作效率低得惊人，而计算机经常闲呆着。为了解决这类问题，操作系统出现了并且成熟了。操作系统是一种专门用来控制计算机本身的软件，从分配 CPU(时间)，为多个使用者提供排队服务，分配其他的计算机资源如总线、输入/输出设备、存储器空间，监控用户程序的运行，到控制例行操作。计算机安装了操作系统，就便于使用。使用者不用为计算机本身应当负责的全部操作和处理而烦心了。

到了 70 年代中期，为中型和大型计算机使用的最著名的 UNIX 操作系统出现了。它主要是为多道程序设计和分时系统而设计。另一种类型的 UNIX 操作系统是一种实时系统，常用作专用控制系统。在此种操作系统中使用传感器，以进行和调整控制，改变被控过程。实时操作系统有预先定好的、固定的时间限制。操作必须在这固定的时限内完成，否则系统就要崩溃。

DOS 和 Windows 是为微机或个人计算机开发的两种操作系统(详见有关个人计算机的课文)。与 UNIX 比较，这些操作系统简单得多，但就例行操作而言，其重要性一点也不逊色。

应用软件指的是那些方便在特定领域内为解决实际问题由用户创立的程序运行的软件。每一种用户程序都必须用某一种特定的语言编写，例如，汇编语言，COBOL，FORTRAN，BASIC，PASCAL，C，C + +，LISP，JAVA。每一种语言都有特定的用途，例如，

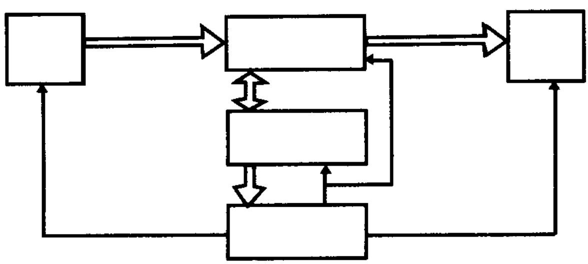
COBOL 方便编写财务和会计的程序，FORTRAN 便于科学计算，PASCAL 广泛用于工程设计，C 适合用于自动控制工程，LISP 专门用于人工智能研究，BASIC 通常用于简单的程序设计，做基本算术运算、逻辑运算和打印操作。汇编语言是面向机器的语言，用汇编语言编写的程序起来要比用高级编程语言——FORTRAN，PASCAL，LISP 等快些。

高级编程语言在某种程度上，在句子结构和词汇上类似自然语言，因此，用高级编程语言编写程序比较容易。但是，用高级编程语言编写的程序超出了计算机能力——它们根本不懂这种程序。为了弥补这一差距，需要编译程序把用高级编程语言编写的程序翻译成为计算机能执行的机器码。每一种高级编程语言都有自己的编译程序。所有的计算机都安装了一定数量的编译程序，以满足各种领域的需要。

Lesson 2

Basic Structure of Digital Computers

All digital computers of von Neumann type, however complicated, can be divided into 4 basic units: the input/output unit, the arithmetic unit, the control unit and the memory as shown in the following diagram.



The Basic Structure of Digital Computers

The Input/Output Unit

The input/output unit is equipment to read the information and the computer instructions into the machine (the input) and to print or display the results of the computation (the output). Many devices are available to perform these tasks; the most widely used one is the CRT (cathode ray tube) terminal.

Usually, the rate of processing information in the arithmetic unit is much faster than in either the input or output equipment. This is primarily because the input/output units usually involve some mechanical operations, whereas the internal calculations proceed at electronic speed. This speed difference is not critical if the information required to be processed and the result are small in amount