

◆ 高等教育应用型本科人才培养系列教材

# 计算机系统导论

JISUANJI XITONG DAOLUN

主 编 高 迪

副主编 寇 亮 郎大鹏 赵国冬

付小晶 张如意



100101101010010100101010100101101010001001011100111001100101010010010101  
0101010100101101010001001011100111001100101010010010101101010101010100101010100101101

 哈尔滨工程大学出版社  
Harbin Engineering University Press

型本科人才培养系列教材

# 计算机系统导论

主 编 高 迪  
副主编 寇 亮 郎大鹏  
赵国冬 付小晶  
张如意



哈尔滨工程大学出版社  
Harbin Engineering University Press

## 内容简介

本书介绍了计算机的起源、发展、基础学科和计算机新兴热门领域的相关内容,主要包括计算机的发展、运算、系统组成、软件操作、计算机网络及应用、算法与数据结构、计算理论、人工智能、软件工程、信息安全与网络安全、计算机伦理与职业道德等。

本书既可作为高等院校计算机专业本科教材,也可供相关人员参考。

## 图书在版编目(CIP)数据

计算机系统导论/高迪主编. —哈尔滨:哈尔滨  
工程大学出版社,2018.7

ISBN 978 - 7 - 5661 - 2024 - 3

I. ①计… II. ①高… III. ①计算机系统 IV.  
①TP30

中国版本图书馆 CIP 数据核字(2018)第 150953 号

选题策划 夏飞洋  
责任编辑 张忠远  
封面设计 刘长友

---

出版发行 哈尔滨工程大学出版社  
社 址 哈尔滨市南岗区南通大街 145 号  
邮政编码 150001  
发行电话 0451 - 82519328  
传 真 0451 - 82519699  
经 销 新华书店  
印 刷 哈尔滨市石桥印务有限公司  
开 本 787 mm × 1 092 mm 1/16  
印 张 17.25  
字 数 441 千字  
版 次 2018 年 7 月第 1 版  
印 次 2018 年 7 月第 1 次印刷  
定 价 49.80 元

<http://www.hrbeupress.com>

E-mail:heupress@hrbeu.edu.cn

---

# 前 言

当今计算机科学与技术的发展速度是令人惊叹不已的,计算机技术的应用已不断向深度和广度拓展。计算机拥有准确的逻辑判断能力、完善的运算能力与强大的存储能力,高速发展的硬件和软件功能的逐步完善使计算机与科学计算、人工智能、信息安全和实时控制等息息相关,并帮助人类进入了一个高度发达、高度文明的信息社会。由此,计算机教育的高度普及被不断加速,现如今,掌握计算机技术就如同拥有了开启信息时代大门的钥匙。

在各大高校教学改革中,为了顺应信息时代的发展,助力高等院校计算机基础教学,计算机系统导论被设置为大学生的基础课程,它是其他计算机相关课程的前导课程。对于非计算机专业的学生来说,这门课是了解计算机基本构造和运行原理并且引导学生实际应用的计算机基础知识课。由此,通过总结多年教学经验、开展调研并参考最新、最前沿的技术,本书的编写结合了具有基础性、科学性和前沿性的学科内容,充分反映了本学科领域的基本构成和最新科技成果,实现了教育与科研的有效结合。通过教材的编写,调整学生的知识结构体系和增强学生的能力素质,充分体现了当前高等教育所需要的新形势、新技术和新目标。本书架构新颖,内容基础实用,起点适中,由浅入深,结合了基础性、实用性和创新性等特点,书中各章均配有习题并附有参考答案。

本书共分为十章。

第1章计算机系统基本概述,主要包括计算机概述、计算机基本构成、计算机学科发展。

第2章计算机运算基础技术,主要包括计算机编码、进位计数制、二进制算术运算、数的符号表示、字符信息的表示。

第3章计算机系统组成,主要介绍了包括结构和指令系统的计算机硬件,计算机系统单元,输入、输出设备。

第4章计算机软件操作,主要包括操作系统、程序设计语言和数据库原理三大模块。其中操作系统部分对操作系统的结构、功能与设计、进程管理、调度与死锁等主要内容有所讲解。程序设计语言浅显地对基本构成和处理系统进行说明并对程序设计过程有所涉及。

第5章计算机网络及应用,主要介绍了计算机网络及其应用。在计算机网络应用中,介绍了应用方面的新技术及时下热门的新概念“互联网+”。

第6章算法与数据结构。算法部分由浅入深地介绍了算法的基本概念、特征与分类和设计描述。数据结构部分包括线性表、栈和队列,数组、广义表和树结构,图结构、查找和内部排序等学科基本概念。

第7章计算理论、人工智能。计算理论部分主要包括计算机的数学基础离散数学,图灵

机、歌德尔数、停机问题和问题的复杂度等。在人工智能部分为读者介绍了知识表示、专家系统、感知、搜索、神经网络等方面。

第8章软件工程基础,介绍了软件工程的本质,以及软件工程所涉及的流程,包括需求分析、设计和后续的测试与风险管理。

第9章信息安全与网络安全。其中信息安全包括密码学、系统安全和内容安全;网络安全包括漏洞、恶意软件、入侵检测技术及当下热门的新技术。

第10章计算机伦理与职业道德,介绍了与计算机伦理学及职业道德相关的内容和法律法规。

本书由哈尔滨工程大学计算机科学与技术学院教师编写,向在本书编写过程中给予极大帮助的何姝禹和阎梓宁同学表示衷心感谢。

由于计算机学科发展迅速及编者水平有限,书中难免有欠妥之处,欢迎读者批评指正。

编者

2018年4月

# 目 录

第 1 章 计算机系统基本概述 .....	1
1.1 计算机概述 .....	1
1.2 计算机基本构成 .....	2
1.3 计算机学科发展 .....	5
第 2 章 计算机运算基础技术 .....	15
2.1 计算机编码 .....	15
2.2 进位计数制 .....	16
2.3 二进制算术运算 .....	18
2.4 数的符号表示 .....	19
2.5 字符信息的表示 .....	20
第 3 章 计算机系统组成 .....	25
3.1 计算机硬件 .....	25
3.2 计算机系统单元 .....	39
3.3 输入、输出系统 .....	44
第 4 章 计算机软件操作 .....	52
4.1 操作系统 .....	52
4.2 程序设计语言 .....	72
4.3 数据库原理 .....	82
第 5 章 计算机网络及应用 .....	104
5.1 计算机网络 .....	104
5.2 计算机网络应用 .....	126
第 6 章 算法与数据结构 .....	137
6.1 算法的概念 .....	137
6.2 数据结构 .....	147
第 7 章 计算理论和人工智能 .....	160
7.1 计算理论 .....	160
7.2 人工智能 .....	164
第 8 章 软件工程基础 .....	185
8.1 相关概念 .....	185

8.2 软件工程需求与设计 .....	194
<b>第9章 信息安全与网络安全 .....</b>	<b>215</b>
9.1 信息安全相关概念 .....	215
9.2 网络安全的基本概念和术语 .....	228
<b>第10章 计算机伦理与职业道德 .....</b>	<b>242</b>
10.1 计算机伦理学 .....	242
10.2 计算机职业道德 .....	248
<b>参考文献 .....</b>	<b>256</b>
<b>计算机系统导论自学考试大纲 .....</b>	<b>257</b>
题型举例 .....	268

# 第1章 计算机系统基本概述

## 1.1 计算机概述

近年来,随着计算机技术的飞速发展,计算机及其应用已渗透到社会的各个领域,掌握和使用计算机已成为人们必不可少的技能。特别是网络技术的迅猛发展和普及,使得计算机已经不仅是一个工具,而是作为一种全新的处理方式为人们所接受。本章将介绍计算机的基础知识,使读者能对计算机有一个比较全面的了解。

### 1.1.1 计算机的起源

计算机是一种能快速、高效、准确地进行信息处理的数字化电子设备,它能按照人们事先编写的程序自动地对信息进行加工和处理,输出人们所需要的结果,从而为人类的生产、生活服务。这里包括两个方面的含义:

一是计算机是进行信息处理的工具。信息是指所有能被计算机识别和使用的数据,包括字符、声音、图像、视频信号等。计算机是一种获取信息、传递信息和加工信息的智能化设备,而不仅仅是狭义上的算术和逻辑运算的计算工具。

二是计算机能按照人们事先编制并存放在它内部的程序自动完成信息处理任务。计算机只能按照人们编制的程序所规定的步骤对信息进行处理,而且程序必须存储在计算机内,计算机才能自动工作。计算机只是人类发明的工具,它能帮助人们完成一些复杂的工作,但它不是万能的,也不能代替人脑。

综上所述,计算机是一种能够按照人们编写的程序连续、自动地工作,并能对输入的数据信息进行加工、存储、传递,由电子和机械部件组成的电子设备。

### 1.1.2 计算机的基本概念

计算机,顾名思义是一种计算的机器。它由一系列电子器件组成,英文名称为 computer。计算机诞生的初期主要用来进行科学计算。然而现在计算机的处理对象已远远超出了“计算”这个范围。现在计算机可以对数字、文字、颜色、声音、图形、图像等各种形式的数据进行加工处理。

当用计算机进行数据处理时,首先把要解决的实际问题用计算机语言编写成计算机程序,然后将待处理的数据和程序输入到计算机中,计算机按程序的要求一步一步地进行各种运算,直到存入的整个程序执行完毕为止,因此计算机必须是能存储程序和数据的装置。

计算机在数据处理过程中不但能进行加、减、乘、除等算术运算,而且能进行逻辑运算并对运算结果进行判断,进而决定后续操作。因此,计算机具有进行各种计算的能力。



在当今的信息社会里,各行各业随时随地产生大量的信息。人们为了高效地获取、传递、检索信息并从信息中产生各种报表数据,必须在计算机的控制下进行有效的组织和管理信息。因此说计算机是信息处理的工具。

综上所述,计算机可以定义为:计算机是一种能按照事先存储的程序自动、高速地进行大量数值计算和各种信息处理的现代化智能电子设备。

## 1.2 计算机基本构成

计算机是人们对个人计算机的俗称。自从1946年第一台计算机诞生之后,先后经历了电子管时代、晶体管时代、集成电路时代和超大型集成电路时代。1981年,IBM公司推出了划时代的个人计算机(PC)。从此,计算机进入了平常人家,而个人计算机也被人们亲切地称为计算机。要想熟练地组装计算机,必须清楚计算机的基本构成,下面就对计算机的各配件进行简要介绍,使读者对计算机有一个初步的认识。

### 1.2.1 计算机硬件系统

计算机硬件系统是指构成计算机物理结构的电气、电子和机械部件,它是计算机系统的物质基础。1946年美籍匈牙利数学家冯·诺依曼提出了计算机的硬件结构,其主要由运算器、控制器、存储器、输入设备和输出设备五大基本部件组成。

#### 1. 运算器

运算器是计算机进行信息加工的场所,所有的算术运算和逻辑运算都在这里进行。算术运算指加、减、乘、除等各种数值运算;逻辑运算指进行逻辑判断、逻辑比较的非数值运算。

#### 2. 控制器

控制器是计算机的指挥控制中心,是计算机的“神经中枢”。它负责对控制信息进行分析,通过分析发出操作控制信号,控制数据的传输和加工;同时,控制器也接收其他部件送来的信号,协调计算机各个部件步调一致地工作。

#### 3. 存储器

存储器是计算机存储与记忆的装置,用来存放计算机的数据与程序。存储器通常分为内存储器和外存储器。

内存储器(简称内存,英文为 memory)用来存放当前运行的程序和数据。它存储容量较小,但存取速度快,并可以直接与中央处理器(CPU)交换信息。内存储器可分为只读存储器(ROM)和随机存储器(RAM)。

外存储器(简称外存,英文为 External Memory)也称为辅助存储器,它是内存的扩充。其特点是存储容量大、价格低,但存取速度相对内存较慢。一般用来存放暂时不用的程序和数据,在需要时可成批地与内存交换信息。CPU不能直接访问它,即外存中的信息不能被直接处理,必须预先被送入内存才能被处理。

#### 4. 输入设备

输入设备是计算机用来接收外界信息的设备,主要是把程序、数据和各种信息转换成计算机能识别接收的电信号,按顺序送到计算机内存中。目前常用的输入设备有键盘、鼠

标、扫描仪等。

## 5. 输出设备

输出设备是用来输出数据处理结果或其他信息的,主要是把计算机处理的数据、计算结果等内部信息按人们需要的形式输出。常见的输出设备有显示器、打印机、绘图仪等。

## 1.2.2 计算机软件系统

软件是指计算机系统中的程序、数据和有关文件(文档)的集合。计算机程序是按既定算法,采用某种计算机语言所规定的指令或语句的有序集合;文档是指用自然语言或者形式化语言所编写的文字资料和图表,用来描述程序的内容、组成、设计、功能规格、开发情况、测试结果及使用方法,如程序设计说明书、流程图和用户手册等。

通常按功能可将计算机软件系统分为系统软件和应用软件两大类。

### 1. 系统软件

系统软件是计算机系统中最靠近硬件的软件。系统软件指的是管理、运行、控制和维护计算机系统资源的程序集合。它的主要功能是实现计算机硬件和软件管理,充分发挥计算机的功能,方便用户的使用,为开发人员提供平台支持。系统软件主要包括操作系统软件和实用系统软件。

#### (1) 操作系统

操作系统是系统的核心,起着管理整个系统资源的作用。操作系统的主要功能是控制和管理计算机系统各种硬件和软件资源,提高资源的利用率和使用率,为用户提供一个良好的计算机系统环境。它是硬件(裸机)上扩充的第一层软件。其他的软件,如汇编语言、编译程序和各种服务程序等软件都是在操作系统的支持下工作的。因此操作系统被称为用户与计算机之间的接口。

#### (2) 实用系统软件

实用系统软件包括语言处理程序、编辑程序、连接程序、管理程序、调试程序、故障检查程序和各种实用工具程序等。

##### ① 语言处理程序

人与人之间交往主要是通过语言进行的。同理,人与计算机之间交换信息也必须用一种语言,这种语言就叫计算机语言或程序设计语言。根据计算机科学技术的发展,计算机语言可以分为三类,即机器语言、汇编语言和高级语言。

##### ② 实用工具程序

实用工具程序能配合其他系统软件为用户提供方便和帮助。在 Windows 的附件中也包含了系统工具,包括磁盘碎片整理程序、磁盘清理等实用工具程序。

#### ◆ 机器语言

机器语言是用二进制“0”和“1”构成一系列指令代码表示的程序设计语言。它是计算机能直接识别和执行的语言。它具有执行速度快、占用内存少等优点。但是它难学、难记、难阅读,并且难以纠错。另外,不同型号计算机的机器语言不能通用。

#### ◆ 汇编语言

汇编语言是为了解决机器语言难记忆、编程不方便等问题,使用了一些能反映指令功能的助记符来代替机器指令的符号语言。机器语言和汇编语言都是低级语言(面向机器的语言)。汇编语言比机器语言直观,易阅读、易编程和易修改。汇编语言程序不能直接执

行,需要将汇编语言源程序通过汇编程序翻译成机器语言(目标程序),如图 1.1 所示。

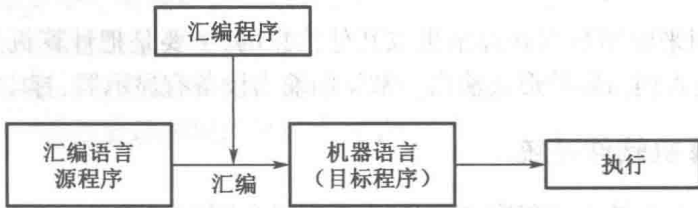


图 1.1 汇编语言源程序通过汇编程序翻译成机器语言

#### ◆高级语言

高级语言是接近人类的自然语言和数字语言而又独立于机器的一种程序设计语言。用高级语言编写的源程序不能被计算机直接识别,需要把高级语言通过“翻译程序”变成机器语言。

不同的计算机语言有不同的语言处理程序,把高级语言编写的源程序翻译成目标程序有两种方式。

一种是解释,即一边翻译一边执行。将高级语言源程序逐条翻译成机器指令,翻译一句执行一句,直到程序全部翻译执行完,这种“翻译”的处理程序称为解释程序,如图 1.2 所示。

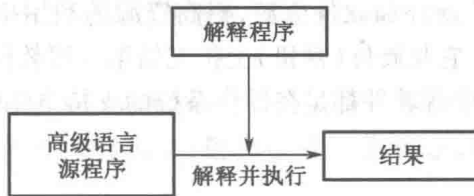


图 1.2 解释过程示意图

另一种是编译,即把高级语言源程序“翻译”成一个完整的目标程序,然后再由计算机执行目标程序,这种“翻译”语言处理程序称为编译程序,如图 1.3 所示。

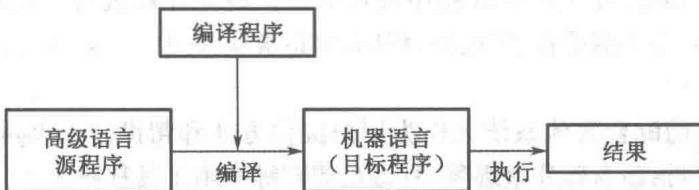


图 1.3 编译过程示意图

## 2. 应用软件

应用软件是为计算机在各个领域中的应用而开发的程序。它是利用计算机软件、硬件资源为解决特定的实际问题而编制的程序和文档。常见的应用软件有各种软件包、数据库应用程序等。前者是指为解决带有通用性问题而研制开发的程序,比如办公应用方面的 Microsoft Office,网站制作方面的 Dreamweaver 等;后者是指用户针对特定问题而编制的程序。

计算机软件已发展成为一个巨大的产业。其应用范围也涵盖了生活的方方面面,很多问题都有相应的软件来解决。下面列举了一些主要应用领域的常见软件。

办公软件:Microsoft Office、WPS等。

平面设计:Photoshop、Freehand、CorelDRAW等。

辅助设计:AutoCAD、Rhino等。

三维制作:3DS MAX、Maya等。

视频编辑与后期制作:Adobe Premiere、Ulead的会声会影等。

多媒体开发:Authorware、Director、Flash等。

### 1.3 计算机学科发展

计算机科学与技术是研究信息过程、用以表达此过程的信息结构和规则及其在信息处理系统中实现的学科。计算机科学与技术研究的主要对象是现代计算机及其相关的现象。该学科的工作集中于计算机系统的结构和操作、计算机系统的设计和程序设计的基本原则、使之运用于各种信息加工任务的有效方法,以及它们的特性和局限的理论特征。学科包括科学与工程两方面,二者相互作用、相互影响。

半个多世纪以来,计算机科学技术迅猛发展,成为当代科技的非常重要的学科。随着电子技术的发展,计算机的逻辑器件不断更新换代,目前已进入了超大规模集成电路的时代。微电子技术的变化发展,直接带动了计算机系统结构的发展,许多行之有效的理论和方法得以应用。计算机已经从早期的单一计算装置发展成多计算机系统、并行分布式计算机系统、计算机网络等多种形式的高性能系统。微型计算机的产生与发展,进一步改变了人类社会生产、生活方式。软件理论、技术的发展和软件工程方法导致了软件设计和开发方法的根本变革。理论研究已经从单纯的计算模型的研究发展为计算机系统理论、软件理论、计算理论和应用技术理论等多个研究分支,并拓展到人工智能等方面。

计算机科学与技术学科可分为理论计算机科学、计算机软件、计算机系统结构、计算机应用技术等领域及与其他学科交叉的研究领域,如人工智能、应用数学等。通常,本学科可概括为计算机软件与理论、计算机系统结构、计算机应用技术等三个二级学科。

计算机软件与理论主要研究软件设计、开发、维护和使用过程中涉及的软件理论、方法和技术,探讨计算机科学与技术发展的理论基础。计算机系统结构研究计算机硬件与软件的功能分配,软硬件界面的划分,计算机硬件结构、组成、实现方法与技术。计算机应用技术研究计算机在各个领域的原理、方法和技术,所涉及的研究内容非常广泛。

计算机科学与技术不但自身已经成为高技术的重要学科之一,而且对于当代众多的科学技术领域的发展有着重要的影响。随着人类社会信息化程度的提高,计算机科学与技术所涉及的研究领域和方向会不断扩展、深化,对于人类社会的生产和生活将发挥越来越重要的作用。

计算机应用技术是一门应用十分广泛的专业,它以计算机基本理论为基础,突出计算机和网络的实际应用。学生通过系统地学习计算机的软、硬件与应用的基本理论、基本技能和方法,具备初步运用专业基础理论及工程技术方法进行系统开发、应用、管理和维护的能力。

### 1.3.1 计算机学科的知识结构、学科形态

随着计算机的发展,计算机专业教育受到了高度的重视,出现了许多专业名称,如计算机技术、计算机科学、计算机工程和计算机应用等。它们有区别吗?要回答这个问题,首先必须搞清楚什么是科学、技术、工程。科学是关于自然、社会和思维的发展与变化的知识体系;技术是泛指根据生产实践经验和科学原理而发展形成的各种工艺操作方法、技能和技巧;工程是指将科学原理应用到工农业生产部门中去而形成的各门学科的总称。因此,科学、技术、工程是有区别的。

正因为如此,长期以来国内外一直对计算机科学与技术是属于科学还是工程范畴存在着争议,计算机界也一直争论不休。其反映在专业课程设置、核心课程确定上是“仁者见仁,智者见智”,强调科学的不重视技术与工程实践;强调应用的轻视理论,在教与学的过程中过分注重计算机实际操作的应用,就如同懂电的强调电,懂硬件的强调硬件,懂数学的强调数学,懂软件的强调编程。

针对这一问题,美国计算机协会(ACM)和国际电气电子工程师协会计算机学会(IEEE/CS)组织了一个由20多位资深计算机专家(其中有不少是“图灵”奖的获得者)组成的联合小组,从1985年起用了五年的时间对计算机科学、计算机工程领域的相关问题进行了翔实的论证与分析,最后他们得出的结论是:计算机科学和计算机工程在本质上没有区别,两者是一回事。为什么会这样呢?这是由计算机学科的特点所决定的。

事实上,代表计算机科学的各个分支学科的理论和方法与代表计算机工程的各分支的学科的工程开发方法和技术、技巧、技艺通常既有理论特征,又有技术特征,甚至还具有工程特征,三者之间的界限往往难以区分。从本质上讲,它们是从不同的角度和层面对各种问题是否可计算,即能行性及其求解方法和过程的描述,以及通过对各种问题反映其能行性的内在规律的描述,折射出求解方法和求解过程的。它们的方法论的理论基础都是以离散数学为代表的构造性数学。

基于上述认识,近年来科学界倾向于将计算机科学、计算机技术、计算机工程统一为一个学科,称之为计算科学。对计算科学比较一致的定义是:计算科学是对描述和变换信息的算法过程,包括理论、分析、设计、效率分析、实现和应用的系统研究。全部计算科学的基本问题是:什么能自动进行,什么不能自动进行。本学科来源于对数理逻辑、计算模型、算法理论、自动计算机器的研究,形成于20世纪30年代后期。

那什么是计算机应用?计算机应用是一个范畴很大的领域,广义地讲,凡是和计算机使用相关联的领域都可被纳入计算机应用的范畴。但是从基本的和核心的角度来看,支持计算科学向各个学科渗透、应用和发展的是一些最基本的共性理论、方法和技术。因此,首先我们应将计算机技术导论的操作和应用区分开来。操作是对现有软件的理解和使用,而应用又可分为具体应用和基本应用。将计算机与各行各业的具体事务结合起来的应用,如信息系统、电子商务等,称为计算机具体应用;而研究计算机应用具体领域的共性理论、方法和技术,称为计算机的基本应用,属于计算科学的范畴。我们通常所说的计算机应用,是指计算机的基本应用,而不是计算机的具体应用,更不是计算机的操作。

多年来,计算机学科已形成完整的理论及应用体系,具有明显的专业特征和特定的研究领域。进入21世纪,计算机学科的知识内容也有了很大的变化和发展,为此ACM和IEEE/CS以及中国计算机学会教育分会组织专家进行了全面的讨论、论证,比较一致的看

法是,新世纪计算机学科的知识领域可分为十四个主科目(核心知识体),它们分别是:

- (1)离散结构(DS);
- (2)程序设计基础(PF);
- (3)算法与复杂性(AL);
- (4)程序设计语言(PL);
- (5)组成与系统结构(AR);
- (6)操作系统(OS);
- (7)网络计算(NC);
- (8)人机交互(HC);
- (9)图形与可视化(GV);
- (10)智能系统(IS);
- (11)信息管理(IM);
- (12)软件工程(SE);
- (13)社会和职业专题(SP);
- (14)计算科学与数值方法(CN)。

### 1.3.2 计算机学科核心概念和典型方法

在计算机学科中有十二个核心概念。

#### 1. 绑定

绑定是指通过将一个抽象的概念与附加特性相联系,从而使一个抽象概念具体化的过程。

例如,把一个进程与一个处理机、一种类型与一个变量名、一个库目标程序与子程序中的一个符号引用等分别关联起来。

在逻辑程序设计中,用面向对象语言将一个方法与一个消息相关联,从抽象描述建立具体实例。

绑定有时又译为联编、结合等。译为绑定既可表音,又能达义,在计算机专业英语的汉译中能达到这一境界的并不多。

绑定在计算机领域中存在很多实例。面向对象程序设计中的多态性特征将这一概念发挥得淋漓尽致。程序在运行期间的多态性取决于函数名与函数体相关联的动态性,只有支持动态绑定的程序设计语言,才能表达运行期间的多态性,而传统语言通常只支持函数名与函数体的静态绑定。

还可为绑定找到一个更通俗的实例。将配偶这一抽象概念与某位异性相关联,这一过程称作绑定。指腹为婚为静态绑定,自由恋爱为动态绑定。现有的面向对象程序设计语言都不允许离婚或重婚,但在一定程度上允许再婚。

#### 2. 大问题的复杂性

大问题的复杂性是指随着问题规模的增长,复杂性呈非线性增加的效应。这是区分和选择各种方法的重要因素。以此来度量不同的数据规模、问题空间和程序规模。

假如我们编写的程序只是处理全班近百人的成绩排序,那么选择一个最简单的排序算法就可以了。但如果我们编写的程序负责处理全省几十万考生的高考成绩排序,就必须认真选择一个排序算法,因为随着数据量的增大,一个不好的算法的执行时间可能是按指数



级增长的,从而使人最终无法忍受等待该算法输出结果的过程。

软件设计中的许多机制正是面向复杂问题的。例如在一个小程序中标识符的命名原则是无关重要的,但在一个多人合作开发的软件系统中这种重要性会体现出来;goto 语句自由灵活、可随意操控,但实践证明了在复杂程序中控制流的无序弊远大于利;结构化程序设计已取得不错的成绩,但在更大规模问题求解时保持解空间与问题空间结构的一致性显得更加重要。

从某种意义上说,程序设计技术发展至今的两个里程碑(结构化程序设计的诞生和面向对象程序设计的诞生)都是因为应用领域的问题规模与复杂性不断增长而驱动的。

### 3. 概念和形式模型

概念和形式模型是指对一个想法或问题进行形式化、特征化、可视化和思维化的各种方法。例如,在逻辑、开关理论和计算理论中的形式模型;基于形式模型的程序设计语言的风范;关于概念模型,诸如抽象数据类型、语义数据类型和用于指定系统设计的图形语言,如数据流和实体关系图。

概念和形式模型主要采用数学方法进行研究。例如,用于研究计算能力的常用计算模型有图灵机、递归函数、 $\lambda$  演算等;用于研究并行与分布式特性的常用并发模型有 Petri 网、CCS、 $\pi$  演算等。

只有跨越了形式化与非形式化的鸿沟,才能到达软件自动化的彼岸。在程序设计语言的语法方面,由于建立了完善的概念和形式模型,包括线性文法与上下文无关文法、有限自动机与下推自动机、正则表达式与巴克斯范式等,所以对任何新设计语言的词法分析与语法分析可实现自动化,典型的软件工具有 lex 和 yacc。

在形式语义方面,虽然操作语义学、指称语义学、公理语义学和代数语义学四大流派均取得不少成果,但语义分析工具目前还仅限于实验室应用。

至于程序设计语言的语用方面,由于严重缺乏概念和形式模型,人们对语言的语用知之甚少,更谈不上什么自动化工具。

### 4. 一致性和完备性

在计算机中一致性和完备性概念的具体体现包括诸如正确性、健壮性、可靠性这类相关的概念。一致性包括用作形式说明的一组公理的一致性、观察到的事实与理论的一致性、一种语言或接口设计的内部一致性等。正确性可看作部件或系统的行为对设计说明的一致性。

完备性包括给出的一组公理使其能获得预期行为的充分性、软件和硬件系统功能的充分性,以及系统在出错和非预期情况下保持正常行为的能力。

一致性与完备性是一个系统必须满足的两个性质,在形式系统中这两个性质更加突出。如果你提出了一个公理系统,人们首先会提出的问题就是该系统是否一致,该系统是否完备。

一致性是一个相对的概念,通常是在对立统一的双方之间应满足的关系,例如实现相对于规格说明的一致性(即程序的正确性),数据流图分解相对于原图的一致性,函数实现相对于函数原型中参数、返回值、异常处理的一致性等等。

完备性也应该是一个相对的概念,通常是相对于某种应用需求而言的。完备性与简单性经常会产生矛盾,应采用折中的方法获得结论。

## 5. 效率

效率是指关于诸如空间、时间、人力、财力等资源耗费的度量。例如,一个算法的空间和时间复杂性理论的评估。

可行性是表示某种预期的结果(如项目的完成或元件制作的完成)达到的效率,以及一个给定的实现过程较之替代的实现过程的效率。

对算法的时空效率进行分析是最常见的实例。但设计与实现算法的人力、财力等资源耗费经常会被忽略。销量好的像 Windows 一样的商品化软件投入再多人力、财力也在所不惜,但作为普通的应用软件当然不值得这样精益求精。

与其他商品的生产一样,软件生产不能单纯追求产品的性能,同样重要的是提高产品的性能价格比。软件产业追求的目标不仅仅是软件产品运行的效率,还包括软件产品生产的效率。

考虑效率的最佳方法是将多个因素综合起来,通过折中获得结论。

## 6. 演化

演化是指更改的事实、意义和应采取的对策。更改时各层次所造成的冲击,以及面对更改的事实,抽象、技术和系统的适应性及充分性。例如,形式模型随时间变化表示系统状况的能力,以及一个设计对环境要求的更改和供配置使用的需求、工具和设备的更改的承受能力。

演化要表达的实际上是生命周期的概念,软件设计活动贯穿了整个软件生命周期,包括各种类型的系统维护活动。

在工业生产的并行工程中采用了一系列称为 DFX 的技术,如 Design For Assembling、Design For Manufacturing 等,主张在设计阶段就全面考虑产品的整个生命周期。可以说,在软件开发中早就采用了与 DFX 类似的技术,毕竟在软件生命周期中维护期占的比例更大。

## 7. 抽象层次

抽象层次是指计算中抽象的本质和使用。在处理复杂事物、构造系统、隐藏细节及获取重复模式方面使用抽象,通过具有不同层次的细节和指标的抽象能够表示一个实体或系统。例如,硬件描述的层次、在目标层级内指标的层次、在程序设计语言中类的概念,以及在问题解答中从规格说明到编码提供的详细层次。

计算机学科为认知论带来的贡献并不多,相反它从其他学科(如数学、科学、工程等)中借用了许多思维方式,而这些思维方式是人类认识与改造世界的基本方法。

抽象是人类认知世界的最基本的思维方式之一。罗素曾断言:发现一对鸡、两昼夜都是数 2 的实例,一定需要花费很多年的时间,其中所包含的抽象程度确实不易达到;至于 1 是一个数的发现,也必定很困难。

近世代数又称抽象代数,其名称与思路均很好地体现了抽象这一思维方式。在许多具体的代数系统中数学家们抽取了群、环、域等抽象代数,而在这些抽象代数中抽取共性,还可提炼出更抽象的概念——范畴。范畴是这些抽象代数的一种抽象。

抽象源于人类自身控制复杂性能力的不足:我们无法同时把握太多的细节,复杂的问题迫使我们将这些相关的概念组织成不同的抽象层次。日常生活中的 is - a 关系是人们对概念进行抽象和分类的结果,例如苹果是一种水果,水果是一种植物等。生物学采用的界、门、纲、目、科、属、种标准生物分类方法是这一思维方式的经典应用。将这种 is - a 关系在程序中直接表达出来而形成的继承机制,是面向对象程序设计最重要的特征之一。



在软件设计中太容易找到不同的抽象层次,例如变量→类型、对象→类→ADT、实现→规格说明(程序正确性定义的相对性)、数据流图的分解与平衡等。从绑定这一概念的定义可看出,只有在不同抽象层次的前提下才会存在绑定。

## 8. 空间有序

空间有序是指在计算学科中局部性和近邻性概念。除了物理上的定位(如在网络和存储中)外,空间有序还包括组织方式的定位(如处理机进程、类型定义和有关操作的定位),以及概念上的定位(如软件的辖域、耦合、内聚等)。

正如画家或雕塑家在平面或立体上创造一件艺术杰作,空间有序追求的也是一种空间上的美感。这种美感会真实地存在(例如一份可读性极佳的源程序清单),但更多的是在思维空间之中(例如程序的结构或模块之间的关系)。计算机专业训练的目标之一是培养我们良好的审美观。

在软件领域中,这种美感小到程序中的一行注释,大到逻辑上与物理上的模块构成,乃至整个软件的体系结构。

软件工程师的桌面总是整洁的,因为他喜欢空间有序。桌面一塌糊涂的人可能是天才,但未必能成为一名合格的软件工程师。

## 9. 时间有序

时间有序是指按时间的先后所分成的顺序。这包括在形式概念中把时间作为参数(如在时态逻辑中)、把时间作为分布于空间的进程同步的手段、时间算法执行的基本要素。

时间有序作为一种和谐的美存在,其最大的特点是在生命周期中表现出的对称性:有对象创建就有对象消亡,有构造函数就有析构函数,有保存屏幕就有恢复屏幕,有申请存储空间就有释放存储空间。

时间有序与空间有序是天生的一对。程序中时间的有序应尽量与空间的有序保持一致,如果一个对象的创建与消亡分别写在两个毫无关联的程序段中,那么其潜在的危害性是可想而知的。

在并行与分布式系统中,时间有序占有更重要的地位。对并发的同时性考虑的角度不同,导致有两大类不同的并发模型:真并发模型与交错模型。并发系统中的顺序、并发、选择(冲突)、冲撞等现象均与时间有密切关联。

## 10. 重用

重用是指在新的情况或环境下,特定的技术、概念和系统成分可被再次使用的能力。例如,可移植性、软件库和硬件部件的重用,促进软件成分重用的技术,以及促进可重用软件模块开发的语言抽象等。

可能还没有哪一个行业的重用情况会像软件行业这样糟糕。有没有见过哪家汽车生产厂商自己采矿炼钢?有没有见过哪间家具厂自己种林伐木?但这种现象在软件行业却司空见惯。

软件重用的对象除源代码外,还包括规格说明、系统设计、测试用例等,软件生命周期中越前端的重用意义越重大。现有的许多努力都是面向源代码一级的重用,例如程序的模块化、封装与信息隐藏、数据抽象、继承、异常处理等机制,包括当前热门的 CORBA、DCOM 等利用构件组装软件系统的技术。

软件重用被认为是软件行业提高生产率的有效途径,然而许多技术与非技术因素阻碍了软件重用的应用与推广。从技术上看,只要形式化方法的研究(特别是作为理论基础的