CAD

程序设计基础

CAD
程序设计基础

裘春航　郭秀玲　刘军献　编著

上海交通大学出版社

内 容 简 介

本书旨在为工科院校非计算机软件专业的本科生、研究生和工程技术人员在进行本专业计算机应用及 CAD 开发等方面提供一本比较深入和系统的教材。这种针对工程应用软件开发的专著，过去并不多见。

本书共分五章，主要包括：程序设计逻辑基础、程序设计基本原理、数据结构、CAD图形学基础、工程数据库等内容。

本书的组织既注意到学科阐述的系统性同时也注意到前后内容、形式的一致性、连贯性，但各章内容又保持相对的独立性。全书以 CAD 软件系统开发的基本程序设计方法和原理为中心，并附有大量例题及可供实际应用的程序，反映了作者们多年的工作经验，对 CAD 软件开发是一本有实用价值的教材。

本书既可作为工科院校非软件专业的教学用书，也可供工程设计人员、计算机应用人员、研究生等使用和参考。

CAD
程 序 设 计 基 础
上海交通大学出版社出版
（淮海中路 1984 弄 19 号）
浙江上虞汤浦印刷厂排版
新华书店上海发行所发行
常熟市印刷二厂印装

开本 787×1092 毫米 1/16 印张 22.25 字数 543,000

1989年2月第1版 1989年2月第1次印刷

印数：1—3,700

ISBN 7—313—00291—2/TP·39 科技书目：180—252

定价：4.35 元

前　言

随着计算机技术的迅速发展，计算机的应用范围已由原来的单纯数值计算扩展到数据处理、工业控制、辅助设计及人工智能等多个领域。特别是计算机辅助设计技术的发展与应用，使得越来越多的工程设计人员加入到研制、开发和使用计算机应用软件的行列。为此，作者编写了本书，以便为非计算机软件专业的人员在进行计算机应用和CAD开发等方面提供一本比较深入和系统的教材。

本书以CAD软件系统开发的基本程序设计方法和原理为中心，并附有大量的例题及可供实际应用的程序。全书共分五章，第一章讨论了计算机系统和CAD的一般概况；第二章介绍了程序设计的逻辑基础和进行程序设计的基本原理，并且还给出了一个简单编译程序的例子；第三章论述了进行CAD程序设计所经常用到的数据结构；第四章对CAD图形学基础进行了介绍，包括曲线、曲面生成，图形变换，消隐线算法和微机图形支撑软件的开发等内容；第五章则对工程数据库管理系统进行了讨论，并介绍了一个适用于工程应用的数据管理软件。本书还包括两个附录，分别介绍了8088汇编语言和MS-Pascal语言的语法及使用，以便读者阅读书中程序。

本书所采用的大量例子具体说明了CAD程序设计的原理与方法，其中有些例子取自于行之有效的应用软件，期望这些例子对读者有所帮助。

大连理工大学机算机系黄耀富副教授为本书编写了概论的前三节，工程力学研究所赵永哲、黄杏萍、邓忠、曲春雷同志参加了部分段落的编写和程序调试，作者在此表示衷心感谢。

在本书的编写过程中，始终得到了钱令希教授、钟万勰教授的热情指导与鼓励，在此表示诚挚的感谢。

由于作者水平有限，缺点错误在所难免，望读者批评指正。

作　者

1988年12月

目　录

第一章 概 述

1.1 硬件和软件

计算机是能执行大量计算，包括许多算术运算和逻辑运算，而在运行期间无需操作员干预的一种功能装置。

电子计算机系统由计算机硬件和计算机软件组成。**计算机硬件**是指数据处理中使用的物理设备，它是计算机系统中的实际装置的总称。尽管计算机的种类繁多，但对于冯·诺依曼型计算机来说，它们的硬件组成和基本结构是相似的。图1.1是微型计算机简图。它的关键控制装置就是由运算器和控制器组成的中央处理器（简称 CPU）和内存储器。在这里，所有的判定都是通过程序指令的执行来实现的。这些指令被存储在内存储器中，而当 CPU 执行所需要的任务时就去取指令。**指令**是规定计算机操作类型及操作数地址的一组字符。大多数指令都是很基本的，只是做某种简单操作，比如，读取数据指令，它把数据从 CPU 的一个寄存器移到另一个寄存器，在运算器中执行某种算术运算或逻辑运算等等。存储器既存储指令，也存储数据。

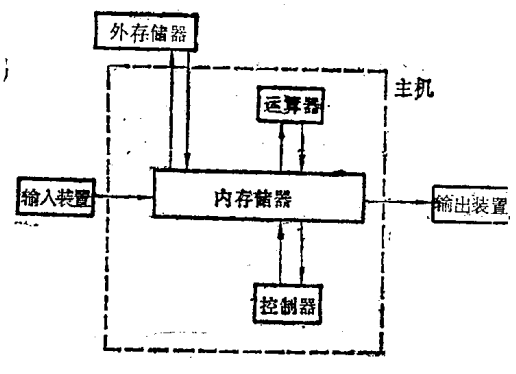

图 1.1 微型计算机简图

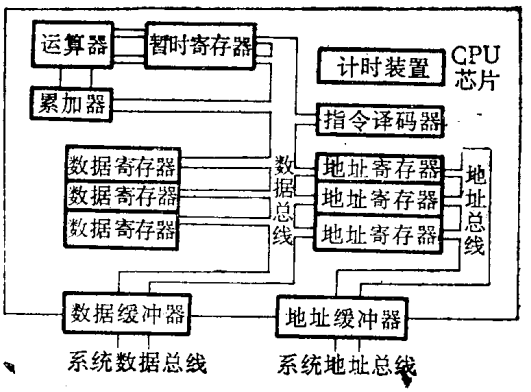

图 1.2 CPU芯片

典型的 CPU 芯片由几个逻辑部件所组成，如图 1.2所示。CPU 芯片上有运算器、累加器、临时寄存器、指令译码器、数据寄存器和地址寄存器等。

运算器是完成算术和逻辑操作的部件，所以又称算术和逻辑运算单元。运算器能够接收数据，并对接收的数据进行诸如加、减、乘、除等算术运算，以及“与”、“或”等逻辑运算。运算开始时，累加器用来存放一个操作数；运算结束时，累加器一般用来存放运算结果。

存储器是用来存放计算所需的原始数据、中间数据、输出之前的最终结果以及指示计算机对数据如何进行操作的指令的器件。存储器可根据实际需要把原来记录的内容抹去而重新记录新的内容，或把原记录的内容取出但不破坏原有的记录。在解题前，程序和原始数据通过

键盘和CPU送入**存储器**中保存起来。在运算过程中，存储器一方面不断地向运算器提供运算所需要的数据；另一方面还能保存从运算器送来的计算结果。

控制器则是从存储器顺序地取出指令，翻译指令代码，安排操作顺序，并向各部件发出相应的命令，使它们一步步执行程序所规定的任务。控制器是统一指挥和控制计算机各部件的中央机构，它一方面向各部件发出执行任务的命令，另一方面又接收执行部件向控制器发回的有关任务执行情况的返回信息。这种返回信息将对控制器下一步的工作状态产生重要的影响，控制器将把这些信息作为下一步发出哪些命令的工作条件，根据各种工作条件的成立与否，来决定下一步相应地发出哪几种命令。

输入装置是用来提供计算所需的数据和告诉计算机如何对数据进行操作的程序，以及对计算机下达各种对话式命令的装置。输入装置有字母数字键盘、卡片机和磁带机数字化仪等。

输出装置是用来输出计算的中间结果或最后结果，以及计算机对各种对话式命令所做出反应的装置。输出装置有屏幕显示、打印机和绘图机等。

中央处理器是运算器和控制器的总称。中央处理器又称为CPU。

计算机中暂时存放一组数据的存储装置叫做**寄存器**。这组数据可以是数、指令等。

总之，由中央处理器、存储器、输入装置和输出装置就组成计算机硬件。利用这样的计算机固然能执行计算任务，但是它只能识别并执行利用机器语言所写的程序。这种未配置任何软件的计算机称为**裸机**。直接使用裸机，不仅不方便，而且人的工作效率和机器的使用效率都非常低。为了运行、管理、维修和开发计算机就需要给它配置软件。**计算机软件**就是与计算机系统的操作有关的程序、过程、规则，以及与之有关的文档说明及数据。计算机软件可分为系统软件和应用软件。**系统软件**是一系列程序，它不同于普通的应用程序，这些程序的着眼点是利用计算机本身的逻辑功能，合理地组织整个解题和处理流程，简化或代替用户在各个环节上所承担的工作，达到充分开发计算机中的资源，最大限度地发挥计算机的效率，便于用户使用和管理、维修的目的。这些程序的总称就是**系统软件**。系统软件包括：各种语言的汇编程序、解释程序和编译程序、计算机的监控管理程序、调试程序、故障检查和诊断程序、程序库、操作系统等。用户利用计算机以及它所提供的各种系统软件，编制解决用户各种实际问题的程序，这些程序统称为**应用软件**。例如，工程设计方面的组合结构分析程序系统、CAD/CAM程序系统、军事对策、物资管理、工资管理、飞机订票、铁路局列车调度系统等等。

硬件提供计算机应用的基础；而软件扩大了计算机的功能。未配备软件的裸机所能执行的任务有很大的局限性。同一台微型计算机，如果配备上良好的软件之后，就仿佛变成了另一台全新的计算机，能表现出前所未有的功能。所以，硬件与软件的结合，才构成一个完整的计算机系统。

1.2 程序设计支撑环境

软件开发环境主要由硬件、系统软件、工具系统群、指导软件开发的各种方法、从事软件开发的技术人员和管理人员等构成。这与机械生产的生产环境由各种车床、车间、测量工具、图纸、从事机械生产的技术人员和工人等所构成的情况十分类似。

程序设计支撑环境有两重含义。广义地说，它指的是进行程序设计工作的计算机系统

条件(包括硬件与软件)。狭义地说，它指的是进行程序设计工作时的软件条件，即指操作系统、各种语言及其编译程序、解释程序、编辑程序、调试程序、验证程序等等。

也可以说，**程序设计支撑环境**是指通过单一命令语言来使用的工具的完整集合，用以提供在整个软件生命周期中的程序设计支持能力。环境典型地包括在设计、编辑、编译、装入、测试、配置管理及计划管理中所用的工具。

现在的电子计算机概念与早期的概念不同，都是以"计算机系统"的面貌出现的。在60年代和70年代初期，电子计算机，主要指的是"裸机"，那时所说的"计算机系统"，也无非是裸机加上各种外围设备而已。而现在所说的"计算机系统"主要指包括大量软件的硬件，即10%～20%是硬件，而80%～90%是软件。各种软件分属于两大层次，直接在硬件外面的是系统软件，包括操作系统、编译系统、各种服务程序等。包在系统软件外面的是应用软件，对计算机系统的应用是通过"用户界面"(或称"用户接口")进行的。一个应用软件的效率取决于"用户界面"的质量，好的"用户界面"称为"用户友好界面"。用户界面下面是另一层界面，即系统界面，这一层界面是由系统软件所提供的程序设计支撑环境(见图1.3)。显然，这种支撑环境的好坏，对应用软件系统的质量起着决定性的作用。

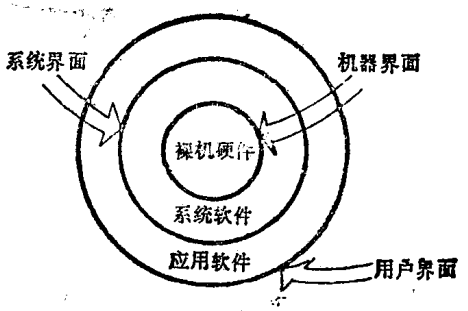

图1.3 计算机系统的三种界面

1.3 操作系统

计算机中的资源，是指计算机系统本身所提供的中央处理器时间、存储空间、输入输出设备以及计算机中的各种软件。

在50年代，计算机的工作基本上采用人工操作，用户将纸带装进光电输入机，把程序和数据输入，然后通过控制台开关启动程序。仅当程序执行完毕并取走纸带和计算结果后，才让下一个用户上机操作。这种人工操作方式具有两个弱点：

1. 用户独占全机　整台计算机为一个用户所独占，其全部资源由他一人支配。虽然该用户可以很方便地使用资源，不会出现资源为别的用户占用而等待的现象，但是资源利用率却非常低。这正如藏书超过百万册的图书馆在同一时刻只准一位读者进馆内看书，而众多的读者在门口排队等候的现象一样。

2. 中央处理器等待人工操作。

以上两点充分说明人工操作方式严重地影响资源的利用率。要知道，计算机的运算速度是极快的，它每秒钟可完成百万次的运算。随着计算机硬件的发展和机器运算速度的提高，这种人机矛盾越来越尖锐，计算机资源的浪费现象到了不能容忍的程度。

人们自然会想到，在一个用户输入数据或程序发生故障时，最好能让主机为别的用户

务，以提高计算机的资源的利用率。为此，又设计出一个关键性硬件——**通道**。而缓冲技术特别是脱机输入输出操作，又进一步缓和了CPU和I/O设备间速度的矛盾。早期的脱机输入操作是把许多用户的程序和数据在一台外围计算机的控制下，预先从低速输入设备输入到磁带上。这就意味着各个用户程序的处理顺序已经按先后次序事先排定。机器将首先处理磁带上的第一个作业，即把第一个程序和数据调入内存，然后执行该程序，并把控制权交给作业1。当把作业1完成后又把控制权交给系统，系统再去调入磁带上的第二个作业。按这种方式对磁带上的作业自动地一个接一个地依次进行处理，于是便形成了批量处理系统。当某个作业出现故障时就自动地过渡到下一个作业，这样就节省了中央处理器的时间。提高了计算机资源的利用率。为了对系统中的作业和计算机资源进行监督和管理，又相应地配置了监督程序和管理程序，它不仅是操作系统的前身，而且也是后来出现的操作系统的核心部分。

批量处理系统卓有成效地提高了计算机资源的利用率。但是一旦用户把其作业提交给系统后，便失去自己对作业的控制和修改能力。无论用户的程序多么小，也要同别的用户的大程序一起排队。当用户等候多时好不容易轮到机器开始处理自己的作业，可是偏偏自己的程序又被机器指出了语法错误。无论这个语法错误多么微小，也要在改正之后第二次在队尾耐心等待。如果又发现错误还要第三次排队。用户强烈希望机器能边运行他的程序，边告知处理情况，还希望能方便而即时修改他的程序。即系统应具有用户与系统以及用户与自己的程序交互作用的能力。为了满足用户的这种需要和进一步提高计算机资源的利用率，1960年设计出了世界上第一个分时系统。

让一个计算机系统的主机带上几个、十几个，甚至上百个终端，把中央处理器时间和内存空间按一定的时间间隔(即时间片)轮流切换给上述各个终端用户的程序使用。每个程序一次运行一个时间片，且都经由用户终端与一个用户相连，用户便可通过终端与之交互作用。虽然该计算机系统是由许许多多用户共享，但由于计算机响应的时间很短，应答很快，每个用户并不感觉到有别的用户存在，而好像整个计算机系统为他一个用户所独占，这样的系统称为**分时系统**。分时系统是操作系统的一种类型，也是对配备了分时操作系统的整个计算机系统的称呼。

分时系统的工作原理如图1.4所示。时间片由时钟中断控制，每次只把一个终端用户的

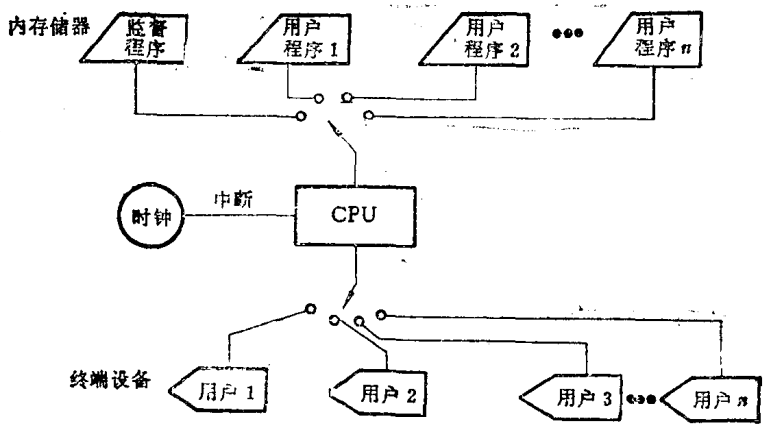

图1.4 分时系统的工作原理

程序放在内存并让它使用中央处理器，其他终端用户的程序均在外存等待。

分时系统基于人的操作和思考速度远比计算机的处理速度慢这一事实。这好比一个勤快的仆人，作到了一仆多主，由于他干活迅速俐落，而致使他的那许多主人们却都各自认为一仆一主。

分时系统的基本特征为：

1．同时性　许多用户能同时或基本上同时使用同一个计算机系统；

2．独立性　各用户可以彼此独立地操作，而不会相互混淆或破坏现象；

3．及时性　用户能在很短的时间内（小于3秒）获得对系统所提要求的回答；

4．交互作用性　用户能与系统进行人-机对话。

分时系统已能获得令人满意的机器利用率和响应时间，但是它仍不能满足某些领域的需要。例如，在把计算机用于飞机飞行和导弹发射等自动控制时，要求计算机能把由测量系统所测得的数据及时地加工处理，以便不误时机地对飞机或导弹进行控制。在计算机用于工业生产的控制时，例如钢铁的冶炼和轧制的自动控制，石油生产的自动控制，都要求计算机能对由传感器送来的数据及时地进行处理，然后去控制相应的执行机构，以使温度或压力等参数按一定的规律变化或恒定不变。这就需要另外一种类型的操作系统。

对实时系统进行管理和协调的操作系统，称为**实时操作系统**。所谓实时，是表示立即、现在，而实时系统是指系统对特定输入作出反应所具有的速度，足以控制发出实时信号的那个设备。或者说，计算机能及时响应外部事件的请求，在规定的时间限制内完成对该事件的处理，并控制所有实时设备和实时任务协调一致地运行。实时系统的突出特点就是一个“快”字，试想，导弹落地之后计算机才得出控制导弹如何飞行的数据，这还有什么意义呢？分时系统虽然对响应时间也有要求，不过它一般是以人所能接受的等待时间来确定的，因此响应时间的数量级通常规定为秒，比如3秒。而实时系统对响应时间的要求，则是以控制过程或信息处理过程所能接受的延迟时间来确定的，通常是秒的数量级，但在某些实时系统中对响应时间的要求，可达到毫秒甚至毫微秒的数量级。

总之，**操作系统**是给计算机配置的一个大型系统程序，它用来实现计算机自身的硬件和软件资源的管理。批量处理系统、分时系统和实时系统是操作系统的三种主要类型。

操作系统由许多程序所组成：

1．控制和管理处理机的程序；

2．控制和管理存储器的程序；

3．控制和管理输入和输出设备的程序；

4．控制和管理用户程序和数据的程序。

随着操作系统变得越来越精巧，它也变得越来越大了。1963年由麻省理工学院投入使用的一个分时系统约由32000个36位的存储单元组成。一年后由IBM公司推出的OS/360有100万条以上的机器指令。1975年由麻省理工学院和贝尔实验室研制的Multics系统增加到2000万条以上的指令。

操作系统为用户提供一套简单易记的操作命令，裸机配备有操作系统和其他系统软件之后，便成为一台既懂命令又懂各种高级语言、使用和操作十分方便的计算机系统。

在小型计算机和微型计算机出现之后，自然就考虑也要为它们配备操作系统。比起大型计算机来，这些计算机速度慢而且存储容量小，它们哪能容纳得下大型操作系统呢？为了把握

作系统压缩得比较小以适应小型机和微型机，操作系统的功能被划分开。几乎所有程序都需要的服务，如控制和管理，输入和输出的例行程序被放入计算机的核心程序，且运行时总是在内存里。其他程序称为系统的实用程序，这些程序存放在磁盘上，仅当需要时才读入内存。从过去几年中投入使用的操作系统来看，它好像是管理单机资源必须的最小核心，包括有几万条指令。

为微型计算机配备的操作系统有 MS-DOS、UNIX、CP/M 和 CDOS 等。通常，这些操作系统只有命令解释、输入输出设备驱动、文件管理等功能，在结构原理上也比大型计算机的操作系统要简单得多。

MS-DOS 是美国微软件(Microsoft)公司为 IBM-PC 微型计算机开发的通用 16 位单用户的磁盘操作系统，它也适用于以 8086/8088 微处理器为基础的一切微型计算机。它的结构优良，由命令处理模块、磁盘操作管理模块和输入输出接口模块这三层程序模块组成。这些模块都是以命令文件的形式驻留在磁盘上，当系统启动时，由启动程序将其装入内存。

UNIX 是美国贝尔实验室，起初在数据设备公司的 PDP-11 小型机上运行的通用交互式分时操作系统。它于 1970 年研制成功，1974 年公布第一版本，经过多次修改，但基本思想没有改变，现在流行的是第七版本。UNIX 及其变种目前正在众多的微、小、中及大型计算机上显示出其强大的生命力。它可移植性好、功能齐全、性能优良。UNIX 在 16 位微型机软件系统中占有独特的地位。

1.4 CAD 概况

所谓 CAD 系统是指由计算机硬件和软件所构成的体系。设计人员可借助于 CAD 系统进行制图，代替木模和胶模造型，对设计进行分析计算等，从而使他们有更多的精力用于创造性劳动，从繁重的分析、计算和手工绘图中解放出来。目前，CAD（计算机辅助设计）和诸如 CAM(计算机辅助制造)、CAE、CIM 等缩写字，已成为现代工业界的热门术语。

可供设计人员使用的硬件系统有集中式、多主机式、分布式和独立工作站等四种形式。如以使用较广泛的 32 位微机工程工作站来看，一般都配有较通用的 CPU，标准容量为 2～4 兆字节的内存储器(可进一步扩充容量)，分辨率约为 1000×1000 像素的图形显示器，百兆字节左右的硬磁盘，以及数字化仪、绘图仪等输入输出设备，如图 1.5 所示。应该指出，即使在16

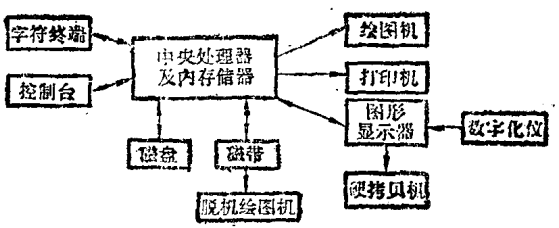

图 1.5 CAD工作站硬件配置

位微机，如 IBM-PC/XT、长城 0520 或高一档的 IBM-PC/AT 上建立 CAD 系统也是大有可为的。由于微机的功能将不断增强，据预测，到 1990 年使用微型计算机的 CAD/CAM 系

统将会超过 9/10。

CAD 系统除应该有操作系统(如 UNIX)等软件支持外，组成它的其他软件通常有：

(1) 工程数据库管理系统(EDBMS)；

(2) 图形处理软件，其中包括计算机辅助制图和造型软件；

(3) 包括前、后处理的有限元分析软件等。

在上述各类软件的支持下，可进一步开发各种具体的 CAD 应用软件。

在工程数据库管理系统的支持下，可建立相应的工程数据库，以便统一管理在设计全过程中的所有数据，实现数据一体化和数据共享。

为了处理图形数据，除了需要基本的图形支撑软件(如 GKS等)外，还要几何造型和计算机辅助制图等软件。

制图的基本形式是二维的，用点和线表示一个结构或产品。目前，市场上已有作为软件产品的计算机辅助绘图系统，提供标准的二维制图和绘图功能，大多数这样的软件都包括自动加注尺寸的子程序。

传统的造型要求制造一个线条或实体模型给出产品的形象，通常这是相当费钱的；而 CAD 系统中的造型软件，可预先在屏幕上显示出产品将具有的三维形象，且便于修改。三维对象可以用线框、曲面和实体这三种方法造型，其中以实体造型软件产生的图形最为逼真，它能给出设计对象在形状、表面、体积和密度等特性的完整描述。当然，实体造型要占用较大的计算机内存资源，通常约为 400K 字节。三维造型有多方面的应用，例如，可用于复杂结构的有限元几何模型的生成、机器人制造等。

分析计算对工程设计过程是必不可少的，有限元分析软件已日益成为分析计算的重要辅助手段。通常，有限元分析软件带有包括网格自动生成、绘制各种等值线图等功能的前后处理程序。

CAD 系统可通过交互方式用菜单和命令驱动。菜单被预先设计好并显示在屏幕上，命令驱动能给设计者以更大的灵活性。

第二章　程序设计基本原理

随着计算机技术的不断发展，计算机的应用领域和应用范围越来越广。计算机软件是由计算机程序和与之相应的各种文档组成，如软件研制说明书、设计说明书、使用说明书等都是程序文档的典型例子。计算机软件和硬件一样，作为计算机应用的基础都起着很大的作用。

本章将通过对软件生命期和软件质量，模块化程序设计，循环控制结构，程序设计逻辑基础，以及一个简单编译程序展开的例子，具体介绍程序设计的基本原理和基本思想。完整和清晰的程序文档对提高程序的可靠性，以及便于程序维护、开发、交流和推广应用等方面起着不可低估的作用。

2.1　程序设计各阶段和程序质量评价

2.1.1 程序设计各阶段

计算机程序是通过程序设计（Programming）而生成的。一般意义下，程序设计是包含下列内容的全过程：

1．说明(Specification)；

2．设计(Design)；

3．编程(Coding)；

4．调试(Testing)；

5．维护(Maintenance)。

程序的说明，指给程序的功能、即所完成任务的一个定义。对任意给定的问题，首先要对它进行需求性分析，得到关于“做什么”的描述。程序的说明是进行后继程序设计及编程和调试的最起码的前提。不难想象，如果我们不知道要做什么，我们就无法具体去做；而如果我们理解错了问题的需求，则也就不能设计出正确的程序。程序的说明可通过形式化和非形式化两种方法而得到。所谓形式化说明，是指用一种有精确定义的语法和语义的语言来表示程序的说明，形式化说明可以完全规定程序的抽象行为，对自动程序设计和正确性证明具有一定的帮助；但对绝大多数的程序设计问题来说，由于形式化说明过分强调细节和数学上的严密性，给出相应的形式化说明通常是困难的，有的甚至是不可能的。所以，通常采用的方法是非形式化说明。非形式化说明结合了自然语言和数学等各种形式的表达方式，虽然和形式化说明比较起来其严密性和逻辑性会差一些，但却更接近于人们的自然表达方式，便于表达程序说明，易于被人们理解，因此常常能更清楚地说明程序的主要功能。

程序的设计指对所涉及的算法与相应的数据结构，以及模块的划分与组织方式，作出逐步精化的考虑。在给出上述程序功能，即“做什么”说明的基础上，便可以考虑“怎么做”，即进入程序的设计阶段。设计一般分总体设计、模块设计等几个步骤，每个步骤所考虑的细节有所不