

新起点 电脑教程

C/C++ 程序设计基础教程 与上机指导

计算机职业教育联盟 主编

- 面向程序设计初学者，精心编排内容，入门容易
- 针对上课课时的需要，特别安排了合适的讲课内容和上机指导
- 大量的经典上机习题，可供学员和老师随时练习和参考



清华大学出版社

新起点电脑教程

C/C++程序设计基础教程 与上机指导

计算机职业教育联盟 主编

清华大学出版社
北京

内 容 简 介

本书编写的目的旨在提高C语言的教学质量,改善C语言难讲、难学、难以掌握的现状,着眼于培养学生的独立编程能力,书中按照难点分解、平滑过渡、循序渐进的思路组织结构框架和编配内容。各章均配有大量上机指导和习题,努力为课堂教学过渡到独立编程提供模拟环境和更大的知识惯性。本书分为五部分:第1~6章介绍了C语言程序设计的基础知识,这也是高级语言的共性和基础部分;第7~12章是C语言程序设计的提高部分,对C语言公认的难点(例如指针、结构体、预处理等)进行了详细的介绍和重点讲解;第13~14章为C语言高级编程部分,介绍了图形程序和字符处理程序的编写;第15和第18章为C语言程序设计的技巧部分,重点介绍了C语言的调试、编程风格以及部分编程的技巧;第16~17章介绍了C++的初步知识以及初步的编程方法,介绍C++对C的扩充以及面向对象编程的基础知识,以便让学生能快速地从C过渡到C++编程。

本书内容新颖,体系合理,逻辑性强,文字流畅,通俗易懂,是学习C和C++语言的理想教材,只要具有计算机的初步知识即可学会本书所介绍的内容,可作为大中专院校和各类计算机培训学校的专用教材,也可作为自学的参考用书。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

C/C++程序设计基础教程与上机指导/计算机职业教育联盟主编. —北京:清华大学出版社,2003
(新起点电脑教程)

ISBN 7-302-06797-X

I. C... II. 计... III. C语言—程序设计—教材 IV. TP312

中国版本图书馆CIP数据核字(2003)第054510号

出 版 者:清华大学出版社(北京清华大学学研大厦,邮编100084)

<http://www.tup.com.cn>

<http://www.tup.tsinghua.edu.cn>

策划编辑:林章波

文稿编辑:章忆文

印 刷 者:北京鑫丰华彩印有限公司

发 行 者:新华书店总店北京发行所

开 本:787×1092 1/16 印张:25.75 字数:580千字

版 次:2003年8月第1版 2003年8月第1次印刷

书 号:ISBN 7-302-06797-X/TP·5058

印 数:0001~5000

定 价:34.00元

序 言

一、中国职业培训行业面临的契机和挑战

中国高等职业教育和职业培训服务近年来取得了快速发展，为社会经济发展作出了巨大的贡献。中国正式成为世界贸易组织(WTO)成员后，整个中国教育界尤其是职业教育界都将面临一个新的契机和挑战。我国的职业教育主要包括学校职业教育和职业培训两种模式。学校职业教育基本是正规学历教育，已经不能适应目前经济的高速发展、职业多变和终身教育的需要。随着入世后就业结构调整和技术技能折旧速度的加快，劳动力跨行业流动更加频繁，职业培训进一步成为涉及面最广、受益面最大的教育，从而越来越成为职业教育的重要组成部分，越来越受到人们的广泛欢迎。

加入 WTO 对中国职业培训将产生积极的影响，至少表现在以下几个方面：

- 加快引进国外优质职业教育资源，有利于推动职业教育办学体制和运作机制改革，提高职业教育的整体水平；
- 有利于吸收国外多种职业教育模式，加快改进职业教育的办学模式、课程体系和教学培训方法，形成具有多元化特色的职业教育体系；
- 引进国外职业资格证书及其培训模式，加快完善我国的职业资格证书制度，有利于培训内容及培训资格的国际化；
- 经济增长和产业结构调整将扩大职业培训和继续教育的市场需求，有利于职业培训的持续发展；
- 加快教育的市场化步伐，有利于推进相关的法制建设进程，改变人们的传统教育观念，提高职业培训的社会地位。

同时，境外职业教育和培训机构的进入必将对职业教育领域造成巨大的冲击。境外职业教育与培训机构十分看好中国的职业培训和继续教育市场。此前由于教育服务业开放方面的原因，多为依托跨国公司在开展经济活动的同时建立培训机构，或通过职业资格证书培训来拓展职业教育市场，入世后放宽境外职业教育培训机构合作办学的条件，并允许其获得多数拥有权，这将大大加快其入境提供职业教育服务的步伐。他们凭借丰富的职业教育和培训经验，及其国际通用的职业资格证书，使中国职业教育服务领域形成新的竞争格局，一批办学条件差、专业设置陈旧及教学水平落后的职业教育和培训机构将被淘汰出局。

二、社会发展对计算机培训行业提出的新要求

IT 行业的发展和激烈的人才竞争对 IT 从业人员的综合素质提出了越来越高的要求，总结起来可以归纳为五种“基本能力”与三项“基本素质”。五种基本能力包括合理利用与支配各类资源的能力；处理人际关系的能力；获取并利用信息的能力；综合与系统分析能力；运用各种技术的能力。三项基本素质包括基本技能、思维方式与个人品质。在市场经

济体制和新技术飞速发展的条件下，计算机人才只掌握一门特定的技能已经远远不够。必须全面强调学生的基础知识、基本能力和基本素质，而且专业面要宽，以适应技术进步与市场的不断变化。例如电脑平面设计师，像2000年以前那样只掌握1~2种常用的平面设计软件已经远远不能胜任工作，还必须具备良好的基础和创意能力，这就对整个计算机培训行业提出了更高的要求。被广大培训机构广泛采用的只重知识传授，不重能力培养的教学模式已不能适应社会对计算机培训行业提出的要求。计算机培训必须进行从知识传授到能力培养的转变，全面提高学生的职业能力，即其在就业后的工作及配合中综合运用知识与技术的能力：能够做到迅速收集、分析、归纳意见与信息，与他人交流思想与信息，制定计划与组织活动，与他人合作共事，解决实际问题，学习和使用最新的技术，成为适合社会进步的人才。

三、计算机培训教学的新模式和新探索

社会对学生的职业能力要求催化出新型的课程结构和教学模式。新型教学模式必须以工作为基础的模仿学习，它是将学生置于一种模拟环境中，呈现给学生的是具有挑战性、真实性和复杂性的问题，使学生在身体和经济不受到损失的前提下，得到较真实的锻炼。新型课程结构必须按照职业能力的要求创建并组织实施新的教学模式。教学以专项能力的培养展开，以综合能力的形成告终，能力的培养既是教学目标，又是评估的依据和标准。在能力标准的制定、考核与评估中，均要有企业或行业参与。

新型课程结构在具体的实施中又派生出两大结构，即模块式结构与阶梯式结构。模块式结构，即按照程序模块化的构想和原则设计课程。其特点是每个模块均为独立的教学单元，有特定的目标和评估标准，教学时间相对较短。模块之间组合灵活，基础性模块可与其他功能模块组成任务更复杂的课程，不同模块的增删可调整课程结构的重心，以满足个人和职业需求的变化。上述这些特性适应了新型职业能力的教学要求，部分计算机培训机构已经开始尝试采用这种课程结构。

阶梯式结构最典型的就是所谓的双元培训，它把整个培训课程分为两个阶段，首先集中学习文化课和职业基础课，第二阶段转入职业领域的专业训练和专向深化。这种以较广泛的科学文化和职业理论为基础，逐步深化职业知识与技能的课程结构，在欧洲国家的职业教育体系中采用较多，在中国的计算机培训机构中基本上没有采用。

以上两种课程结构虽模式不同，但目标一致，即意在使学生以较宽厚的基础和实力去迎接未来多变且多元的就业市场。

长城计算机学校充分吸取了国外职业教育机构的经验，并对大量企业进行了科学广泛的调研分析，然后结合模块式和阶梯式两种结构的优点，从2001年起，设计推出了一系列面向就业的计算机培训课程，并且在教学实践过程中不断探索先进的教学方法，如发现式教学法、启发式教学法、体验式教学法、行为导向教学法、任务驱动教学法等，在大量的实践基础上不断完善课程。经过在各地分校和合作学校的长期推广使用，得到了巨大的成功。

清华大学出版社从建社伊始，就致力于高新技术，特别是计算机与信息技术类图书的出版。20多年来，出版社坚持和发展自己的优势与特色，围绕“开展全方位出版，形成总体优势”的指导方针制定选题规划，重点抓教材与专著的出版，形成品牌与规模。为了适

应信息技术飞速发展和广泛应用的特点，出版社及时出版了一大批计算机应用技术的图书并已形成品牌。为了适应各层次，各类专业人员的需要，出版社组织了门类齐全，层次丰富的多种系列丛书，同时配合电子音像出版物，做到全方位、立体化出版。在国内计算机图书的出版中，清华版图书占有明显的优势。清华版计算机图书以其内在的品质，已经在读者心目中竖起精品品牌，享有很高的声誉。

为了共同推进国内计算机培训行业的入世步伐，长城计算机学校与清华大学出版社亲密合作，共同开发了《新起点电脑教程》，在总结了百所分校的教学经验的基础上，把教学模式和教学成果推向市场，造福社会，并借之与广大教育、培训业专家交流经验，共同提高。本套丛书贯穿了清华大学出版社一贯的严谨、科学的图书风格，融会了长城计算机学校实用、专业的教学理念，是对计算机教学的一种深入、广泛的探索和总结，也得到了信息产业部领导和专家的首肯。我们相信，此套丛书的出版，必将对整个计算机培训市场的发展和规范起到巨大的推进作用！

四、丛书特点

《新起点电脑教程》丛书定位于计算机初、中级用户。不管是培训班学员还是大中专院校的师生，都可以通过本丛书快速进入计算机科学的大门，学到实用的计算机技能；对于自学者，本套教程也有很大的参考价值，大量实用技巧可供自学者在工作中随时参阅。

本套丛书主要具有如下几个方面的特点：

1. 系统性和丰富性

本丛书的所有课程都是围绕着职业素质训练展开的。我们根据计算机相关职业把计算机教程划分为四大类别：

- (1) **应用类**：主要面向广大计算机家庭用户、企事业单位的文员、秘书和行政助理、打字排版人员等电脑应用人员；
- (2) **设计类**：主要面向平面设计师、网页设计师、三维动画设计师等电脑设计专业人员；
- (3) **网络类**：主要面向网络管理员、系统集成工程师、安全工程师等网络类专业人员；
- (4) **编程类**：主要面向软件开发工程师、软件测试工程师等编程类专业人员。

以上四类内容基本涵盖了计算机应用的主要领域。本丛书的选题既考虑了每门课程本身的完整性，又兼顾了课程间的联系与衔接。每一本书可能都自成一体，完全满足相应课程的教学要求，使得培训学校或读者可以根据需要灵活的进行选择 and 组合，满足个性化学习的需要。不管读者是什么样的情况，都能在本丛书中找到自己需要的教程。

2. 先进性和探索性

一般的计算机教材往往只注重课程本身的体系结构和语言的科学准确，术语连篇，让读者感觉高深莫测，而这种做法实际上忽略了人的认知规律：人的学习应该是由具体到抽象、由已知到未知的过程。本丛书的编写充分的溶入了教育心理学的最新成果，在内容的安排和教授方法上进行了大胆的尝试。在介绍各个知识点时，本丛书大量的使用了教学中常用的比喻、类比和举例等方法，在内容的先后顺序上由浅入深、层次分明，使得知识的讲解变得深入浅出，易于理解。

按照成人教育的规律，本丛书每章前配有学习的目标以及重点难点，让读者有明确的目标，了解学习过程中应该重点掌握和比较难以理解以及容易混淆的知识点。在正文的实例演示的操作中，还在必要和适当地方加上“注意”、“技巧”等内容，以让初学者少走弯路，快速掌握到知识的精髓。

从教育心理学的角度来看，遗忘率最高的是知识，技能其次，能力最低。技能、实践必须从亲自动手中得来。老师讲的，学生往往只是一种平面的、苍白的理解。只有自己亲自动手实践的才是有色彩的、立体的、生动的理解和感受。真正的技能源自动手与实践！所以本丛书每章正文后面都配有上机指导，列举了大量实例，并给出操作步骤和实验结果，供读者上机练习时使用。另在每一章的最后安排了习题，包括填空题、判断题、选择题、简答题，操作题等各种形式，供课后练习和巩固知识之用。

另外，丛书的每一本教材都有配套的幻灯片，培训教师可以从网站上下载，满足教学的需要。每本书大约15~20章左右，符合短期培训的课时安排规律；每一章约15~20页，符合授课时的教学内容的量的需要，所以本套丛书完全可以满足各种教学情况。

3. 实用性和专业性

传统教学中通常存在理论强于实践、实践和应用脱节的缺点。而职业教育的特点要求必须加强对学生自学能力和实践动手能力的培养，在教学中鼓励学员探索和分析归纳创新习惯的养成。并且，将学生学习的技术技能同他将来的工作岗位紧密的结合起来。实现了学生学习与职业间的“无缝连结”。

本丛书的内容是在仔细分析初学者学习电脑的困惑和目前电脑图书市场现状的基础上确定的，一切围绕的实际应用和就业需要选择内容，丛书中选取的大量实例都是在企业的实际工作案例的基础上改编来的，让读者在学习每个知识点时能“避虚就实”，在练习实例的同时，就掌握了相应行业的职业规范，学到真正有用的东西。读者只要按书中的实例和上机指导的方法去做成、做会、做熟，就能举一反三，学以致用，从而能真正完成“专业”的作品和项目，就业时能够迅速进入工作岗位。

本套丛书从策划选题到编写校对，全部由多年从事计算机教学的专业老师和图书出版业专家协同完成，并经过业内资深专家审核。我们希望这里的每一本教程都能成为您在学习计算机的道路中的一个新的起点，也成为您人生道路上的一个新的起点！

前 言

C 语言是现代程序设计中最活跃的高级语言之一。在可视化语言出现之前，目前最流行的 Windows、UNIX、Linux 三大操作系统都是用 C 语言编写的。实际上，可视化语言能够实现的东西，用 C 语言几乎都可以实现。在高校教学过程中，C 语言的预期教学目标与实际教学效果存在较大的差距，C 语言难教、难学、难以掌握的现象普遍存在。因此，本书从 C 语言体系的框架结构及内容出发，认真分析和探讨提高 C 语言课程教学质量的途径与方法，努力减轻来自教与学两个方面“三难”压力。当然能否实现初衷，还需要通过实践检验。

本书的特点之一是对 C 语言公认的难点进行了分解，例如指针、结构体、函数、数组等，本着先易后难、平滑过渡、循序渐进、易读易懂的思路进行了整体结构设计。在结构体系方面充分注意到章节知识点的渐进性，前后表述的一致性和疑难点的可理解性。

本书的特点之二是针对解题难、实验难的问题，结合全国计算机等级考试和高校计算机等级考试编写了配有答案的大量习题，这对于深化许多重要概念，掌握各章节知识要点、解读知识难点会有很大的帮助，也会使读者节约许多时间。C 语言程序设计教学的最终目的是培养学生又快又好的编程能力。要缩短从课堂教学到转化为编程能力的时间差，最有效的方法就是阅读大量配有答案的习题，然后进行编程模拟训练，当编程模拟量积累到一定程度时，做题和编写程序就会很轻松。

本书的特点之三就是强调引导性教学，一方面强调教师精讲，另一方面强调学生完成自学内容，各章都针对所讲内容提出指导性的教学目标，并在每一章安排了上机指导。学生在学习各章节内容之前，教师应花 10~20 分钟时间讲解有关要点，引导学生应掌握哪些知识点，注意哪些问题。自学能力是大学生的重要培养目标之一。对自学部分的重要章节，建议至少要安排一次学生交流讨论的机会。

本书的特点之四就是安排了 C++ 的内容，由于 C 语言是结构化和模块化的语言，它是面向过程的。在处理较小模块的程序时，程序员使用 C 语言比较得心应手。但是当问题比较复杂、程序的规模比较大时，结构化程序设计就显出它的不足。C 程序的设计者必须细致地设计程序中的每一个细节。而 C++ 虽然是在 C 语言的基础上开发出来的，除了保留原有的优点外，增加了面向对象的机制。C++ 与 C 完全兼容，用 C 语言写的程序可以不加修改地用于 C++ 上，也可以混合编程。它既可以用于结构化程序设计，又可用于面向对象的程序设计。因此本书在 16 章和 17 章特别安排了 C++ 的内容。为进一步学习 C++ 内容打下良好的基础，当然，只有学好 C 语言，才能更进一步学习 C++ 语言。

对于课时比较充分的院校(60~80 课时)，课时分配计划建议为：课堂教学与上机实习为 40~50 课时，其中包括若干次课堂讨论。大型作业安排 3~4 个，需要 30~40 课时为宜。安排大型作业的目的是为了系统巩固全书各章知识的要点和重点。对于课时比较紧张的学校，16~18 章中 C++ 的内容和编程技巧的内容可以作为自学内容课后学习。

我们的愿望是好的，工作是努力的，但限于时间、水平和能力，书中难免存在不妥之处，恳请读者多加批评指正。

编者

2003年6月

目 录

第 1 章 程序设计基础	1	2.3.2 整型变量	24
1.1 程序设计的基本概念	1	2.3.3 实型变量	25
1.1.1 数据	1	2.3.4 字符变量	26
1.1.2 指令	1	2.3.5 变量初始化	27
1.1.3 程序	1	2.4 数值运算符与优先级	28
1.1.4 算法	2	2.4.1 算术运算符	28
1.1.5 程序设计语言	4	2.4.2 数值运算与优先级	28
1.2 程序设计方法	5	2.4.3 赋值运算符与表达式	30
1.2.1 结构化程序设计方法	5	2.4.4 逗号运算符与表达式	31
1.2.2 面向对象程序设计	5	2.5 类型转换	31
1.3 C 语言简介	6	2.6 自加 1、自减 1 运算	32
1.3.1 语言的发展历史	6	2.6.1 自增运算符	32
1.3.2 C 语言特点	6	2.6.2 自减运算符	33
1.3.3 学习 C 语言的必要性	6	2.7 数据的输入与输出	33
1.3.4 C 语言程序结构	7	2.7.1 #include 命令与头文件	34
1.4 Turbo C++编译器	9	2.7.2 格式输入与输出—— printf 与 scanf	35
1.4.1 Turbo C++集成开发环境	10	2.7.3 输出/输入	44
1.4.2 Turbo C++的命令行编译 连接	11	2.8 上机指导	44
1.5 上机指导	12	2.9 习题	46
1.5.1 使用 Turbo C++编译器编 译程序	12	第 3 章 逻辑运算	50
1.5.2 阅读和理解程序	14	3.1 关系运算符及表达式	50
1.6 习题	16	3.1.1 关系运算符及其优先次序	50
第 2 章 数据类型	18	3.1.2 关系表达式	51
2.1 C 语言的数据类型	18	3.2 逻辑运算符及表达式	52
2.2 常量	19	3.2.1 逻辑运算符及其优先顺序	52
2.2.1 一般常量	19	3.2.2 逻辑表达式	54
2.2.2 符号常量	20	3.3 if 语句及条件运算符	54
2.3 变量	22	3.3.1 if 语句	54
2.3.1 变量定义赋值	22	3.3.2 条件运算符	58
		3.3.3 开关语句	60
		3.4 上机指导	63

3.5 习题	66	5.6.1 计算 Fibonacci 数列	108
第 4 章 循环语句	70	5.6.2 找出二维矩阵中的最小值 及其位置	109
4.1 for 循环	70	5.6.3 身份验证	109
4.2 while 循环	73	5.7 习题	110
4.3 do-while 循环	76	第 6 章 函数	114
4.4 break、continue 和 goto 语句	77	6.1 函数的声明与定义	114
4.4.1 break 语句	77	6.1.1 函数声明	115
4.4.2 continue 语句	78	6.1.2 函数定义	116
4.4.3 goto 语句	79	6.2 函数的调用	117
4.5 几种循环的比较	80	6.2.1 函数的简单调用	117
4.6 上机指导	80	6.2.2 函数的参数传递	118
4.6.1 乘法表	80	6.3 函数的调用	123
4.6.2 输出国际象棋棋盘	81	6.3.1 函数的简单调用	123
4.6.3 求解最大公约数和最小 公倍数	82	6.3.2 函数的嵌套调用	124
4.7 习题	83	6.3.3 函数的递归调用	125
第 5 章 数组	90	6.4 函数的作用范围	127
5.1 一维数组	90	6.4.1 内部函数	127
5.1.1 一维数组的定义与赋值	90	6.4.2 外部函数	128
5.1.2 一维数组元素的引用	93	6.5 函数变量的作用域	129
5.1.3 一维数组应用举例	95	6.5.1 局部变量	130
5.1.4 一维数组的排序	95	6.5.2 形式参数	130
5.2 字符数组	97	6.5.3 全局变量	131
5.2.1 字符数组的声明与初值	97	6.6 变量的生存期	131
5.2.2 字符数组的引用	97	6.6.1 auto 类型	132
5.2.3 字符串操作	98	6.6.2 static 类型	133
5.3 二维数组与多维数组	102	6.6.3 extern 类型	134
5.3.1 二维数组的定义	102	6.6.4 register 类型	135
5.3.2 二维数组中元素的引用	102	6.7 上机指导	135
5.3.3 二维数组的初始化	103	6.7.1 定义函数求最大、 最小值	135
5.3.4 二维数组程序举例	104	6.7.2 定义函数求直角三角形 面积	136
5.4 多维数组	106	6.8 习题	137
5.5 字符数组应用举例	106	第 7 章 指针和地址	145
5.5.1 统计单词	106	7.1 指针变量的定义	145
5.5.2 输入三个字符串并找出其中 最大者	107	7.2 指针变量的引用	146
5.6 上机指导	108		

7.3 地址运算	148	10.3 结构数组和结构指针	204
7.3.1 运算方式	148	10.3.1 结构数组	204
7.3.2 指针和数组	148	10.3.2 结构指针	206
7.4 字符指针	153	10.4 结构的复杂形式	208
7.5 指针数组	155	10.4.1 嵌套结构	208
7.6 指针与函数	156	10.4.2 位结构	208
7.6.1 指针函数	156	10.5 链表	209
7.6.2 指针作为函数参数	158	10.5.1 单链表的概念	210
7.6.3 指向函数的指针	160	10.5.2 创建一个链表	212
7.7 上机指导	163	10.5.3 输出一个链表	213
7.7.1 利用变量指针进行排序	163	10.5.4 连接两个链表	214
7.7.2 求一个字符串的长度	164	10.6 上机指导	216
7.8 习题	165	10.7 习题	219
第 8 章 预处理指令	170	第 11 章 联合、枚举、类型说明	226
8.1 #define 指令	170	11.1 联合的使用	226
8.2 #error 指令	173	11.2 枚举类型的说明	228
8.3 #include 指令	174	11.3 类型声明方法	229
8.4 条件编译预处理指令	176	11.4 上机指导	231
8.5 #undef 指令	179	11.5 习题	235
8.6 上机指导	179	第 12 章 位运算	241
8.6.1 定义宏	179	12.1 概述	241
8.6.2 定义带参数的宏	180	12.1.1 字节与位	241
8.6.3 灵活应用#include	181	12.1.2 数的表达方式	242
8.7 习题	182	12.1.3 符号的表达方式	243
第 9 章 文件操作	186	12.1.4 电压的编码表示	244
9.1 标准文件函数	186	12.1.5 声音、图像的编码表示	244
9.1.1 文件的打开和关闭	186	12.2 位运算	244
9.1.2 有关文件操作的函数	188	12.2.1 按位与运算符“&”	245
9.2 非标准文件函数	193	12.2.2 按位或运算符“ ”	246
9.2.1 文件的打开和关闭	193	12.2.3 异或运算符“^”	247
9.2.2 读写函数	194	12.2.4 取反运算符“~”	248
9.2.3 随机定位函数	194	12.2.5 左移运算符“<<”	248
9.3 上机指导	194	12.2.6 右移运算符“>>”	249
9.4 习题	196	12.2.7 位运算符与赋值运算符 结合, 组成新的赋值 运算符	249
第 10 章 结构体	200	12.3 位段	249
10.1 结构声明和结构变量定义	200		
10.2 结构变量的使用	201		

12.4	上机指导	251	16.1.1	引用变量	311
12.5	习题	255	16.1.2	无名联合体	312
第 13 章	Turbo C++作图	258	16.2	运算的扩充	313
13.1	图形模式的初始化	258	16.3	函数功能的扩充	314
13.2	独立图形运行程序的建立	261	16.3.1	参数带默认值的函数	314
13.3	屏幕颜色的设置和清屏函数	262	16.3.2	函数重载	315
13.4	基本图形函数	263	16.4	对输入输出的扩充	318
13.4.1	画点	264	16.5	动态管理内存	320
13.4.2	画线	264	16.6	上机指导	321
13.4.3	任意封闭图形的填充	270	16.7	习题	324
13.4.4	有关图形窗口和图形屏幕 操作函数	271	第 17 章	面向对象编程基础 ——类与对象	327
13.4.5	图形模式下的文本 输出	273	17.1	概述	327
13.5	上机指导	277	17.1.1	对象	327
13.6	习题	281	17.1.2	类	327
第 14 章	字符屏幕	285	17.1.3	继承和派生	328
14.1	文本窗口的定义	285	17.1.4	类的属性和方法	328
14.2	文本窗口颜色的设置	286	17.1.5	类的构造与析构	328
14.3	窗口内文本的输入输出函数	288	17.1.6	成员类型	328
14.3.1	窗口内文本的输出函数	288	17.2	类与对象的定义	328
14.3.2	窗口内文本的输入函数	288	17.3	类的构造与析构	332
14.4	有关屏幕操作的函数	289	17.3.1	构造函数	332
14.5	上机指导	290	17.3.2	带参数的构造函数	334
14.6	习题	292	17.3.3	重载的构造函数	335
第 15 章	程序调试	293	17.3.4	类的析构函数	337
15.1	Turbo C++集成调试器	293	17.4	类的继承与派生	339
15.2	Turbo C++编译、连接和运行时 的常见错误	295	17.4.1	派生类的声明	339
15.2.1	编译时的常见错误	296	17.4.2	成员覆盖	340
15.2.2	连接时的常见错误	300	17.4.3	派生类的构造函数	341
15.2.3	运行时的常见错误	300	17.4.4	从多个类继承	342
15.3	上机指导	303	17.5	上机指导	342
15.4	习题	305	17.6	习题	346
第 16 章	在 C 基础上 C++的扩充	311	第 18 章	C/C++语言编程格式 与技巧	349
16.1	简单数据类型的扩充	311	18.1	C/C++语言的编程格式	349
			18.2	C 语言“switch 语句”的 使用技巧	351

18.3	“=”的使用技巧	352	18.6	上机指导	364
18.4	排序	353	18.7	习题	369
18.4.1	快速排序	353	附录 A	ASCII 码表	377
18.4.2	整数希尔排序	355	附录 B	C 语言中的关键字	378
18.4.3	整数堆排序	356	附录 C	部分库函数	379
18.5	单链表的基本运算	357	附录 D	参考答案	386
18.5.1	查找	357			
18.5.2	插入(后插)	359			
18.5.3	删除	361			

第 1 章 程序设计基础

教学提示：设计是一种创造性的工作，同样，程序设计也是一种创造性的工作。计算机语言和平常我们使用的语言不同，只有计算机通过特定的方式才能识别。用来进行程序设计的计算机语言很多，本书将重点介绍 C 程序设计语言。C 语言是现代比较流行的程序设计语言之一，它特有的指针类型的数据，可以方便地实现对内存的访问，从而使得程序能更加方便地对硬件进行控制。另外，C 语言同样可以应用在其他领域。正是由于拥有如此强大的功能，使掌握 C 语言已经成为一个程序员必备的能力之一。

教学目标：通过本章的学习，读者能够了解 C 语言的基本结构，程序设计的基本方法，以及 Turbo C++ 编译器的使用。

1.1 程序设计的基本概念

C 语言是 C 高级语言的简称。要想学习、使用 C 语言，就必须了解程序设计的基本知识。程序设计是一门专业的课程，本书不可能详细地介绍，而只是向读者介绍有关程序设计的基本概念，以便方便读者学习 C 语言。

1.1.1 数据

凡是能够被计算机接收、识别、处理的一切符号都可称之为数据。“1234”是数据，一张图片也可以做数据，一支歌，一串有确切含意的字符串，如“北京”、“WTO”也是数据。根据数据的不同属性，数据又可以划分为不同的类型，如能够进行算术运算的叫做数值型数据；能够说明事物某个特征的一串字符叫字符型数据。如“人民大会堂”、“UFO”。

1.1.2 指令

在计算机中，指令和命令是同义语(尽管有时候两者有一点差异)。它指示计算机进行什么操作和操作什么。进行操作的符号叫做操作码，操作对象叫操作数。所以一条指令包含操作码和操作数两个部分。如

```
sum=1+2;
```

这是一条 C 程序指令，该指令省略了操作码，但包括了加法运算和赋值运算两步操作，1、sum、2 是指令的操作对象。

1.1.3 程序

程序可以理解为“符号+规则”，用规定的符号和符合规则的一组有意义的指令集合称

之为程序。为了解决同一个问题而编写的若干个程序的集合叫做程序集。在程序集中，首先被执行的程序叫主程序，在主程序执行期间被调用的程序称为子程序。在程序定义中的“有意义”强调了程序要解决问题的目的性。

1.1.4 算法

算法是为了解决某个特定的问题而采取的确定的步骤(步骤有限)。算法是程序设计的灵魂，算法加数据结构等于程序设计。比如今天晚上要做一个红烧肉，可以描述的算法是：先到菜市场去买肉，回来洗干净切好，准备调料，生火制作；中间可能还需要看菜谱，没有的还要去买菜谱或者到网上查找相关信息，这就需要用到判断；另外，还需要针对每一步骤进行细分，如如何到菜市场买菜，怎么去，买多少；这就涉及到结构化设计的问题。算法有以下3个特性：

1. 有穷性

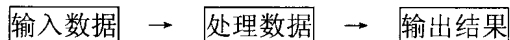
一个算法通过若干步骤即可实现预定目标，算法的每一步都可以在合理的时间内完成。

2. 确定性

算法中的每一条指令都必须有确切的含义，如果具有同样的步骤、同样的条件，则结果也是相同的。

3. 输入和输出

计算机工作大体由3步组成：



计算机工作的3个步骤具有不可逆的方向性。简单和复杂的计算机都是这样的，也就是说一个算法再简单，也要有输入、输出、处理3个步骤。

算法是可以描述的，描述算法的常用工具是伪代码和流程图。

所谓伪代码是指用自然语言和指令语句(不要求绝对正确的语句)结合起来描述算法的一种方法。这种方法与画流程图相比省时省力，转换为程序容易，但是清晰度与层次性不如流程图。伪代码描述算法没有统一的规定，写出来只要自己或别人看懂即可。

例如下面的伪代码描述的是某系统的主控制模块，可以看出伪代码比较接近自然语言，没有严格的语法，所以比较容易为人所理解，而且能较快地得出相应的程序清单。

```
PROCEDURE 模块名()  
清屏；  
显示 xx 系统用户界面；  
PUT ("请输入用户口令：")；  
GET (password)；  
IF password<>系统口令  
    提示警告信息；  
    退出运行；  
ENDIF
```



```

显示本系统主菜单;
WHILE (true)
    接收用户选择 ABC;
    IF ABC="退出"
        Break;
ENDIF
调用相应下层模块完成用户选择功能;
ENDWHILE
清屏;
RETURN
END

```

传统流程图是描述算法的一种常用工具，常由图 1.1 中所示的几种基本图形组成。

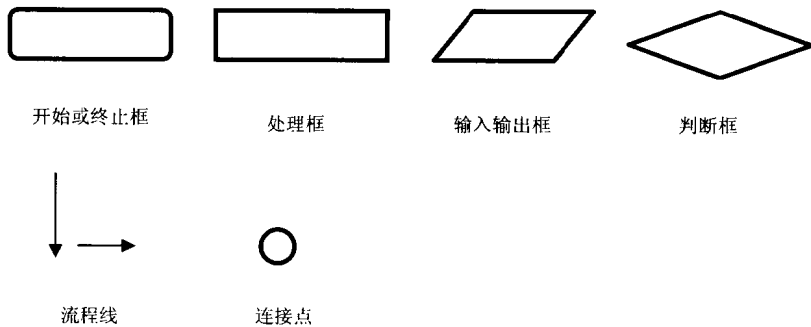


图 1.1 传统流程图的组成元素

通常用由这些框和流程线组成的流程图来表示算法，允许任意转向、形象直观且简单方便。但是，使用这种流程图描述复杂的算法时，所占篇幅较多，绘制费时费力，且不易阅读。

N-S 流程图是由美国学者 I·Nassi 和 B·Shneiderman 于 1973 年提出来用于描述算法的一种形式。算法的每一步都用一个矩形来描述。把一个个的矩形框按执行次序连接起来就构成了一个完整的算法描述。N-S 流程的特点是取消了流程线，禁止任意转向，描述复杂算法时和传统流程图相比所占篇幅少得多，绘制时也省时省力，层次性、易读性要好于传统流程图。

N-S 流程图常用矩形框表示，如图 1.2~图 1.5 所示。

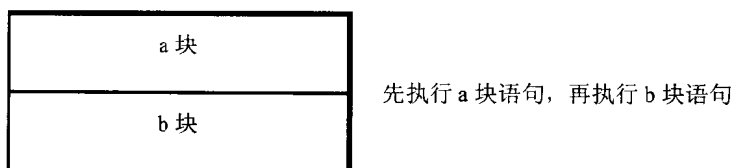


图 1.2 顺序结构框