

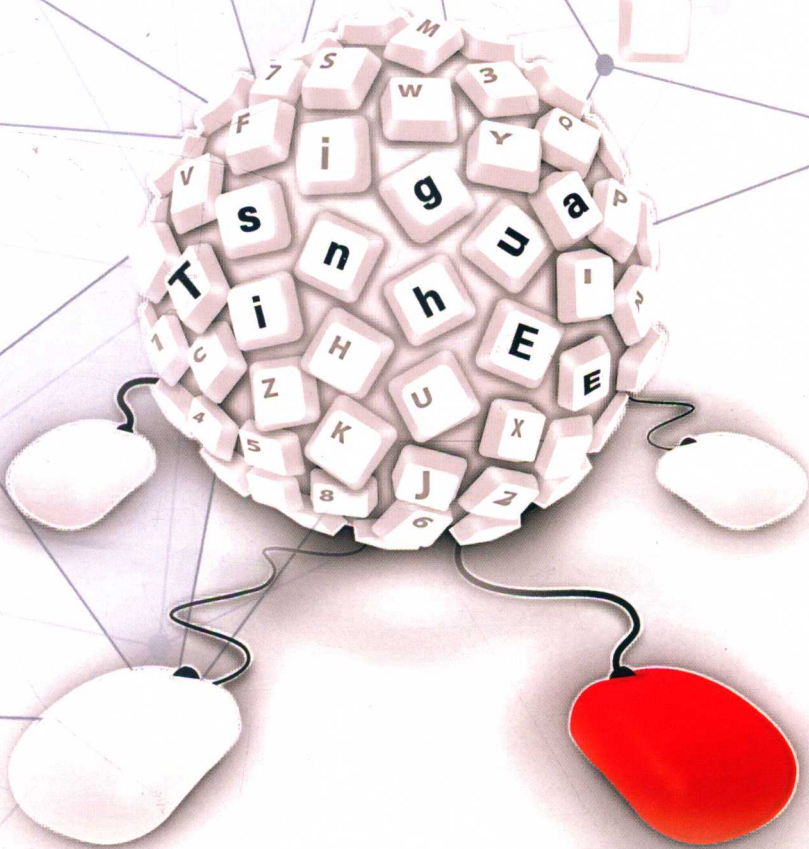
清华大学电子工程系核心课系列教材



C/C++ Programming

C/C++程序设计教程

◎黄永峰 孙甲松 编著



清华大学出版社

清华大学电子工程系核心课系列教材



C/C++ Programming

C/C++ 程序设计教程

◎黄永峰 孙甲松 编著

清华大学出版社
北京

内 容 简 介

本书是作者根据长期教学实践编写而成的。全书内容由浅入深,逐步介绍 C/C++ 语言中的基本概念和语法,使读者全面而系统地理解和掌握用 C/C++ 语言进行程序设计的方法。

本书叙述简明扼要,通俗易懂,例题丰富,有利于读者自学。本书可作为各专业的学生学习 C/C++ 语言程序设计的教材。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C/C++ 程序设计教程/黄永峰,孙甲松编著. —北京:清华大学出版社,2019

(清华大学电子工程系核心课系列教材)

ISBN 978-7-302-52690-2

I. ①C… II. ①黄… ②孙… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2019)第 057425 号

责任编辑:文 怡

封面设计:台禹微

责任校对:李建庄

责任印制:沈 露

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:三河市铭诚印务有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:44

字 数:1123 千字

版 次:2019 年 6 月第 1 版

印 次:2019 年 6 月第 1 次印刷

定 价:89.00 元

产品编号:072064-01

清华大学电子工程系经过整整十年的努力,正式推出新版核心课系列教材。这成果来之不易!在这个时间节点重新回顾此次课程体系改革的思路历程,对于学生,对于教师,对于工程教育研究者,无疑都有重要的意义。

高等电子工程教育的基本矛盾是不断增长的知识量与有限的学制之间的矛盾。这个判断是这批教材背后最基本的观点。

当今世界,科学技术突飞猛进,尤其是信息科技,在 20 世纪独领风骚数十年,至 21 世纪,势头依然强劲。伴随着科学技术的迅猛发展,知识的总量呈现爆炸性增长趋势。为了适应这种增长,高等教育系统不断进行调整,以把更多新知识纳入教学。自 18 世纪以来,高等教育响应知识增长的主要方式是分化:一方面延长学制,从本科延伸到硕士、博士;一方面细化专业,比如把电子工程细分为通信、雷达、图像、信息、微波、线路、电真空、微电子、光电子等。但过于细化的专业使得培养出的学生缺乏处理综合性问题的必要准备。为了响应社会对人才综合性的要求,综合化逐步成为高等教育主要的趋势,同时学生的终身学习能力成为关注的重点。很多大学推行宽口径、厚基础本科培养,正是这种综合化趋势使然。通识教育日益受到重视,也正是大学对综合化趋势的积极回应。

清华大学电子工程系在 20 世纪 80 年代有九个细化的专业,20 世纪 90 年代合并成两个专业,2005 年进一步合并成一个专业,即“电子信息科学类”,与上述综合化的趋势一致。

综合化的困难在于,在有限的学制内学生要学习的内容太多,实践训练和课外活动的时间被挤占,学生在动手能力和社会交往能力等方面的发展就会受到影响。解决问题的一种方案是延长学制,比如把本科定位在基础教育,硕士定位在专业教育,实行五年制或六年制本硕贯通。这个方案虽可以短暂缓解课程量大的压力,但是无法从根本上解决知识爆炸性增长带来的问题,因此不可持续。解决问题的根本途径是减少课程,但这并非易事。减少课程意味着去掉一些教学内容。关于哪些内容可以去掉,哪些内容必须保留,并不容易找到有高度共识的判据。

探索一条可持续有共识的途径,解决知识量增长与学制限制之间的矛盾,已是必需,也是课程体系改革的目的所在。

学科知识架构是课程体系的基础,其中核心概念是重中之重。这是这批教材背后最关键的

布鲁纳特别强调学科知识架构的重要性。架构的重要性在于帮助学生利用关联性来理解和重构知识；清晰的架构也有助于学生长期记忆和快速回忆，更容易培养学生举一反三的迁移能力。抓住知识架构，知识体系的脉络就变得清晰明了，教学内容选择就会有公认的依据。

核心概念是知识架构的汇聚点，大量的概念是从少数核心概念衍生出来的。形象地说，核心概念是干，衍生概念是枝、是叶。所谓知识量爆炸性增长，很多情况下是“枝更繁、叶更茂”，而不是产生了新的核心概念。在教学时间有限的情况下，教学内容应重点围绕核心概念来组织。教学内容中，既要有抽象的概念性的知识，也要有具体的案例性的知识。

梳理学科知识的核心概念，这是清华大学电子工程系课程改革中最为关键的一步。办法是梳理自1600年吉尔伯特发表《论磁》一书以来，电磁学、电子学、电子工程以及相关领域发展的历史脉络，以库恩对“范式”的定义为标准，逐步归纳出电子信息科学技术知识体系的核心概念，即那些具有“范式”地位的学科成就。

围绕核心概念选择具体案例是每一位教材编者和教学教师的任务，原则是具有典型性和时代性，且与学生的先期知识有较高关联度，以帮助学生从已有知识出发去理解新的概念。

三

电子信息科学与技术知识体系的核心概念是：信息载体与系统的相互作用。这是这批教材公共的基础。

1955年前后，斯坦福大学工学院院长特曼和麻省理工学院电机系主任布朗都认识到信息比电力发展得更快，他们分别领导两所学校的电机工程系进行了课程改革。特曼认为，电子学正在快速成为电机工程教育的主体。他主张彻底修改课程体系，牺牲掉一些传统的工科课程以包含更多的数学和物理，包括固体物理、量子电子学等。布朗认为，电机工程的课程体系有两个分支，即能量转换和信息处理与传输。他强调这两个分支不应是非此即彼的两个选项，因为它们都基于共同的原理，即场与材料之间相互作用的统一原理。

场与材料之间的相互作用，这是电机工程第一个明确的核心概念，其最初的成果形式是麦克斯韦方程组，后又发展出量子电动力学。自彼时以来，经过大半个世纪的飞速发展，场与材料的相互关系不断发展演变，推动系统层次不断增加。新材料、新结构形成各种元器件，元器件连接成各种电路，在电路中，场转化为电势（电流电压），“电势与电路”取代“场和材料”构成新的相互作用关系。电路演变成开关，发展出数字逻辑电路，电势二值化为比特，“比特与逻辑”取代“电势与电路”构成新的相互作用关系。数字逻辑电路与计算机体系结构相结合发展出处理器（CPU），比特扩展为指令和数据，进而组织成程序，“程序与处理器”取代“比特与逻辑”构成新的相互作用关系。在处理器基础上发展出计算机，计算机执行各种算法，而算法处理的是数据，“数据与算法”取代“程序与处理器”构成新的相互作用关系。计算机互联出现互联网，网络处理的是数据包，“数据包与网络”取代“数据与算法”构成新的相互作用关系。网络服务于人，为人的认知系统提供各种媒体（包括文本、图片、音视频等），“媒体与认知”取代“数据包与网络”构成新的相互作用关系。

以上每一对相互作用关系的出现，既有所变，也有所不变。变，是指新的系统层次的出现和范式的转变；不变，是指“信息处理与传输”这个方向一以贯之，未曾改变。从电子信息的角度看，场、电势、比特、程序、数据、数据包、媒体都是信息的载体；而材料、电路、逻辑（电路）、处

理器、算法、网络、认知(系统)都是系统。虽然信息的载体变了,处理特定的信息载体的系统变了,描述它们之间相互作用关系的范式也变了,但是诸相互作用关系的本质是统一的,可归纳为“信息载体与系统的相互作用”。

上述七层相互作用关系,层层递进,统一于“信息载体与系统的相互作用”这一核心概念,构成了电子信息科学与技术知识体系的核心架构。

四

在核心知识架构基础上,清华大学电子工程系规划出十门核心课:电动力学(或电磁场与波)、固体物理、电子电路与系统基础、数字逻辑与 CPU 基础、数据与算法、通信与网络、媒体与认知、信号与系统、概率论与随机过程、计算机程序设计基础。其中,电动力学和固体物理涉及场和材料的相互作用关系,电子电路与系统基础重点在电势与电路的相互作用关系,数字逻辑与 CPU 基础覆盖了比特与逻辑及程序与处理器两对相互作用关系,数据与算法重点在数据与算法的相互作用关系,通信与网络重点在数据包与网络的相互作用关系,媒体与认知重点在媒体和人的认知系统的相互作用关系。这些课覆盖了核心知识架构的七个层次,并且有清楚的对应关系。另外三门课是公共的基础,计算机程序设计基础自不必说,信号与系统重点在确定性信号与系统的建模和分析,概率论与随机过程重点在不确定性信号的建模和分析。

按照“宽口径、厚基础”的要求,上述十门课均被确定为电子信息科学类学生必修专业课。专业必修课之前有若干数学物理基础课,之后有若干专业限选课和任选课。这套课程体系的专业覆盖面拓宽了,核心概念深化了,而且教学计划安排也更紧凑了。近十年来清华大学电子工程系的教学实践证明,这套课程体系是可行的。

五

知识体系是不断发展变化的,课程体系也不会一成不变。就目前的知识体系而言,关于算法性质、网络性质、认知系统性质的基本概念体系尚未完全成型,处于范式前阶段,相应的课程也会在学科发展中不断完善和调整。这也意味着学生和教师有很大的创新空间。电动力学和固体物理虽然已经相对成熟,但是从知识体系角度说,它们应该覆盖场与材料(电荷载体)的相互作用,如何进一步突出“相互作用关系”还可以进一步探讨。随着集成电路发展,传统上区分场与电势的条件,即电路尺寸远小于波长,也变得模糊了。电子电路与系统或许需要把场和电势的理论相结合。随着量子计算和量子通信的发展,未来在逻辑与处理器和通信与网络层次或许会出现新的范式也未可知。

工程科学的核心概念往往建立在技术发明的基础之上,比如目前主流的处理器和网络分别是面向冯·诺依曼结构和 TCP/IP 协议的,如果体系结构发生变化或者网络协议发生变化,那么相应地,程序的概念和数据包的概念也会发生变化。

六

这套课程体系是以清华大学电子工程系的教师和学生的基本情况为前提的。兄弟院校可以参考,但是在实践中要结合自身教师和学生情况做适当取舍和调整。

清华大学电子工程系的很多老师深度参与了课程体系的建设工作,付出了辛勤的劳动。在这一过程中,他们表现出对教育事业的忠诚,对真理的执着追求,令人钦佩!自课程改革以来,特别是2009年以来,历届清华大学电子工程系的本科同学也深度参与了课程体系的改革工作。他们在没有教材和讲义的情况下,积极支持和参与课程体系的建设工作,做出了重要的贡献。向这些同学表示衷心感谢!清华大学出版社多年来一直关注和支持课程体系建设工作,一并表示衷心感谢!

王希勤

2017年7月

前言

程序设计是每个科技工作者使用计算机的基本功。C/C++语言是目前使用非常广泛的一种程序设计语言,具有丰富的数据类型,它所提供的数据结构和控制结构适合于进行结构化程序设计,并且利用 C/C++语言可以实现汇编语言的大部分功能,使用灵活,可移植性好。

本书不仅详细介绍了 C/C++语言的语法规则,而且对某些功能的系统实现以及程序的执行过程也做了必要的分析。本书着重强调 C/C++语言的基本概念,通过大量的例题分析和程序实例,使读者理解和掌握利用 C 语言进行程序设计的方法。

书中所有程序都经过实际调试。每章后面安排足够多的练习,并且在最后几个练习中提出了编程的具体要求,通常也可以作为实验使用。

本书的特点是简明扼要,通俗易懂,例题丰富。

由于作者水平有限,书中难免存在错误和不妥之处,恳请读者批评指正。

最后,在本书编写过程中得到了许多同事和同学的帮助,特别是陶怀舟博士和齐伊宁博士等。同时本书的编写内容也参考了多部国内相关的教材,由于篇幅限定,无法一一列出,在此,一并表示感谢。

作者

2019年3月

目 录

上 篇

第 1 章 绪论	3
1.1 程序设计概述	3
1.2 程序设计语言	9
1.3 简单的 C 语言程序	11
1.4 C 语言程序的上机步骤	14
练习 1	15
第 2 章 C 语言基本数据类型	17
2.1 数据在计算机中的表示	17
2.1.1 计算机记数制	17
2.1.2 计算机中数的表示	24
2.2 常量与变量	31
2.3 基本数据类型常量	32
2.3.1 整型常量	32
2.3.2 实型(浮点型)常量	34
2.3.3 字符型常量	36
2.4 基本数据类型变量的定义	36
2.4.1 整型变量的定义	36
2.4.2 实型变量的定义	40
2.4.3 字符型变量的定义	41
练习 2	43
第 3 章 数据的输入与输出	44
3.1 格式输出函数	44
3.1.1 基本的格式输出语句	44
3.1.2 printf() 函数中常用的格式说明	49
3.1.3 使用 printf() 函数时的注意事项	52
3.2 格式输入函数	53

3.2.1 基本的格式输入语句	53
3.2.2 scanf 函数中常用的格式说明	56
3.2.3 通过 scanf 函数从键盘输入数据	57
3.3 字符输出函数	59
3.4 字符输入函数	60
练习 3	61
第 4 章 C 语言表达式与宏定义	63
4.1 赋值运算及其表达式	63
4.2 算术运算及其表达式	64
4.3 关系运算及其表达式	65
4.4 逻辑运算及其表达式	68
4.5 其他运算符	71
4.5.1 增 1 与减 1 运算符	71
4.5.2 sizeof 运算符	72
4.5.3 逗号运算符	74
4.6 标准函数	75
4.7 宏定义	75
4.7.1 符号常量定义	75
4.7.2 带参数的宏定义	77
4.7.3 带 # 的宏定义	80
练习 4	81
第 5 章 选择结构	86
5.1 语句与复合语句	86
5.2 if 语句	89
5.3 if...else 结构	94
5.4 条件运算符	99
5.5 switch 结构	102
5.6 程序举例	110
练习 5	112
第 6 章 编译预处理	116
6.1 文件包含命令	116
6.2 条件编译命令	118
6.3 #pragma 命令	123
6.4 #line 命令	124
第 7 章 循环结构	126
7.1 当型循环与直到型循环	126

7.2	while 语句	127
7.3	do...while 语句	129
7.4	对键盘输入的讨论	133
7.5	for 语句	137
7.6	循环的嵌套与其他有关语句	139
7.6.1	循环的嵌套	139
7.6.2	break 语句	142
7.6.3	continue 语句	143
7.7	程序举例	146
	练习 7	158
第 8 章	模块(函数)设计	161
8.1	模块化程序设计与 C 函数	161
8.1.1	模块化程序设计的基本概念	161
8.1.2	函数的定义	162
8.1.3	函数的调用	165
8.2	模块间的参数传递	169
8.2.1	形参与实参的结合方式	169
8.2.2	局部变量与全局变量	172
8.2.3	动态存储变量与静态存储变量	174
8.2.4	内部函数与外部函数	179
8.3	模块的递归调用	181
8.4	程序举例	187
	练习 8	191
第 9 章	数组	194
9.1	数组的基本概念	194
9.2	数组的定义与引用	196
9.2.1	一维数组	196
9.2.2	二维数组	197
9.2.3	数组的初始化	198
9.3	字符数组与字符串	202
9.3.1	字符数组的定义与初始化	202
9.3.2	字符串	203
9.3.3	字符数组与字符串的输入与输出	204
9.3.4	字符串处理函数	207
9.4	数组作为函数参数	210
9.4.1	形参数组与实参数组的结合	210
9.4.2	二维数组作为函数参数	213
9.5	程序举例	216

练习 9	223
第 10 章 指针	227
10.1 指针变量	227
10.1.1 指针的基本概念	227
10.1.2 指针变量的定义与引用	228
10.1.3 指针变量作为函数参数	232
10.1.4 指向指针的指针	235
10.2 指针数组	237
10.3 数组与指针	238
10.3.1 一维数组与指针	238
10.3.2 二维数组与指针	241
10.3.3 数组指针作为函数参数	244
10.4 动态内存的申请与释放	249
10.4.1 malloc()函数	249
10.4.2 calloc()函数	253
10.4.3 realloc()函数	253
10.4.4 free()函数	254
10.5 字符串与指针	255
10.5.1 字符串指针	255
10.5.2 字符串指针作为函数参数	260
10.5.3 strstr 函数	263
10.6 函数与指针	263
10.6.1 用函数指针变量调用函数	263
10.6.2 函数指针数组	266
10.6.3 函数指针变量作为函数参数	266
10.6.4 返回指针值的函数	268
10.7 main 函数的形参	269
10.8 程序举例	271
练习 10	272
第 11 章 结构体与联合体	276
11.1 结构体类型变量	276
11.1.1 结构体类型变量的定义与引用	276
11.1.2 结构体的嵌套	279
11.1.3 结构体类型变量的初始化	280
11.1.4 结构体与函数	282
11.2 结构体数组	284
11.2.1 结构体类型数组的定义与引用	284
11.2.2 结构体类型数组作为函数参数	287

11.3	结构体与指针	289
11.3.1	结构体类型指针变量的定义与引用	289
11.3.2	结构体类型指针作为函数参数	290
11.3.3	结构体的大小与 #pragma 中 pack 的关系	294
11.4	链表	297
11.4.1	链表的基本概念	297
11.4.2	链表的基本运算	300
11.4.3	多项式的表示与运算	304
11.5	联合体	309
11.6	枚举类型与自定义类型名	311
11.6.1	枚举类型	311
11.6.2	自定义类型名	313
练习 11		315
第 12 章	文件	318
12.1	文件的基本概念	318
12.1.1	文本文件与二进制文件	318
12.1.2	缓冲文件系统	319
12.1.3	文件类型指针	320
12.2	文件的基本操作	321
12.2.1	文件的打开与关闭	321
12.2.2	文件的读写	322
12.2.3	文件的定位	330
12.2.4	文件缓冲区的清除	333
12.2.5	文件指针错误状态的清除	337
12.3	程序举例	339
练习 12		343
第 13 章	位运算	346
13.1	二进制位运算	346
13.2	位段	351
13.3	程序举例	354
练习 13		356

下 篇

第 14 章	C++ 类与对象	361
14.1	从 C 语言到 C++ 语言	361
14.2	面向对象的程序设计方法	361
14.3	类与对象	365

14.3.1	类和对象的概念	365
14.3.2	类的声明方法	366
14.3.3	对象的定义	367
14.3.4	对象的引用	368
14.4	类的成员函数	369
14.4.1	成员函数的声明与定义	370
14.4.2	inline 成员函数	371
14.4.3	成员函数的存储方式和 this 指针	373
14.5	类的构造函数和析构函数	374
14.5.1	对象的初始化与构造函数	374
14.5.2	构造函数类型	375
14.5.3	构造函数的重载	377
14.5.4	使用默认参数的构造函数	379
14.5.5	类的析构函数	380
14.6	对象数组和对象指针	385
14.6.1	对象数组	385
14.6.2	对象指针	386
14.7	共用数据的保护	388
14.7.1	常对象及常成员函数	389
14.7.2	对象的常数据成员	390
14.7.3	指向对象的常指针和指向常对象的指针变量	390
14.7.4	对象的常引用	392
14.7.5	const 型数据的小结	393
14.8	同类对象间的数据共享及类的静态成员	394
14.8.1	类的静态数据成员	394
14.8.2	类的静态成员函数	396
14.9	对象的动态构建和释放	398
14.10	对象的赋值和复制	399
14.11	友元和友元类	404
14.11.1	友元函数	404
14.11.2	友元类	406
14.12	应用程序举例——公司人事管理系统	407
	本章小结	411
	练习 14	411
第 15 章 运算符重载		414
15.1	运算符重载的一般方法	414
15.2	运算符重载函数作为类的成员函数或友元函数	417
15.2.1	双目运算符的重载应用举例	419
15.2.2	单目运算符的重载举例	421

15.2.3	流插入运算符和流提取运算符的重载	426
15.3	类对象与标准类型数据之间的转换方法	429
15.3.1	转换构造函数	429
15.3.2	基于运算符重载机制的类型转换函数	430
15.3.3	综合程序举例	432
15.4	综合程序应用——某公司人事管理系统	446
	本章小结	450
	练习 15	450
第 16 章	继承与派生	452
16.1	继承与派生的概念	452
16.2	派生类的声明方式及其组成	453
16.2.1	派生类的声明方式	453
16.2.2	派生类的组成	454
16.3	派生类成员的访问属性与继承方式	454
16.3.1	公用继承方式	455
16.3.2	私有继承方式	457
16.3.3	保护成员和保护继承	459
16.3.4	使用 using 声明来改变基类成员在派生类中的访问属性	461
16.4	派生类的构造函数和析构函数	462
16.4.1	单继承的派生类构造函数	462
16.4.2	内嵌子对象的派生类的构造函数	464
16.4.3	多级派生类的构造函数	466
16.4.4	派生类构造函数的定义规则总结	467
16.4.5	派生类的析构函数	468
16.5	多重继承与虚基类	468
16.5.1	多重派生类构造函数的定义方法	468
16.5.2	多重继承中同名数据引起的二义性问题	471
16.5.3	虚基类	472
16.6	继承与组合	478
16.7	综合程序应用——公司人事管理系统	483
	本章小结	509
	练习 16	509
第 17 章	多态性与虚函数	513
17.1	基类与派生类的对象兼容关系	513
17.2	多态性与虚函数	516
17.2.1	多态性概念及类型	516
17.2.2	虚函数	517
17.2.3	虚析构函数	521

17.2.4	多重继承中的虚函数	524
17.3	纯虚函数与抽象类	526
17.3.1	纯虚函数的概念	526
17.3.2	抽象类	527
17.3.3	应用实例	528
17.4	综合程序举例	531
	本章小结	532
	练习 17	532
第 18 章	输入输出流	535
18.1	C++的输入和输出流类	535
18.1.1	C++输入输出的类别和特点	535
18.1.2	C++输入输出流和流类	535
18.2	标准的输出流输入流	537
18.2.1	标准输出流	537
18.2.2	标准输入流	543
18.3	文件 I/O 操作与文件流	548
18.3.1	文件类型和文件流	548
18.3.2	文件流类与文件流对象	548
18.3.3	ASCII 文件的读写操作	550
18.3.4	二进制文件的读写操作	552
18.4	字符串流的输入和输出	555
18.5	综合程序应用——公司人事管理系统	558
	本章小结	558
	练习 18	559
第 19 章	C++语言工具	560
19.1	函数模板和类模板	560
19.1.1	函数模板	560
19.1.2	类模板	563
19.2	异常处理	569
19.2.1	异常处理的概念	569
19.2.2	异常处理的方法	570
19.2.3	函数声明中的异常情况指定	575
19.3	命名空间	576
19.3.1	命名空间的概念和作用	576
19.3.2	同名冲突解决方法与命名空间	578
19.4	C++语言的函数库	583
19.5	C++标准模板库	585
19.5.1	STL 的组件以及关系	586

19.5.2	容器	586
19.5.3	迭代器	589
19.5.4	算法与函数对象	592
19.6	综合程序应用——某公司人事管理系统	596
	本章小结	675
	练习 19	675
附录 A	基本 ASCII 码表.....	677
附录 B	C 语言常用库函数	680