

高等院校计算机教育规划教材

C++程序设计

王春玲 编著



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

高等院校计算机教育规划教材

C++程序设计

王春玲 编著

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书作者结合自己丰富的教学经验和编写教材的经验,针对大学生的知识结构和特点,精心策划、组织了本书的内容结构,力求使其成为高等院校 C++语言程序设计的理想教材。

本书的主要内容包括 C++语言概述,数据类型、运算符和表达式,基本控制结构,函数,数组、指针与引用,类和对象,继承和派生,虚函数与多态性,运算符重载,模板以及 C++流等。

本书结构体系合理,内容循序渐进、深入浅出、重点突出,语言叙述清晰、通俗易懂、逻辑性强。本书既适合作为高等学校非计算机专业的教材,也可以作为 C++程序设计爱好者的自学教材。

图书在版编目(CIP)数据

C++程序设计/王春玲编著. —北京:中国铁道出版社,
2007.7

高等院校计算机教育规划教材
ISBN 978-7-113-08073-0

I. C… II. 王… III. C语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2007)第 109722 号

书 名: C++程序设计

作 者: 王春玲

出版发行: 中国铁道出版社(100054,北京市宣武区右安门西街8号)

策划编辑: 严晓舟 秦绪好

责任编辑: 翟玉峰

特邀编辑: 王 胜

封面制作: 白 雪

责任校对: 贾 星

印 刷: 三河市国英印务有限公司

开 本: 787×1092 1/16 印张: 17 字数: 389 千

版 本: 2007年8月第1版 2007年8月第1次印刷

印 数: 1~5 000 册

书 号: ISBN 978-7-113-08073-0/TP·2431

定 价: 25.00 元

版权所有 侵权必究

本书封面贴有中国铁道出版社激光防伪标签,无标签者不得销售
凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社计算机图书批销部调换。

高等院校计算机教育规划教材

编委会名单

主任：毛汉书

副主任：李凤霞 严晓舟 杨小平

编委：（按姓氏音序排列）

古梅 雷光复 刘丽 秦绪好 孙俏

史秀璋 宋旭明 王金红 徐秋红 訾秀玲

赵轩 朱丽萍

出版说明

自 20 世纪 80 年代开始,随着我国教育事业的不断发展,在非计算机专业的大学生中,普及计算机知识与应用技能的计算机基础教育正在不断发展和完善。目前,在我国普通高校计算机基础教育中,普遍采用的是 1+X 课程体系,即开设一门涉及计算机(信息)知识面较广泛的计算机基础公共课和若干门有一定深度的计算机应用技术课的课程体系。为了适应高校广大师生对计算机教材的需求,本社特邀请(按学校名称拼音排序)北方工业大学、北京城市学院、北京理工大学、北京联合大学、北京林业大学、中国人民大学、中国政法大学及社内一些专家组成编委会,邀请有丰富教学经验,并且从事计算机基础教育多年的教师编写了这套高等院校计算机教育规划教材。本系列教材的特点是:

1. 读者对象明确,内容符合教育部对大学计算机教育的基本要求

教材主要面向普通高校非计算机专业的低年级学生。教材以应用为目的,以 1+X 方式组织教学内容,以适应不同专业的需求。本系列教材主要包括有关程序设计、数据库技术、多媒体与网络技术、硬件技术等方面的基础知识。编写中统筹考虑了主要知识点在各课程中的分布与比重,满足了教育部提出的对大学计算机教学的基本要求,为学生今后结合自身所学专业,进一步学习和应用计算机技术打下坚实基础。

2. 贯彻以学生为主的教学思想

目前,刚入学的多数大学生已在中学学习了一些计算机信息技术方面的基础知识,但由于各种原因,实际知识水平不齐。为了满足部分学生在达到本校计算机课程学习目标的同时,也能参加校外各种计算机等级考试的需求,本系列教材中,作为选修部分涵盖了《全国计算机等级考试考试大纲(2004 年版)》的主要内容,特别在习题与实验指导书中增加了这方面有针对性的训练内容。

3. 提供立体化教材,方便教与学

教材包括主讲教材、辅助教材和自学辅导光盘,另外,也可以从我社的网站免费下载相关课程的电子课件及习题答案。

主讲教材采用任务驱动的教学模式,引导学生在完成若干任务的过程中学习计算机基本理论与技术知识,每一教学单元的前面都有相关知识点提示。

辅助教材包括习题课与实验课的主要内容。教材中用大量实例来讲解分析问题和解决问题的方法。

自学辅导光盘包括我社专为参加计算机等级考试的读者出版的模拟考试环境软件,读者可以通过自学,熟悉某课程考试的一般过程,通过试做光盘中提供的多套模拟考题获得一定的考试经验。

4. 广泛采用专家研究成果

我国的计算机基础教育经过 20 多年的研究与实践,在教育理念、教学目标、教学内容和教学方法等各个方面研究取得了丰硕成果。本系列教材在编写过程中,广泛吸取了教育部有关计算机教育的教学指导委员会、全国高等院校计算机基础教育研究会、教育部考试中心等单位的专家意见和研究成果,以确保教材的科学性、先进性和实用性。在此对各位专家表示衷心的感谢。

由于计算机技术发展迅速，读者需求面广，一本书或一套教材显然不能满足广大读者的需求。因此我社将继续本着对读者高度负责的态度，不断推出定位准确、内容翔实、技术先进、符合教学规律的计算机教材，为我国计算机教育事业的发展及我国的信息化建设做出贡献。

欢迎广大读者对本系列教材的不足之处提出宝贵的批评和建议。

中国铁道出版社

2005年7月

前 言

本书是高等院校计算机教育规划教材之一，不仅适合面向对象的程序设计语言 C++ 课程的教学，同时也可以作为全国计算机等级考试二级 C++ 语言的参考书籍。

C++ 语言是在 C 语言的基础上逐渐发展起来的，是目前应用较广的一种优秀的高级程序设计语言。C++ 语言包括 C 语言的全部特征和优点，并且弥补了 C 语言的一些不足，同时添加了许多新特征和新功能，它既保留了对传统的结构化程序设计方法的支持，同时又添加了对面向对象程序设计方法的完全支持，是一种具有代表性的面向对象的程序设计语言。本书既适合具有一定 C 语言基础的读者学习，同时也适合未学过 C 语言，但有一定程序设计基础的读者学习。

本书全面系统地讲述了 C++ 语言的基本概念、语法知识和面向对象编程的方法，并结合作者多年的教学经验对 C++ 语言面向对象的有关概念（类和对象、数据封装、继承和派生、虚函数和多态性等）进行了详尽的介绍。本书在编写过程中力求做到结构体系合理，内容循序渐进、深入浅出、重点突出，语言叙述清晰、通俗易懂、逻辑性强。

全书共分为 11 章，主要包括：

第 0 章主要介绍 C++ 的产生与发展、C++ 的特点、C++ 程序的基本框架、C++ 的词汇以及 C++ 程序的开发过程；第 1 章主要介绍 C++ 语言的数据类型、常量、变量、运算符和表达式以及数据类型转换。本章是 C++ 语言的语法基础；第 2 章主要介绍 C++ 程序的基本控制结构，重点介绍 C++ 的各种语句的使用方法；第 3 章主要介绍函数，不仅包括函数的基本概念、定义、声明、调用等的实现方法，还包括函数的递归调用、函数重载、内联函数等知识；第 4 章主要介绍构造数据类型，其中包括数组、指针、引用、结构体和共用体；第 5 章主要介绍面向对象程序设计中的两个最基本的概念：类和对象。重点是如何建立一个类以及类的使用方法；第 6 章主要介绍继承和派生，通过继承可以在已有类的基础上派生新的类，从而实现了软件重用；第 7 章主要介绍虚函数和多态性。数据封装、继承性和多态性共同组成了面向对象程序设计的三大机制；第 8 章主要介绍运算符重载的意义以及实现运算符重载的方法；第 9 章主要介绍函数模板和类模板的概念、定义以及使用；第 10 章主要介绍 C++ 流的概念、数据输出的格式控制和文件流。

另外，本书每一章节针对重点语法都给出了详细的实例加以说明，对基本概念的讲解简明易懂；每一章结束都对本章重点内容进行总结，并分别给出相应的习题，以进一步强化巩固所学的知识。

本书在编写过程中得到了毛汉书教授、陈志泊副教授、王旻龙先生、孟伟老师的帮助和支持，另外，张慧萍、黄金燕等参加了课件的制作工作，在此表示衷心的感谢！

由于编者水平有限，书中存在的疏漏和不妥之处在所难免，恳请广大读者提出宝贵意见。作者电子邮箱：wang_chunling@126.com。

编 者

2007 年 4 月

目 录

第 0 章 C++语言概述.....	1
0-1 C++的产生与发展.....	1
0-2 C++的特点.....	2
0-3 C++程序的基本框架.....	2
0-3-1 最简单的 C++程序.....	2
0-3-2 结构化程序设计框架.....	3
0-3-3 面向对象程序设计框架.....	4
0-4 C++的词汇.....	6
0-4-1 基本符号.....	6
0-4-2 保留字.....	7
0-4-3 标识符.....	7
0-4-4 常量和变量.....	8
0-4-5 运算符和表达式.....	8
0-4-6 标点符号.....	8
0-5 C++程序的开发过程.....	9
0-5-1 Visual C++ 6.0 集成开发环境简介.....	9
0-5-2 程序的编辑.....	10
0-5-3 程序的编译.....	10
0-5-4 程序的连接.....	10
0-5-5 程序的运行.....	10
0-5-6 程序的调试.....	10
0-5-7 一个简单程序的开发过程.....	11
0-6 小结.....	13
思考与练习题.....	13
第 1 章 数据类型、运算符和表达式.....	15
1-1 数据类型.....	15
1-1-1 基本数据类型.....	15
1-1-2 类型修饰符.....	16
1-2 常量.....	16
1-2-1 逻辑常量.....	17
1-2-2 整型常量.....	17
1-2-3 实型常量.....	17
1-2-4 字符常量.....	18
1-2-5 字符串常量.....	18
1-2-6 符号常量.....	19
1-2-7 枚举常量.....	19
1-3 变量.....	20

1-3-1	变量的概念	20
1-3-2	变量的定义	20
1-3-3	变量的使用	21
1-3-4	常变量	22
1-4	运算符和表达式	23
1-4-1	运算符和表达式简介	23
1-4-2	赋值运算符和赋值表达式	23
1-4-3	算术运算符和算术表达式	25
1-4-4	自增、自减运算符	26
1-4-5	关系运算符和关系表达式	26
1-4-6	逻辑运算符和逻辑表达式	27
1-4-7	条件运算符和条件表达式	28
1-4-8	逗号运算符和逗号表达式	29
1-4-9	sizeof 运算符	29
1-4-10	位运算符	30
1-4-11	运算符的优先级和结合性	31
1-5	数据类型转换	31
1-5-1	隐式类型转换	31
1-5-2	显式类型转换	32
1-6	小结	32
	思考与练习题	33
第 2 章	基本控制结构	35
2-1	C++语句概述	36
2-2	顺序结构	36
2-2-1	声明语句	36
2-2-2	表达式语句	37
2-2-3	复合语句	37
2-2-4	空语句	38
2-2-5	基本输入/输出语句	38
2-2-6	顺序结构举例	40
2-3	选择结构	41
2-3-1	if 语句	41
2-3-2	switch 语句	44
2-3-3	选择结构举例	46
2-4	循环结构	47
2-4-1	while 语句	48
2-4-2	do-while 语句	49
2-4-3	for 语句	50
2-4-4	循环的嵌套	51
2-4-5	三种循环语句的比较	53

2-4-6 循环结构举例.....	53
2-5 跳转语句.....	54
2-5-1 goto 语句.....	54
2-5-2 break 语句.....	55
2-5-3 continue 语句.....	56
2-5-4 return 语句.....	56
2-6 小结.....	57
思考与练习题.....	57
第 3 章 函数	62
3-1 函数定义.....	62
3-2 函数的返回值.....	63
3-3 函数调用.....	63
3-3-1 函数调用形式.....	64
3-3-2 调用函数的前提条件.....	64
3-3-3 函数定义与函数声明的区别.....	65
3-3-4 函数的嵌套调用.....	65
3-4 函数参数.....	66
3-4-1 形式参数与实际参数.....	66
3-4-2 参数的传递方式.....	66
3-4-3 有默认参数值的函数.....	67
3-5 函数的递归调用.....	68
3-6 内联函数.....	70
3-7 函数重载.....	71
3-7-1 函数重载定义.....	71
3-7-2 匹配重载函数的顺序.....	72
3-7-3 定义重载函数时的注意事项.....	73
3-8 变量的作用域和存储类别.....	73
3-8-1 变量的作用域.....	73
3-8-2 变量的存储类别.....	77
3-9 编译预处理.....	78
3-9-1 宏定义命令.....	78
3-9-2 文件包含命令.....	79
3-9-3 条件编译命令.....	79
3-10 小结.....	81
思考与练习题.....	81
第 4 章 数组、指针与引用	84
4-1 数组.....	84
4-1-1 一维数组.....	84
4-1-2 二维数组.....	87
4-1-3 字符数组.....	90

4-1-4	字符串函数.....	92
4-2	指针.....	95
4-2-1	指针与地址.....	95
4-2-2	const 指针.....	98
4-2-3	指针与数组.....	99
4-2-4	指针与字符串.....	102
4-2-5	指针数组和指向指针的指针.....	103
4-3	引用.....	104
4-3-1	引用的定义.....	104
4-3-2	引用的使用.....	104
4-3-3	用引用作为函数的参数.....	106
4-3-4	如何使一个被调函数同时返回多个值.....	107
4-4	结构体和共用体.....	108
4-4-1	结构体.....	108
4-4-2	共用体.....	111
4-5	堆内存的分配和释放.....	112
4-5-1	堆内存的分配.....	112
4-5-2	堆内存的释放.....	113
4-6	小结.....	114
	思考与练习题.....	114
第 5 章	类和对象.....	118
5-1	类.....	118
5-1-1	类的定义.....	118
5-1-2	类成员的访问权限.....	119
5-1-3	类的数据成员.....	120
5-1-4	类的成员函数.....	120
5-2	对象.....	123
5-2-1	对象的定义.....	123
5-2-2	对象指针的定义.....	123
5-2-3	对象的成员.....	123
5-2-4	this 指针.....	126
5-2-5	对象的作用域与生存期.....	126
5-3	构造函数和析构函数.....	127
5-3-1	构造函数.....	128
5-3-2	构造函数的重载.....	129
5-3-3	默认构造函数.....	130
5-3-4	有默认参数的构造函数.....	130
5-3-5	拷贝构造函数.....	132
5-3-6	析构函数.....	134
5-4	静态成员.....	135

5-4-1	静态数据成员	135
5-4-2	静态成员函数	137
5-5	友元	139
5-5-1	友元函数	139
5-5-2	友元类	142
5-6	常成员	143
5-6-1	常对象	143
5-6-2	常数据成员	144
5-6-3	常成员函数	145
5-7	对象数组	147
5-8	成员对象	148
5-9	小结	151
	思考与练习题	152
第 6 章	继承和派生	163
6-1	继承的概念	163
6-2	派生类的定义	164
6-2-1	单继承的定义	164
6-2-2	多重继承的定义	164
6-2-3	派生类的成员构成	165
6-3	派生类对基类成员的访问	165
6-4	派生类的构造函数和析构函数	170
6-4-1	单继承方式下派生类的构造函数	170
6-4-2	多重继承方式下派生类的构造函数	175
6-4-3	有子对象的派生类的构造函数与析构函数	177
6-4-4	派生类的析构函数	179
6-5	多重继承的二义性	180
6-5-1	多重继承的二义性问题	181
6-5-2	多重继承的二义性问题的解决方法	182
6-6	虚基类	183
6-7	小结	186
	思考与练习题	187
第 7 章	虚函数与多态性	192
7-1	多态性	192
7-1-1	多态的类型	192
7-1-2	多态的实现	193
7-1-3	基类指针和公有派生类对象	195
7-2	虚函数	197
7-3	纯虚函数与抽象类	199
7-4	小结	201
	思考与练习题	202

第 8 章 运算符重载	207
8-1 运算符函数.....	207
8-2 运算符重载的应用.....	207
8-2-1 运算符重载为成员函数.....	207
8-2-2 运算符重载为友元函数.....	209
8-2-3 运算符重载的限制.....	210
8-3 典型运算符的重载.....	211
8-3-1 赋值运算符的重载.....	211
8-3-2 单目运算符的重载.....	212
8-3-3 I/O 运算符的重载.....	214
8-4 小结.....	215
思考与练习题.....	216
第 9 章 模板	219
9-1 函数模板.....	219
9-1-1 函数模板的定义.....	220
9-1-2 函数模板的使用.....	220
9-2 类模板.....	222
9-2-1 类模板的定义.....	223
9-2-2 类模板的使用.....	223
9-3 小结.....	226
思考与练习题.....	227
第 10 章 C++流	230
10-1 C++流的概念.....	230
10-2 数据输出的格式控制.....	231
10-2-1 域宽控制.....	231
10-2-2 填充字符控制.....	232
10-2-3 数制控制.....	233
10-2-4 浮点数输出控制.....	234
10-2-5 对齐方式控制.....	234
10-3 文件流.....	235
10-3-1 文件的打开和关闭.....	236
10-3-2 文件的读写.....	236
10-4 小结.....	241
思考与练习题.....	241
参考文献	245
附录 A ASCII 表	246
附录 B 常用库函数	247
附录 C 2007 年全国计算机等级考试二级 C++语言考试大纲	249
附录 D 全国计算机等级考试二级 C++语言考试样题及参考答案	251

第 0 章 C++语言概述

C++语言是在 C 语言的基础上逐渐发展起来的，目前已成为应用较广的一种优秀的高级程序设计语言。C++语言包括 C 语言的全部特征和优点，保留了对传统的结构化程序设计方法的支持，同时添加了对面向对象程序设计方法的完全支持。

本章主要内容：

- C++的产生、发展及特点
- C++程序的基本框架（结构化程序设计框架、面向对象程序设计框架等）
- C++的词汇（基本符号、保留字、标识符、常量、变量、运算符、标点符号等）
- C++程序的开发过程

0-1 C++的产生与发展

C++语言是在 C 语言的基础上，为支持面向对象程序设计方法而研制的一种编程语言。

C 语言是由贝尔实验室的 Dennis Ritchie 在 BASIC 语言的基础上开发出来的。设计 C 语言的最初目的是用做 UNIX 操作系统的描述语言。因此，C 语言的产生和发展与 UNIX 操作系统有着十分密切的联系。1972 年在一台 DEC PDP-11 计算机上实现了最初的 C 语言。由于 C 语言设计严谨，使得把用 C 语言编写的程序移植到大多数计算机上成为可能。到了 20 世纪 70 年代末，C 语言已经演化为现在所说的“传统 C 语言”。C 语言在各种计算机上的快速推广，致使产生了许多 C 语言版本。这些版本虽然是类似的，但通常是不兼容的。为了明确地定义与机器无关的 C 语言，1989 年美国国家标准协会制定了 C 语言的标准(ANSI C)。至此，C 语言以其独有的特点风靡全世界：

- (1) 语言简洁、紧凑，使用方便、灵活。C 语言只有 32 个关键字，程序书写形式自由。
- (2) 丰富的运算符和数据类型。
- (3) 可以直接访问内存地址，能进行位操作，使其能够胜任开发操作系统的工作。
- (4) 生成的目标代码质量高，程序运行效率高。
- (5) 代码的可移植性好。

然而，在 C 语言盛行的同时，也暴露了其局限性：

- (1) 数据类型检查机制相对较弱，这使得程序中的一些错误不能在编译阶段被及时发现。
- (2) C 语言本身几乎没有支持代码重用的语言结构，因此，一个精心设计的程序很难为其他程序所用。
- (3) 当程序的规模达到一定程度时，程序员很难控制程序的复杂性，不适合开发大型的软件。

为了满足管理程序的复杂性的需要，1980 年贝尔实验室的 Bjarne Stroustrup 开始对 C 语言进行改进和扩充。最初的成果称为“带类的 C”，1983 年正式命名为 C++，在经历了 3 次修订后，于 1994 年制定了 ANSI C++标准的草案。这个草案于 1998 年被国际标准化组

织 (ISO) 批准为国际标准 (ISO/IEC 14882)。C++ 在 20 多年的发展过程中不断完善, 至今已成为一门充满活力的程序设计语言。

0-2 C++的特点

C++语言的特点主要表现在两个方面: 一方面是对 C 语言的全面兼容, 另一方面是对面向对象程序设计方法的支持。

C++语言包括了 C 语言的全部特征和优点, 它不仅保持了 C 语言的简洁、高效、接近汇编语言等特点, 而且对 C 语言的数据类型系统进行了改革和扩充, 增强了其数据类型检查机制, 使编译系统能够检查出更多的数据类型错误。

C++语言最突出的特点是对面向对象程序设计方法的完全支持。虽然与 C 语言的兼容使得 C++语言具有结构化程序设计和面向对象程序设计的双重特点, 但是它在概念上是与 C 语言完全不同的语言, 应该注意按照面向对象的思想去编写程序。

0-3 C++程序的基本框架

C++程序的基本框架包括结构化程序设计框架和面向对象程序设计框架。

0-3-1 最简单的 C++程序

下面来看一个最简单的 C++程序。

【例 0-1】 一个最简单的 C++程序。

```
//Li0_1.cpp
#include<iostream>           //包含头文件
using namespace std;        //声明命名空间
int main()
{
    cout<<"A simple C++ program."<<endl; //输出字符串
    return 0;
}
```

程序的运行结果为:

```
A simple C++ program.
```

这个程序虽然很短, 但是却包含了 C++程序的几个基本组成部分。

1. 注释

注释是程序员对程序进行的注解和说明, 目的是提高程序的可读性。编译系统在对程序进行编译的时候忽略注释。在 C++中有两种注释方法:

- (1) 行注释 "//": 注释的内容从 "//" 开始到本行末尾结束。
- (2) 块注释 "/*" 和 "*/": 注释的内容从 "/*" 开始到 "*/" 结束。

2. 编译预处理

C++中所有以 "#" 开头的代码, 都是编译预处理。如程序中第 1 行代码:

```
#include<iostream>
```

其作用是将头文件 `iostream` 中的内容加到程序中。`iostream` 文件中设置了 C++的 I/O

相关环境，并定义了输入/输出流对象 `cin` 和 `cout`。

3. 命名空间

程序中的第 2 行代码：

```
using namespace std;
```

意思是使用标准命名空间 `std` 中的函数。`using namespace` 是针对命名空间的指令。`iostream` 是一个标准函数库，`cout` 是标准函数库提供的一个对象；标准函数库在 `namespace` 说明书中被指定为 `std`。因此，如果想要在代码中使用标准函数库，就必须加入此行代码。

4. 程序主体

每个 C++ 程序有且仅有一个 `main` 函数。`main` 函数（也称主函数）是整个程序运行时的入口。无论主函数处在程序的什么位置，程序都要从它开始执行。`main` 函数前面的 `int` 表示它的返回值类型为整型，意味着程序运行结束后要向操作系统返回一个整型值。

所有的函数体都由一对大括号“{}”括起来，其内容由若干条语句构成，函数体的内容决定了该函数的功能。如程序中的第 5 行代码：

```
cout<<"A simple C++ program."<<endl;
```

其作用是将字符串“A simple C++ program.”输出到屏幕上。其中，`cout` 是标准输出流对象；“<<”是插入运算符，可以连续使用；`endl` 的作用是回车换行。

另外，函数体中的每一条语句后面都要有“;”，表示一条 C++ 语句的结束。

0-3-2 结构化程序设计框架

程序设计问题是从科学计算开始的，科学计算问题一般所描述的数据大多不是很复杂，因此，可以依赖于编程语言的内置数据类型以及它们之间的简单组合来解决。程序设计的主要问题就是制定出问题的解决方法和步骤（即算法），当一个程序较大时，可按具体情况将其分解为多个相对独立的模块，每个模块完成某一特定功能，各模块的功能往往通过函数来实现。这种方法体现了结构化程序设计“自顶向下，逐步求精”的指导思想。

结构化程序设计框架如图 0-1 所示。

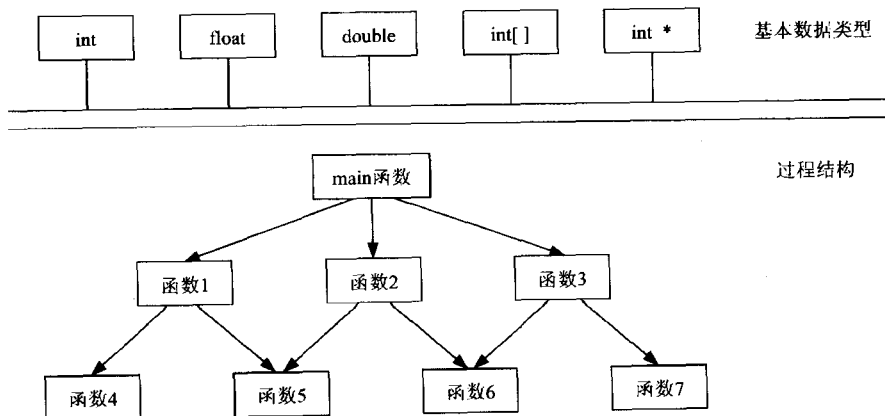


图 0-1 C++ 的结构化程序设计框架

在结构化程序设计框架中，使用的是语言中已有的基本数据类型，如 C++ 语言的基本数据类型（int、float、double 等）以及相应的构造数据类型（数组、指针等）。函数是程序的基本组成单位，即一个 C++ 程序是由一个或若干个函数构成的，但其中必须有且仅有一个 main 函数存在。在 main 函数中可以调用其他函数，并且其他函数之间可以相互调用，同一个函数可以被一个或多个函数调用任意多次。

如图 0-1 所示，可以形式化地将结构化程序设计的框架描述为：

```

<函数 1 的说明>
<函数 2 的说明>
...
<函数 n 的说明>

main 函数
{
    <函数 1 的调用>
    <函数 2 的调用>
    ...
    <函数 n 的调用>
}

<函数 1 的定义>
<函数 2 的定义>
...
<函数 n 的定义>

```

其中，在函数 1~n 的定义中可以包含相应的其他函数（main 函数除外）的调用。

0-3-3 面向对象程序设计框架

面向对象的程序设计方法强调直接以问题域（现实世界）中的事物为中心来思考和认识问题，并按照这些事物的本质特征把它们抽象为对象，以作为构成软件系统的基础，这样，在现实世界中有哪些值得注意的事物，在程序中就有哪些对象与之对应。由于程序与现实世界间具有极强的对应关系，因此，程序可以用对象的概念很自然地进行思考，从而大大降低了软件开发的难度。

1. 面向对象程序设计的有关概念

在面向对象的程序设计中，包括对象、类、继承、封装、消息传递和多态性等基本概念。

(1) 对象 (Object)

对客观事物的数值化描述就是所谓的对象。每个对象都具有属性 (Attribute) 和方法 (Method) 这两方面的特征。对象的属性描述了对应的状态和特征；对象的方法说明了对应的行为和功能，并且对象的属性值只应由这个对象的方法来读取和修改，两者结合在一起就构成了对象的完整描述。

例如：在一个学生学籍管理系统中，可以把每一位同学看成是一个对象，其学号、姓名、班级和已修学分等都是对象的属性，同时每一位同学都有修改学分的功能（即方法），如某位同学新学期选修了一门课程，则可以通过修改学分的方法使学分增加。