

从零开始学

C++ 程序设计

强调理论与实践相结合，按C++的功能由浅入深地
介绍程序设计语言的精髓

立即拥有面向对象
程序设计能力的 **16** 堂课

吴惠茹 等编著

大量范例程序，运行准确无误
上机实训测验，强化编程能力
精心设计习题，验收教学成果





从零开始学

C++ 程序设计

吴惠茹 等编著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

从零开始学C++ 程序设计 / 吴惠茹等编著. —北京: 机械工业出版社, 2017.4

ISBN 978-7-111-56456-0

I. ①从… II. ①吴… III. ①C语言—程序设计IV. ①TP312.8

中国版本图书馆CIP数据核字 (2017) 第065385号

本书注重理论与实践相结合, 按C++的功能由浅入深地详细介绍C++程序设计语言的精髓。

本书分16章说明C++语言相关的语法与应用, 包括变量与数据类型、表达式与运算符、流程控制结构、数组与字符串、指针与地址、函数、预处理指令与宏、自定义数据类型、构造函数与析构函数、类、运算符重载、继承与多态、数据流和文件、异常处理以及C++模板的程序设计方式等内容。

本书的编写以教学为背景, 除了在各章正文的讲解中穿插了大量范例程序的分析外, 在各章的后面还辅以课后习题与解答, 并提供了丰富的上机实训测试题。本书既适合作为大专院校计算机及相关专业的教材, 也适合作为程序设计初学者的自学教材, 还可供有一定编程经验、想快速掌握C++语言的程序员学习参考。

从零开始学 C++ 程序设计

出版发行: 机械工业出版社 (北京市西城区百万庄大街22号 邮政编码: 100037)

责任编辑: 夏非彼 迟振春

印刷: 中国电影出版社印刷厂

版次: 2017年5月第1版第1次印刷

开本: 188mm×260mm 1/16

印张: 26.5

书号: ISBN 978-7-111-56456-0

定价: 69.00元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光/邹晓东

前言

C++是一种编译式语言，源于贝尔实验室，是其原创者 Bjarne Stroustrup 以 C 语言作为基本架构，并引入面向对象的概念开发出来的。严格说来，C++语言融合了传统的面向过程的语言、面向对象的程序设计以及 C++模板三种不同程序设计的方式，使它成为最受重视且非常普及的程序设计语言。从不易出错的角度来说，C++比 C 语言更简单易学，因为它改进了 C 语言中一些容易混淆出错的部分，并且提供了更实用与完整的面向对象的设计功能。

本书内容

本书强调理论与实践相结合，按照 C++的功能循序渐进、由浅入深地详细介绍 C++程序设计语言的精髓。另外，各章除列举和解说丰富的范例程序外，还提供了上机实习课程（上机实训测验），可以让学生强化程序编写与调试能力；精心设计的课后习题用于检验学习成效，相信本书可以作为 C++程序设计的优选教材之一。另外，书中所有的 C++程序都是以免费的 Dev C++集成开发环境来编译与运行的。全书可分成四个重点部分。

（1）基础语法：第 1~4 章

先简单说明如何进行 C++程序编写、编译、运行与调试，接着导入变量、常数以及各种数据类型的介绍，最后介绍各种运算符及流程控制指令。

（2）高级语法：第 5~10 章

先介绍数组与字符串的声明与综合运用，再讲解指针与地址的概念与实践，并示范如何自定义函数、参数传递与函数高级应用，最后介绍预处理指令、宏以及自定义数据类型。

（3）面向对象：第 11~14 章

这是本书的精华部分，先以生活化的案例切入面向对象程序设计的基础和常识，再完整呈现面向对象程序设计的精华。

（4）其他议题：第 15~16 章以及附录 A、B、C

介绍数据流和文件的入门概念，并探讨各种文件类型的操作技巧与管理。另外，这一部分还介绍了异常处理的错误控制机制以及 C++模板的程序设计方式。

下载资源

本书的范例程序包含两类：一类是各章节正文讲解使用的范例程序，另一类是为各章“上机实训测验”提供的参考范例程序。读者可以从下面这个网址免费下载所有范例程序的源代码：

<http://pan.baidu.com/s/1qYCpzze>

在下载过程中，注意区分英文字母大小写以及数字和英文。如果下载有问题，请发送电子邮件至 booksaga@126.com，邮件主题设置为“求从零开始学 C++程序素材”。

本书主要由吴惠茹编著，卞诚君、王叶、刘雪连、吉媛媛、闫秀华、关静、孟宗斌、魏忠波、王翔、郭丹阳等人也参与了本书的编写与校对工作。虽然本书校稿过程力求无误，但是难免有疏漏之处，还望各位不吝赐教！

吴惠茹

2017年2月

目 录

前言

第 1 章 初识 C++	1
1-1 C++的由来.....	1
1-1-1 C 与 C++的关联	2
1-1-2 认识面向对象设计	2
1-2 我的第一个 C++程序.....	3
1-2-1 编写程序	4
1-2-2 保存文件	5
1-2-3 编译程序	5
1-2-4 运行程序	6
1-2-5 程序代码解析	6
1-3 程序架构简介	6
1-3-1 头文件区	7
1-3-2 程序注释	8
1-3-3 主程序区——main()函数	9
1-3-4 命名空间	10
1-3-5 输入/输出功能简介	10
1-3-6 程序语句的格式	11
1-3-7 标识符与保留字	12
1-4 上机实训测验	13
1-5 课后练习	13
第 2 章 变量与数据类型	18
2-1 变量简介	19
2-2 常数	21
2-3 基本数据类型	22
2-3-1 整数	22
2-3-2 浮点数	24

2-3-3	字符	27
2-3-4	转义字符	28
2-3-5	布尔数据类型	30
2-4	上机实训测验	31
2-5	课后练习	32
第3章	表达式与运算符	35
3-1	表达式的表示法	35
3-2	认识运算符	36
3-2-1	运算符优先级	36
3-2-2	赋值运算符	37
3-2-3	算术运算符	38
3-2-4	关系运算符	39
3-2-5	逻辑运算符	41
3-2-6	位运算符	42
3-2-7	递增与递减运算符	47
3-2-8	复合赋值运算符	49
3-3	认识数据类型转换	50
3-3-1	自动类型转换	50
3-3-2	强制类型转换	52
3-3-3	转型运算符简介	53
3-4	上机实训测验	55
3-5	课后练习	56
第4章	流程控制结构	60
4-1	顺序结构	60
4-2	选择结构	61
4-2-1	if 条件语句	61
4-2-2	if-else 条件语句	62
4-2-3	if else if 条件语句	64
4-2-4	条件运算符	66
4-2-5	switch 条件语句	68
4-3	循环结构	70
4-3-1	for 循环语句	71
4-3-2	for 循环的控制子句	72
4-3-3	嵌套 for 循环	73
4-3-4	while 循环语句	74

4-3-5 do-while 循环语句.....	76
4-4 循环控制指令.....	78
4-4-1 break 指令.....	78
4-4-2 continue 指令.....	80
4-4-3 goto 指令.....	81
4-5 上机实训测验.....	82
4-6 课后练习.....	85
第 5 章 数组与字符串.....	89
5-1 数组简介.....	89
5-1-1 数组表示法.....	89
5-1-2 数组与变量.....	90
5-1-3 一维数组.....	91
5-1-4 二维数组.....	93
5-1-5 多维数组.....	95
5-2 字符串简介.....	97
5-2-1 字符串声明.....	97
5-2-2 字符串数组.....	100
5-3 String 类.....	102
5-3-1 声明 String 类.....	103
5-3-2 字符串运算符功能.....	105
5-3-3 String 类成员函数.....	106
5-4 上机实训测验.....	108
5-5 课后练习.....	110
第 6 章 指针与地址.....	113
6-1 认识地址.....	113
6-1-1 指针的作用.....	113
6-1-2 变量地址的存取.....	114
6-1-3 指针变量的声明.....	115
6-1-4 指针运算.....	118
6-1-5 多重指针.....	121
6-2 指针与数组.....	122
6-2-1 指针与一维数组.....	124
6-2-2 指针与多维数组.....	125
6-2-3 指针与字符串.....	128
6-2-4 指针数组简介.....	130

6-3	动态分配功能.....	134
6-3-1	动态分配与静态分配.....	134
6-3-2	动态分配变量.....	135
6-3-3	动态分配数组.....	136
6-4	引用类型.....	138
6-4-1	引用类型声明.....	138
6-4-2	指针引用简介.....	140
6-5	上机实训测验.....	141
6-6	课后练习.....	143
第7章	函数入门.....	149
7-1	函数功能简介.....	149
7-1-1	自定义函数.....	149
7-1-2	函数声明模式.....	151
7-1-3	函数调用语法.....	154
7-2	认识参数传递.....	155
7-2-1	传值调用模式.....	155
7-2-2	传址调用模式.....	157
7-2-3	传引用调用模式.....	159
7-2-4	参数默认值.....	160
7-2-5	数组参数.....	162
7-2-6	指针返回值.....	163
7-3	上机实训测验.....	165
7-4	课后练习.....	166
第8章	函数的高级应用.....	170
8-1	函数指针简介.....	170
8-1-1	声明函数指针.....	170
8-1-2	参数型函数指针.....	172
8-1-3	函数指针数组.....	175
8-2	命令行参数.....	176
8-3	变量的种类.....	179
8-3-1	全局变量.....	179
8-3-2	局部变量.....	179
8-3-3	区块变量.....	179
8-4	变量等级说明.....	180
8-4-1	自动变量.....	180

8-4-2	静态局部变量.....	180
8-4-3	外部变量.....	182
8-4-4	静态外部变量.....	183
8-4-5	寄存器变量.....	185
8-5	特殊函数功能.....	185
8-5-1	递归函数.....	185
8-5-2	内联函数.....	187
8-5-3	函数重载.....	189
8-6	上机实训测验.....	191
8-7	课后练习.....	192
第 9 章	预处理指令与宏.....	196
9-1	预处理指令.....	196
9-2	#define 指令.....	198
9-2-1	定义基本指令.....	198
9-2-2	定义函数.....	200
9-2-3	宏函数简介.....	201
9-2-4	标准预处理宏.....	203
9-3	条件编译指令.....	204
9-3-1	#if、#endif、#else、#elif 指令.....	204
9-3-2	#ifdef、#ifndef 指令.....	206
9-4	上机实训测验.....	207
9-5	课后练习.....	207
第 10 章	认识自定义数据类型.....	211
10-1	结构.....	211
10-1-1	结构的声明方式.....	211
10-1-2	结构的存取.....	212
10-1-3	结构指针.....	213
10-2	结构与数组.....	216
10-2-1	结构数组.....	216
10-2-2	结构的数组成员.....	218
10-2-3	结构指针数组.....	219
10-3	嵌套结构.....	221
10-4	函数与结构.....	224
10-4-1	结构传值调用.....	224
10-4-2	结构传址调用.....	226

10-4-3	结构传引用调用	226
10-5	其他自定义数据类型	227
10-5-1	类型定义指令	227
10-5-2	枚举类型	228
10-5-3	联合类型	231
10-6	上机实训测验	233
10-7	课后练习	235
第 11 章	面向对象程序设计入门	240
11-1	类的基本概念	240
11-1-1	类对象的声明	241
11-1-2	存取权限关键字	242
11-1-3	创建类对象	243
11-1-4	范围解析运算符	245
11-2	构造函数与析构函数	246
11-2-1	构造函数	246
11-2-2	范围解析运算符	248
11-2-3	构造函数重载功能	250
11-2-4	析构函数	251
11-2-5	创建指针对象	253
11-3	上机实训测验	255
11-4	课后练习	256
第 12 章	类的高级应用	259
12-1	对象数组与友元关系	259
12-1-1	对象数组	259
12-1-2	友元函数	262
12-1-3	友元类	265
12-2	this 指针与静态数据成员	267
12-2-1	this 指针	267
12-2-2	静态数据成员	269
12-2-3	嵌套类	271
12-3	函数与对象传递	273
12-3-1	对象传值调用	273
12-3-2	对象传址调用	273
12-3-3	对象传引用调用	274
12-4	上机实训测验	274

12-5 课后练习.....	275
第 13 章 运算符重载	277
13-1 运算符重载简介.....	277
13-1-1 重载的定义与规则.....	277
13-1-2 一元运算符重载.....	280
13-1-3 二元运算符重载.....	282
13-2 特殊运算符重载介绍.....	284
13-2-1 “>>”与“<<”运算符重载.....	284
13-2-2 “=”运算符重载.....	286
13-2-3 “==”运算符重载.....	288
13-2-4 “++”与“--”运算符重载.....	291
13-3 类型转换运算符重载.....	293
13-3-1 类类型转换为基本数据类型.....	294
13-3-2 基本数据类型转换为类类型.....	296
13-3-3 类类型转换成其他类类型.....	298
13-4 上机实训测验.....	301
13-5 课后练习.....	301
第 14 章 继承与多态	304
14-1 继承关系.....	304
14-1-1 基类与派生类.....	304
14-1-2 单一继承.....	305
14-1-3 public 关键字.....	305
14-1-4 protected 关键字.....	307
14-1-5 private 关键字.....	309
14-1-6 多重继承.....	312
14-2 派生类的构造函数与析构函数.....	315
14-2-1 单一继承构造函数与析构函数.....	315
14-2-2 多重继承构造函数与析构函数.....	317
14-3 多态与虚拟函数.....	319
14-3-1 静态绑定与动态绑定.....	319
14-3-2 声明虚拟函数.....	322
14-3-3 纯虚拟函数.....	324
14-3-4 抽象基类.....	324
14-3-5 虚拟基类.....	326
14-4 上机实训测验.....	328

14-5	课后练习.....	329
第 15 章	文件入门与处理	331
15-1	数据流的概念.....	331
15-1-1	数据流类.....	332
15-1-2	I/O 处理类.....	332
15-2	文件简介.....	333
15-2-1	文件分类.....	333
15-2-2	按序式与随机式文件.....	334
15-3	文件的输入/输出管理.....	334
15-3-1	文件的打开.....	334
15-3-2	文件的关闭.....	335
15-4	文本文件操作技巧.....	337
15-4-1	文本文件的写入.....	337
15-4-2	文本文件的读取函数.....	338
15-5	二进制文件操作技巧.....	340
15-5-1	二进制文件的写入.....	340
15-5-2	二进制文件的读取技巧.....	341
15-5-3	随机存取模式简介.....	343
15-6	上机实训测验.....	345
15-7	课后练习.....	346
第 16 章	异常处理与模板	349
16-1	异常处理功能的基本认识.....	349
16-1-1	简单的异常处理结构.....	349
16-1-2	try 指令.....	350
16-1-3	catch 区块重载.....	352
16-1-4	嵌套 try...catch 区块.....	353
16-1-5	一次捕获所有异常.....	355
16-1-6	重新抛出异常.....	356
16-2	认识模板功能.....	358
16-2-1	模板分类.....	359
16-2-2	函数模板.....	359
16-2-3	声明函数模板.....	360
16-2-4	非类型参数的模板函数.....	362
16-3	类模板.....	363
16-3-1	声明类模板.....	364

16-3-2 堆栈类模板	366
16-3-3 非类型参数类模板	368
16-4 上机实训测验	370
16-5 课后练习	371
附录 A C++的常用函数库	375
A-1 字符处理函数	375
A-2 字符串处理函数	377
A-3 类型转换函数	380
A-4 时间和日期函数	381
A-5 数学函数	382
A-6 随机数函数	384
附录 B 格式化输入/输出数据	387
B-1 插入运算符<<	387
B-2 常用输出函数	395
B-3 提取运算符>>	396
B-4 常用输入函数	397
B-5 错误状态位	399
附录 C C++编译程序的介绍与安装	402
C-1 C/C++编译程序简介	402
C-1-1 Visual Studio	402
C-1-2 Dev C++	403
C-1-3 GCC	404
C-2 Dev C++的安装与介绍	404
C-2-1 下载 Dev C++	404
C-2-2 安装 Dev C++	405
C-2-3 程序项目的建立	407

第 1 章

◀ 初识C++ ▶

在 1946 年，美国宾州大学教授埃克特与莫克利合作完成了人类第一台真空电子管计算机 ENIAC，接着冯·诺依曼教授首先提出了存储程序方式与二进制的概念，认为数据与程序可以存储在计算机主存储器内，这开启了程序设计语言与程序设计蓬勃发展的序幕。

“程序设计语言”就是一种人类用来和计算机沟通的语言，也是用来指挥计算机运算或工作的指令集合，可以将程序员的思考逻辑和语言转换成计算机能够了解的语言。到了 1954 年，德州仪器（Texas Instruments）公司成功地研制出以硅为半导体材料制造而成的商业用晶体管，从而让现代计算机制造技术真正地起飞了。一些高级语言（High Level Language）也在这时发展出来，取代了以往所使用的机器语言。

高级语言是相当接近于人类使用的自然语言的程序设计语言，虽然执行速度较慢，但是语言本身易学易用，因此被广泛地应用在商业、科学、教学、军事等相关的软件开发上。例如，Fortran、COBOL、Basic、PASCAL、C 或是 C++ 都是高级语言的一员，其中 C/C++ 更是其中的翘楚，在程序设计领域有着非凡的贡献。

1-1 C++的由来

1972 年贝尔实验室的 Dennis Ritchie 以 B 语言为基础，并持续不断地改进和完善它。除了保留 BCLP 和 B 语言中的许多概念外，还加入了数据类型的概念及其他功能，并且将其发表为“C 语言”。

事实上，在许多平台的主机上都有 C 语言的编译程序，例如 MS-DOS、Windows 系列操作系统、UNIX/Linux，甚至 Apple 公司的 Mac 系列系统等都有 C 语言的编译程序。因此，程序员能够轻松地跨越许多平台来开发程序。C 语言仅需克服平台差异即可在不同的机器上编译与运行。

C++ 同样也源自于贝尔实验室，当初其原创者 Bjarne Stroustrup 以 C 作为基本的架构，再引入面向对象的概念，从而形成了最初的 C++ 语言。因此，C++ 可以说是包含了整个 C 语言，也就是说几乎所有的 C 语言程序，只要进行微幅的修改，甚至于完全不需要修改，便可正确地在 C++ 环境中执行。所以 C 语言程序在编译程序上直接将扩展名 c 改为 cpp，即可编译成 C++ 语言程序。

1-1-1 C 与 C++的关联

C++属于一种编译式语言，也就是使用编译程序（compiler，或称为编译器）来将源代码程序转换为机器可读取的可执行文件或目标程序，不过编译程序必须先把源程序读入主存储器后才可以开始编译。而翻译后的目标程序可直接对应成机器代码，故而可在计算机上直接运行，而不需要每次运行都重新进行翻译，因此运行速度自然较快。但是，源程序每修改一次，就必须重新经过编译程序的编译，才能保持其运行文件为最新的版本。

或许大家心中会有疑问，是否有必要在学习 C++前先学会 C 语言呢？事实上，直接学习 C++并不需要重复学习 C 语言，因为 C++完全包含了 C 语言。有人甚至会觉得 C++比 C 更为简单易学，因为它改进了 C 语言中一些容易混淆出错的部分，并且提供了更实用与完整的面向对象设计功能。图 1-1 所示即为 C++和 C 之间的关联。

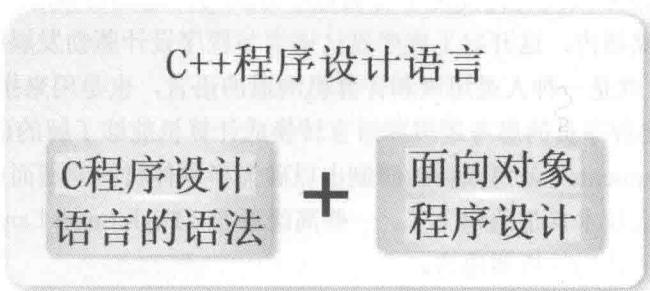


图 1-1 C++和 C 之间的关联

面向对象设计主要是让我们在进行程序设计时能以一种接近生活化的思考方式来编写出可读性高的程序，并且让所设计的程序代码也较容易扩充、修改和维护。严格来说，C++并不是一套绝对的面向对象语言，而 C++中所增加的面向对象功能更适时地解决了大型软件开发时所面临的困境，并能充分加强程序代码的扩展性与重用性。

1-1-2 认识面向对象设计

C++中最让人津津乐道的创新功能就是“面向对象程序设计”，这也是程序设计领域的一大创新。在传统程序设计的方法中，主要是以“结构化程序设计”为主，它的核心精神就是“自上而下设计”与“模块化设计”，也就是将整个程序需求自上而下、从大到小逐步分解成较小的单元，或称为“模块”（Module）。这样使得程序员可针对各个模块分别开发，减轻了设计者的负担，提高了程序的可读性，对于日后的维护也容易许多。

每一个模块会各自完成特定的功能，主程序则组合每个模块后完成最后要求的功能。不过，一旦主程序要求功能变动，许多模块内的数据与程序代码就可能需要同步变动，而这也是结构化程序设计无法有效使用程序代码的主因。

“面向对象程序设计”（Object-Oriented Programming, OOP）是当今主流的程序设计概念，主要让程序员在设计程序时能以一种生活化、可读性更高的设计概念来进行程序的开发，从而

使所开发出来的程序也较容易扩充、修改和维护，以弥补“结构化程序设计”的不足，如 C++、Java 等语言。首先我们要了解面向对象程序设计具备的以下三种特性，如图 1-2 所示。



图 1-2 面向对象程序设计的三种特性

所谓封装 (Encapsulation)，是使用“类”来实现“抽象数据类型”(ADT)。所谓“抽象”，是让用户只能接触到这些方法 (函数)，而无法直接使用数据，即符合信息或数据隐藏的要求，这就是“封装”的主要作用。

继承 (Inheritance) 类似现实生活中的遗传，允许我们去定义一个新的类来继承现有的类 (Class)，进而使用或修改继承而来的方法 (Method)，并可在子类中加入新的数据成员与函数成员。

多态 (Polymorphism) 也是面向对象设计的重要特性，可让软件在开发和维护时达到充分的扩展性。简单地说，多态最直接的定义就是让具有继承关系的不同类对象可以调用相同名称的成员函数，并产生不同的响应结果。

1-2 我的第一个C++程序

其实学习程序设计语言和学游泳一样，没有别的捷径，下水直接体验才是最快的方法。从笔者多年从事程序设计语言的教学经验中得出这样的结论：在教初学者学习新的程序设计语言时就是不要有太多废话，尽快让他们从无到有，实际编写和运行一个程序最为重要，许多编程高手都是程序写多了，对所使用的程序设计语言的领悟才越来越深。

在早期，要编写 C/C++ 语言的程序，首先必须找一种文本编辑器来进行程序的编辑，例如 Windows 系统下的“记事本”编辑器，或是 Linux 系统下的 vi 编辑器，接着选一种 C/C++ 语言的编译程序 (如 Turbo C/C++、mingW、gcc 等) 来编译，然后运行即可。

不过现在不用这么麻烦了，只要找个可将程序的编辑、编译、运行与调试等功能集成于同一个操作环境下的“集成开发环境”(Integrated Development Environment, IDE) 即可。

由于 C/C++ 的应用市场很大，市面上较为知名的 IDE 就有 Dev C++、Visual C++ Express、C++ Builder、Visual C++ 和 GCC 等。现行的几种 C/C++ 集成开发环境 (IDE) 各自都有一些极少数自定义的语法与特殊功能。然而，对于初学者而言，只要从基本的内容着手，将重点放在语法、逻辑等方面就可以了。目前市面上几乎没有单纯的 C 语言编译程序，通常都是与 C++ 编译程序兼容，称为 C/C++ 编译程序或编译器。注意，本书中所有的 C++ 程序文件都是以免费的 Dev C++ 集成开发环境来完成编译和调试的，并可以正确无误地运行。

现在请大家按照附录 C 的说明，在你的计算机中安装好 Dev C++，再开始运行 Dev C++ 集