大学计算机规划教材

C++ 程序设计基础（第5版）（上）

◆ 周霭如　林伟健　编著

中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

大学计算机规划教材

C++程序设计基础

（第5版）（上）

周霭如　林伟健　编著

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

内 容 简 介

本书的例程以 VC 2010 为运行环境，全书分为上、下两册：上册介绍 C++程序设计的基础知识，下册介绍 VC++在.Net 环境下的 Windows 应用程序设计。

本书为上册，共分 12 章：第 1 章基本数据与表达式，第 2 章程序控制结构，第 3 章函数，第 4 章数组，第 5 章集合与结构，第 6 章类与对象，第 7 章运算符重载，第 8 章继承，第 9 章虚函数与多态性，第 10 章模板，第 11 章输入流/输出流，第 12 章异常处理。

本书提供配套的电子课件和习题解答，请登录华信教育资源网（www.hxedu.com.cn）注册后免费下载。电子课件由近 3000 张 PPT 幻灯片组成，以图形语言为设计理念，充分表达程序设计课程的教学特点。

本书可以作为高等学校计算机类、信息类、电类专业本科生高级语言程序设计课程教材，也可以作为教师、学生和 C++语言爱好者的参考书。

图书在版编目（CIP）数据

C++程序设计基础. 上 / 周霭如，林伟健编著. —5 版. —北京：电子工业出版社，2016.5
大学计算机规划教材
ISBN 978-7-121-28595-0

Ⅰ. ①C… Ⅱ. ①周… ②林… Ⅲ. ①C 语言-程序设计-高等学校-教材 Ⅳ. ①TP312

中国版本图书馆 CIP 数据核字（2016）第 078890 号

策划编辑：冉 哲
责任编辑：冉 哲
印 刷：三河市华成印务有限公司
装 订：三河市华成印务有限公司
出版发行：电子工业出版社
北京市海淀区万寿路 173 信箱 邮编 100036
开 本：787×1 092 1/16 印张：29 字数：835 千字
版 次：2003 年 8 月第 1 版
2016 年 5 月第 5 版
印 次：2016 年 5 月第 1 次印刷
定 价：59.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。
本书咨询联系方式：ran@phei.com.cn。

前　言

C++语言是优秀的计算机程序设计语言，它的功能相当强大。我们编写这本书的目的是，为没有任何程序设计基础的理工科大学生提供一本适用教材，使他们掌握从理论到实践都要求很高的C++语言。

一门课程的设置应该放在整个教学培养计划中统筹考虑。我们的教学目标不是马上培养一个会使用某种语言（例如C++语言）的程序员，而是强调对程序设计语言的理解和应用，“计算机语言”的角色是第一位的。所以，在教材编写和组织教学的过程中，我们力图通过对基本语法现象的剖析，由浅入深地让学生理解、掌握语言规则的原理，懂得用计算机指令的模式去分析和求解问题，并在机器上实现简单的程序。至于深入的算法及大程序的组织讨论，将由相关的后续课程（例如，数据结构、算法分析、计算方法、软件工程等）完成。因此，对高级程序设计语言规则的理解和应用是本教材写作的立足点。

我们根据多年从事计算机程序设计教学的经验，按照学生学习的认知规律，精心构造整本教材的体系和叙述方式。原则是：循序渐进、难点分散、通俗而不肤浅。本教材以语法范畴和程序组织为脉络，清晰渐进。从字、词、数据、表达式、语句，到函数、类，是语法范畴构成的基本脉络；在程序功能方面，则以组织数据和组织程序为另外一条基本脉络，并以渐进的、粒度扩大的方式逐步导入分析。

例如，数据的组织方式：基本数据类型—数组—结构—链表，体现如何利用基本数据类型根据需要组织数据；程序的组织方式：语句—函数—类，体现结构化思想和面向对象思想对程序不同的组织方式。

指针是C++语言的重要概念，是操作对象的有力工具。本书没有一般C语言、C++语言教材中专门的“指针”一章。我们从最简单的变量开始，建立对象的名和地址的概念，用对象的不同访问方式贯穿于各章节。从结构化程序设计到面向对象程序设计，采取了比较平滑的过渡。首先，在一开始介绍基本数据类型、程序流程控制、函数等结构化程序设计的基本知识时，就非正式地使用“对象”这个术语（从计算机程序的角度，任何占有存储空间的实体都是对象）；继而，掌握结构到类的演变，给出对象的准确定义；进一步，展开介绍面向对象程序的几个基本特性，即封装、继承、多态和类属在C++语言中的实现方法。同时，我们在本书的阐述中体现一个思想：没有一种对所有问题都是最好的程序设计方法，对特定问题，选择合适的解决方案是程序员必备的素质。

本书之所以取名为《C++程序设计基础》，原因有二：第一，它不是一本C++语言手册，不可能包罗所有语法规则和特定版本提供的各种功能；第二，它没有涉及复杂的算法和工程化的面向对象分析设计方法。这两个问题与教材的定位相关。对第一个原因，我们认为学生在掌握了程序设计的基本概念和基本方法之后，可以通过语言平台（例如Visual C++）或者其他资料学习，拓展对语言功能的了解。我们在有关章节中，也做了类似的引导，例如，STL标准类库的介绍，这些内容提供给教师选择或学生自学。至于第二个原因，那些是计算机专业后续课程的教学内容。本书介绍的程序设计方法和使用到的算法都立足于基本概念和方法，所以，例程通常是简单和小规模的。

本书分别在2003年、2006年、2009年和2012年出版了第1~4版，目前为第5版。上册介绍程序设计的基础知识，在第5章中增加了一节讨论结构参数；第11章增加了文件应用的例程。

本书各章均配有练习题，其中：同步练习按节编号（例如，同步练习 1.1 对应 1.1 节内容），最后给出综合练习，以方便学生自学。下册介绍 VC++在.NET 环境下的 Windows 应用程序设计，增加了一章用于强化综合训练的设计案例。

上册共分 12 章，主要内容包括：基本数据与表达式、程序控制结构、函数、数组、集合与结构、类与对象、运算符重载、继承、虚函数与多态性、模板、输入/输出流、异常处理。

本书提供配套的电子课件和习题解答，请登录华信教育资源网（www.hxedu.com.cn）注册后免费下载。电子课件由近 3000 张 PPT 幻灯片组成，以图形语言为设计理念，充分表达程序设计课程的教学特点。

本书可以作为高等学校计算机类、信息类、电类专业本科生高级语言程序设计课程教材，也可以作为教师、学生和 C++语言爱好者的参考书。

本书的编写过程，是作者不断向学生学习，向同行学习，向 C++学习的过程。在此，对所有使用本书的教师、学生，以及热心向我们提出宝贵意见的读者致以诚挚的感谢！希望继续得到读者的支持和帮助。

作　者

课程简介

华南理工大学计算机学院开设的“高级语言程序设计（C++）”课程是 2007 年国家级精品网络课程，2012 年国家级精品资源共享课程。本书是该课程的使用教材。

相关网站链接可以扫描以下二维码获得。

爱课程：

http://www.icourses.cn/coursestatic/course_5847.html

华南理工大学网络学院：

http://cs.scutde.net/courses/course_10/index.html

目　录

第 1 章　基本数据与表达式

程序设计语言是人指挥计算机工作的工具。C++语言功能强大，使用灵活，是目前工程中应用比较广泛的一种高级程序设计语言。本章介绍高级程序设计语言的基本概念、C++语言的基本语法单位及表达式运算。

1.1　概述

C++语言源于 C 语言。C 语言诞生于 20 世纪 70 年代，最初设计的目的是编写操作系统。因为 C 语言规则简单，不但具有高级语言的数据表示、运算功能，还可以直接对内存操作，程序运行效率高。基于以上优点，C 语言很快成为世界流行的程序设计语言。

然而，人们要求计算机解决的问题越来越多，C 语言在处理大问题、复杂问题时表现出来的弱点也越来越明显，例如，缺乏数据类型检查机制，代码重用性差等。

20 世纪 80 年代，美国 AT&T 贝尔实验室对 C 语言进行扩充改版，成为 C++语言。C++语言保持了 C 语言原有的高效、简洁的特点，强化了数据的类型检查和语句的结构性，增加了面向对象程序设计的支持。由于 C++语言的灵活性、良好的继承性和前瞻性，许多软件公司都为 C++语言设计编译系统，提供不同级别的应用类库以及方便实用的开发环境，使 C++语言得到广泛应用。

1.1.1　程序设计与程序设计语言

在人类社会生活中，“语言”是人与人之间用来表达意思、交流思想的工具，是由语音、词汇和语法构成的一定系统。人类的思维、感情相当丰富，所以，人类的语言系统非常复杂。甚至同一个词、同一个句子，在不同的环境下、以不同的语气表达，都可能解释成完全不同的意思。

“程序设计语言”是人指挥计算机工作的工具。它是一种工程语言，由字、词和语法规则构成的指令系统。一种高级程序设计语言往往只有一百多个词汇、若干条规则。

高级语言提供了常用的数据描述和对数据操作的规则描述。这些规则是“脱机”的，程序员只需要专注于问题的求解，不必关心机器内部结构和实现。我们说的“程序设计语言”一般是指高级语言。

计算机对问题的求解方式通常可以用数学模型抽象。随着社会科学的发展，人们要求计算机处理的问题越来越复杂，计算机工作者不断寻求简洁可靠的软件开发方法。从过程化、结构化到近代出现的面向对象程序设计，体现了程序设计理论、方法的不断发展。

用高级语言编写的程序称为“源程序”。计算机不能直接识别源程序，必须翻译成二进制代码才能在机器上运行。一旦编译成功，目标程序就可以反复高速执行。

程序设计就是根据特定的问题，使用某种程序设计语言，设计出计算机执行的指令序列。程序设计是一项创造性的工作，根据任务主要完成以下两方面工作。

1. 数据描述

数据描述是指把被处理的信息描述成计算机可以接受的数据形式，如整数、实数、字符、数组等。

信息可以用人工或自动化装置进行记录、解释和处理。使用计算机进行信息处理时，这些信息必须转换成可以被机器识别的“数据”，如：数字、文字、图形、声音等。不管什么数据，计算机都以二进制数的形式进行存储和加工处理。数据是信息的载体，信息依靠数据来表达。

有一些数据，可以直接用程序设计语言的“数据类型”描述，如：数值、字符。另外一些数据，虽然一般的程序设计语言没有提供直接定义，但许多开发商都会提供相应的处理工具。例如，Visual Studio .NET Framework 类库提供了丰富的多媒体数据处理方法，可以在界面或程序代码中使用或处理图形、声音等数据。

2. 数据的处理

数据处理是指对数据进行输入、输出、整理、计算、存储、维护等一系列活动。数据处理的目的是为了提取所需的数据成分，获得有用的资料。

程序设计语言的规则都是围绕描述数据、操作数据而设计的。在结构化程序设计中，数据的描述和处理是分离的。用面向对象方法，程序对数据和处理进行封装。按照人们习惯的思维模式和软件重用原则，对象还具有继承、多态等特性。每种程序设计方法都有自己的一套理论框架，相应的设计、分析、建模方法，都有各自的优缺点。采用什么方法设计程序，应该依据问题的性质、规模、特点进行选择。世界上没有一种能解决所有问题的最优方法。

学习 C++语言，不仅为了掌握一种实用的计算机软件设计工具，更重要的是，通过该课程学习，掌握计算机程序设计语言的基本语法规则，掌握结构化程序设计和面向对象程序设计的基本方法，为进一步学习和应用打下良好基础。

1.1.2 一个简单的 C++程序

问题：输入圆的半径，求圆的周长和面积。

【例 1-1】 方法一，用结构化方法编程。

数据描述：半径、周长、面积均用浮点型数据表示。

数据处理：

```
输入半径 r;
计算周长 = 2*π*r;
计算面积 = π*r*r;
输出半径，周长，面积。
```

可以编写如下程序：

```
#include<iostream>
using namespace std;
int main()
{   double r, girth, area;                    //说明数据
    const double PI = 3.1415;
    cout << "Please input radius:\n";
    cin >> r;                                 //输入半径
    girth = 2 * PI * r;                       //计算周长
    area = PI * r * r;                        //计算面积
    cout << "radius = " << r << endl;         //输出数据
    cout << "girth = " << girth << endl;
    cout << "area = " << area << endl;
}
```

上述程序运行后，屏幕显示：

```
Please input radius:
```

用户输入：

```
6.23
```

程序继续执行，计算并输出结果：

```
radius = 6.23
girth = 39.1431
area = 121.931
```

若再次运行程序，可以输入不同的半径值，求得不同圆的周长和面积。

这个程序很容易读懂。第 1 行称为预编译指令，说明该程序要使用的外部文件。C++语言标准头文件 iostream 包含了程序常用的输入 cin 和输出 cout 的定义。

第 2 行是使用名空间的声明。using 和 namespace 都是关键字，std 是系统提供的标准命名空间。详细说明见 3.8 节。

C++语言以函数为程序运行的基本单位，函数的一般形式为：

```
类型  函数名  (参数表)
{
        语句序列
}
```

“函数名”是标识符，用于识别和调用函数。用户自定义函数由程序员命名。一个程序可以由多个文件组成，一个文件可以包含多个函数。每个程序必须有一个，而且只有一个主函数，因为主函数是由系统启动的。最简单的程序只由主函数构成。main 是系统规定的主函数名。

“函数名”之前的“类型”表示函数运行返回表达式值的数据类型。C++主函数返回类型一般为 int 或 void。

“函数名”之后一对圆括号相括的是函数参数。如果没有参数，圆括号也不能省略，它是C++函数的标识。函数名、函数返回类型和参数表组成 C++的函数头（或称为函数首部）。

函数头之后以一对花括号相括的“语句序列”构成函数体。C++语句以分号结束，一行可以写多个语句，一个语句可以分多行写。程序按语句序列执行。

花括号也可以出现在语句序列中。这时，花括号相括的语句称为复合语句或语句块。根据语句的功能不同，有说明语句、执行语句、流程控制语句等。

以双斜杠“//”开始的文本为程序注释，放置在行末。以“/*…*/”相括的注释文本可以放置在程序的任何位置。注释内容不是执行代码，用于增加程序的可读性。系统只显示注释内容，不予编译。

有关函数定义和使用，参见第 3 章的相关内容。

【例 1-2】 方法二，用面向对象方法编程。

当我们用对象思维考虑问题时，可以对问题进一步抽象。所有称为“圆”的这种类型的几何图形，最基本的要素是半径。它决定了圆的大小，也是区分具体圆 A、圆 B、圆 C 等的基本数据。一旦定义了具体的有半径值的圆，就有它特定的周长和面积了。

“圆”是一种类型。在面向对象方法中，称为“类类型”或“类”。“圆”类型的基本数据是“半径”。类的数据称为“属性”或“数据成员”。

数据成员有了具体的值之后，就可以计算周长和面积了。这种“计算”由程序代码实现，并且“封装”在类中，称为类的“方法”或“成员函数”。

下面是用这种方法编写的 C++程序：

```
#include<iostream>
using namespace std;
class Circle                    //说明类
{    double radius;             //类的数据成员
  public:
    //类的成员函数
```

```
    void Set_Radius( double r )
    {   radius = r;   }
    double Get_Radius()
    {   return   radius;   }
    double Get_Girth()
    {   return   2 * 3.14 * radius;   }
    double Get_Area()
    {   return   3.14 * radius * radius;   }
};
int main()
{   Circle A, B;                    //说明对象
    A.Set_Radius( 6.23 );
    cout << "A.Radius = " << A.Get_Radius() << endl;
    cout << "A.Girth = " << A.Get_Girth() << endl;
    cout << "A.Area = " << A.Get_Area() << endl;
    B.Set_Radius( 10.5 );
    cout << "B.radius = " << B.Get_Radius() << endl;
    cout << "B.Girth=" << B.Get_Girth() << endl;
    cout << "B.Area = " << B.Get_Area() << endl;
}
```

程序运行结果：

```
A.Radius = 6.23
A.Girth = 39.1244
A.Area = 121.873
B.Radius = 10.5
B.Girth = 65.94
B.Area = 346.185
```

该例程首先说明一个类 Circle。类中数据成员 radius 用于定义半径。成员函数 Set_Radius 用于设置半径的值，Get_Radius 获取半径，Get_Girth 计算并返回圆周长，Get_Area 计算并返回圆面积。

主函数中说明了圆类 Circle 的两个圆：A 和 B。A 和 B 称为 Circle 类的实例或对象。main 函数中由对象调用成员函数输出两个圆的半径、周长和面积。

这个程序比例 1-1 看起来要烦琐一些。但是，以 Circle 类为基础，可以很方便地派生新的类。新的类对原有类的特性不需要重定义，可以自己定义新的数据，例如，指定圆心坐标，填充圆的颜色；派生出球体、圆柱体等新的几何体。每个新类都可以拥有自己的成员函数，实现自己特有的功能，这是结构化程序设计方法所做不到的。面向对象技术提供了软件重用、解决大问题和复杂问题的有效途径。

1.1.3 程序的编译执行

用高级语言编写的程序称为“源程序”。源程序是文本文件，便于阅读修改。C++的.cpp 文件是文本文件，可以用各种字处理工具打开和编辑。计算机不能直接识别源程序，必须翻译成二进制代码才能在机器上运行。翻译方式有两种：一种称为解释方式，另一种称为编译方式。解释方式是指由“解释程序”对源程序逐个语句地一边翻译，一边执行。这种方式执行速度慢，便于观察调试程序。编译方式是指由“编译程序”把源程序全部翻译成二进制代码。编译后的程序称为“目标程序”，可以反复高速运行。每种高级语言都配有解释或编译系统。

C++提供编译执行方式。实现一个 C++语言源程序主要经过以下 3 个步骤。

1．编辑

使用C++语言编辑器或其他文字编辑器录入源程序。若使用C++语言编辑器，则系统自动生成.cpp文件扩展名；若使用其他文字编辑器，则只有以.cpp为扩展名的文件才能被C++语言识别。.cpp文件是文本文件。

2．编译

把一个.cpp文件编译成.exe目标文件，要经过预处理、编译和连接3个步骤：预处理的作用是执行程序编译之前的准备，例如执行包含指令、宏替换命令；然后编译器对程序进行语法检查，如果发现语法错误，则显示错误信息，让程序员修改，直至正确，生成目标代码；最后把目标代码进行连接处理，往往还会加入一些系统提供的库文件代码。

这些步骤在集成开发环境中会自动完成。

3．运行

VC.NET用文件夹管理应用程序。经过编译后，应用程序文件夹中有一个扩展名为.sln的解决方案文件，需要在C++环境下执行。还生成一个debug文件夹，里面有一个.exe文件，可以直接在操作系统（如DOS，Windows）环境下执行。为了便于测试程序，C++提供了强有力的跟踪调试和错误处理功能。源程序和目标程序都能够作为文件永久保存。

编写源程序难免存在一些错误，这些错误可以分成以下4类。

① 编译错误：在编译源程序时发现的语法错误。例如，表达式(a+b*(c-d)缺少了右括号。

② 连接错误：在程序编译之后，进行连接时出现的错误。例如，找不到连接库文件。

③ 运行错误：执行目标程序时发现的错误。例如，执行标准函数sqrt(x)，求x的平方根，而x的值为负数。

④ 逻辑错误：编译和运行时均不能发现的错误。例如，执行表达式2/4，期望值是0.5，但C++作整除运算，结果为0。

一个程序经常要经过反复的调试、验证会才能完善，投入使用。因此，编写的程序应该力求达到以下目标。

① 正确性：这要求程序员熟悉所使用的程序设计语言，避免语法、语义上的错误；设计简单易行的算法达到预期目的；对复杂的问题，则应考虑使用有效的程序设计方法。

② 易读性：一个程序结构清晰易读，才能便于查错，便于修改。一个程序模块耦合度低、接口清晰，才便于代码的重用。

③ 健壮性：当输入或运行出现数据错误时，程序能够做出适当的反应或进行处理，而不会产生莫名其妙的结果。

④ 运行高效率：程序运行时间较短，而且占用的存储空间较小。

为达到以上目标，需要我们在不断学习和实践中提高程序设计水平。“程序”是人的智力产品。从理论上说，程序是永远不会损坏的。实际上，程序在整个生存周期都会根据需要进行修改、维护，都可能产生错误。所有的硬件产品都允许有误差，但程序错误是不允许的，它有时甚至会产生悲剧性的后果。程序的生产和维护比硬件产品复杂得多。所以，计算机科学界期望有一套工程化的方法进行程序的开发维护。为了体现这种工程思想，程序就要伴随着一套开发、维护、使用的文档。程序加上这些相关文档称为软件。

1.2 C++语言的字符集与词汇

所有的语言系统都是由字符集和规则集组成的。“字符”是语言的不可分解的最基本语法单位。按照规则，由字符可以组成“词”，由词组成“表达式”、“句子”，由各种句子又可以构成

“函数”、“程序”。我们学习一门语言，就是要掌握程序设计语言的规律以及如何根据实际问题应用规则编写程序。

1.2.1 字符集

C++语言的字符集是 ASCII（American Standard Code for Information Interchange）码的子集，包括：

26 个小写字母 a b c d e f g h i j k l m n o p q r s t u v w x y z
26 个大写字母 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
10 个数字 0 1 2 3 4 5 6 7 8 9
其他符号 (空格) ! " # % & ' ( ) * + - / : ; < = > ? [\ ] ^ _ { | } ~ .

1.2.2 词汇

单词是语言中有意义的最小语法单位。根据构成规则，一个单词由一个或多个字符组成。下面介绍 C++语言的词汇。

1. 关键字

关键字又称保留字。关键字是系统预定义的语义符。C++语言不允许对关键字重定义。根据语言版本不同，关键字会有所增减。

下面列举出 C++语言常用的关键字。全部关键字可查阅 MSDN 帮助文档。

array bool break case catch char class const continue default delete
do double else enum extern false float for friend goto if inline int
long namespace new nullptr operator private protected public return short
sizeof static struct switch template this throw true try typedef
typename union unsigned virtual void while

2. 标识符

标识符是由程序员定义的命名符。例如，常量、变量、对象、函数、类型、语句标号等的命名。C++标识符语法是：以字母或下画线开始，由字母、数字和下画线组成的符号串。

注意： ① 关键字是特殊的标识符，C++规定不能使用关键字作为用户标识符。

② C++语言中，字母大小写敏感。例如，Aa 和 aa 是两个不同的标识符。

③ C++语言没有规定标识符的长度（即字符个数）。但不同编译系统有不同的识别长度，例如，有的系统识别 32 个字符。

【例 1-3】 判断以下标识符的正确性。

合法标识符有： a x1 no_1 _a2c sum Name
不合法标识符有： 2a x+y α π a,b a&b const

标识符命名除了符合上述规则外，应该尽可能做到“见名知义”，以提高程序的可读性。例如，年龄用 age，名字用 name，总和用 sum 等。

3. 运算符

运算符是对数据进行操作的简洁表达，以单词的形式调用系统预定义函数。许多符号与数学中的表示形式相同。例如：

+ 加 - 减 * 乘 / 除 > 大于 < 小于
>= 大于或等于 <= 小于或等于 == 等于 != 不等于

4. 分隔符

分隔符用于在程序中分隔不同的语法单位，便于编译系统识别。例如，有说明语句：

```
int a, b, c;
```