高 / 等 / 院 / 校 / 用 / 书

C++ 程序设计

C++ CHENGXU SHEJI

■主　编　陈恒鑫　熊　壮
■副主编　杨广超　王　宁
　　霍敏霞

GAODENG YUANXIAO YONGSHU
C++ CHENGXU SHEJI

重庆大学出版社
http://www.cqup.com.cn

高 / 等 / 院 / 校 / 用 / 书

C++ 程序设计

C++ CHENGXU SHEJI

■ 主　编　陈恒鑫　熊　壮
■ 副主编　杨广超　王　宁
　　霍敏霞

GAODENG YUANXIAO YONGSHU
C++ CHENGXU SHEJI

重庆大学出版社

内容提要

本教材的知识结构建立在读者已基本掌握了C语言并具有基本的程序设计能力基础上，通过本教材的讨论和学习，掌握C++语言在面向过程程序设计方面对C语言的扩充，并以此为基础向学生传授面向对象程序设计的基本思想、方法和技能，培养学生严谨的程序设计思想、灵活的思维方式及较强的动手能力。

教材分为两个部分，第一部分包括第1章到第3章，主要讨论C++语言在面向过程程序设计上对C语言的增强和扩充方面的知识。第二部分包括第4章到第8章，主要讨论C++语言面向对象程序设计方面的基础知识。本教材在附录中还提供了ASCII码表、使用Visual C++ 6.0集成环境开发C/C++程序的基本方法等重要学习资料。

本教材在C语言的基础之上讨论了C++语言的应用基础，内容深入浅出、语言流畅、例题丰富，适合作为程序设计语言课程教材，对于程序设计爱好者也是极佳的参考书。

图书在版编目(CIP)数据

C++程序设计/陈恒鑫，熊壮主编. —重庆：重庆大学出版社，2016.2

ISBN 978-7-5624-9677-9

Ⅰ.C… Ⅱ.①陈…②熊… Ⅲ.C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028420号

C++程序设计

主　编　陈恒鑫　熊　壮

副主编　杨广超　王　宁　霍敏霞

策划编辑：章　可　陈一柳

责任编辑：陈一柳　　版式设计：陈一柳

责任校对：张红梅　　责任印制：张　策

*

重庆大学出版社出版发行

出版人：易树平

社址：重庆市沙坪坝区大学城西路21号

邮编：401331

电话：(023) 88617190　88617185(中小学)

传真：(023) 88617186　88617166

网址：http://www.cqup.com.cn

邮箱：fxk@cqup.com.cn (营销中心)

全国新华书店经销

重庆升光电力印务有限公司印刷

*

开本：787×1092　1/16　印张：16　字数：360千

2016年2月第1版　2016年2月第1次印刷

ISBN 978-7-5624-9677-9　定价：32.00元

前 言

C++语言是目前使用最广泛的面向对象程序设计语言之一，具有易编写、易维护、易移植、运行高效等特点。C++程序设计课程是各类高等院校理工类专业的必修或重要选修课程。在C++程序设计课程的教学过程中，如何让学生能够更快地入门程序设计技术，如何更快、更深入地理解面向对象程序设计思想，如何提高编写程序和调试程序的实际能力，一直都是C++程序设计课程教师努力思考和实践的问题。本教材在介绍C++语言的语法和解决问题方法的同时，将对上述三方面问题的思考融入到所述内容之中，期望能够更好地引导和帮助读者尽快提高使用C++语言进行程序设计的实际能力。

本教材基于读者已基本掌握C语言并具备初步的程序设计能力基础之上。本书总体结构上分为两个部分，第一部分包括第1章到第3章，主要内容有：C++语言在数据表示和数据输入输出方面对C语言的扩充，C++在函数及其参数传递方面增加的重要特性；C++程序中使用C语言风格处理字符串数据的方法、C++程序中使用string类处理字符串数据的方法；C++语言的文件数据处理基础等。通过第一部分内容的学习，读者可以在C语言程序设计的基础上顺利过渡到C++语言程序设计。第二部分包括第4章到第8章，主要内容有：类与对象的概念以及对象的使用方法；继承与派生的概念，公有继承、私有继承、保护继承等继承方式的概念和使用；多态性的基本概念，运算符重载，虚函数和抽象类；类模板的概念、定义和简单应用，STL编程；异常概念、异常处理以及标准异常处理类的使用。通过第二部分内容的学习，读者可以掌握面向对象程序设计的基本思想、方法和技能。

本书选用Microsoft Visual C++ 6.0作为教学环境，书中的所有教学示例都在Microsoft Visual C++ 6.0集成开发环境中通过。附录中还提供了ASCII码表、使用Visual C++ 6.0集成环境开发C/C++程序的基本方法等重要学习资料。

本书适于高等院校理工类各专业本专科作为程序设计语言类课程教材，同时可作为计算机专业本专科学生、计算机应用开发人员、程序设计爱好者、计算机等级考试应试者在学习程序设计语言和程序设计技术时的参考教材。

本书由陈恒鑫、熊壮主持编写，各章节编写分工如下：陈恒鑫（第5章、第6章），熊壮（第1章、第2章）、杨广超（第7章、第8章）、王宁（第3章）、霍敏霞（第4章），全书由陈恒鑫、熊壮进行内容调整、修改，统一定稿。

本书在编写和出版过程中一直得到重庆大学教务处、重庆大学计算机学院和重庆大学

出版社领导的支持和帮助。重庆大学出版社的各位编辑老师为本书的编辑、出版做了大量的工作，编者在此表示衷心的感谢。

限于编者水平，书中错误和不妥之处在所难免，恳请读者不吝赐教。

联系地址：重庆大学计算机学院。

E-mail：chenhengxin@ cqu. edu. cn，xiongz@ cqu. edu. cn

编　者

2016 年 1 月

目 录

第1部分 从C过渡到C++

第2部分 面向对象程序设计基础

第 1 部分
从 C 过渡到 C ++

C++语言并不是一种纯粹的面向对象程序设计语言，在程序设计的过程中，既可以使用C++语言设计面向过程的应用程序，也可以用C++语言设计面向对象的应用程序。

C++语言继承了C语言的所有特点，具有与C语言相同的数据描述方式和程序的基本控制结构。C++语言在数据的描述、数据的输入输出、函数参数的传递、字符串数据的处理、文件数据的处理等方面，不但可以直接使用从C语言继承而来的处理形式，同时又增强了语言在上述各方面的处理能力。

第1章

数据处理与程序基本结构

1.1 数据的表示和数据的输入输出

1.1.1 程序基本结构

C++语言不但能够编写面向对象的程序，也可以像C语言一样编写面向过程的程序。C++程序设计语言从C语言发展而来，继承了C语言的各种特点。特别是在编写面向过程的程序时，无论是在数据的表示形式、语句的书写形式还是程序结构上，都与C程序类似。下面是C程序和C++面向过程程序结构形式的比较：

```
/* C程序结构 */
#include <stdio.h>                    //包含输入输出头文件
int main()
{
  //程序其他代码
  return 0;
}

//C++程序结构
#include <iostream>                   //包含输入输出流头文件
using namespace std;                  //导入标准命名空间
int main()
{
  //程序其他代码
  return 0;
}
```

比较上面两个单函数构成的C程序和C++程序可以看出,C++面向过程程序从程序的基本构成成分、主函数的结构以及注释语句的书写方法等方面都与C程序都非常类似。在书写C++面向过程程序时,很多地方可以直接借鉴C程序的处理方法,包括:数据的描述、语句的书写、程序基本控制结构等。

C++语言的基本控制结构和C语言完全相同,采用与C语言相同的控制结构表现了分支控制、循环控制、多重循环等程序设计基本技术。

1.1.2 数据表示

与C语言类似,C++语言也是一种强类型语言,也必须遵循"先定义后使用"的数据使用原则。C++语言的数据类型分为基本数据类型和导出数据类型两大类。

C++语言的基本数据类型在与C语言相同的字符型(char)、整型(int)、单精度实型(float)、双精度实型(double)、空类型(void)的基础之上,又增加了布尔型(bool)数据。C++语言的导出数据类型在C语言的数组、指针、结构体、共用体、枚举的基础之上,又增加了引用和类。

布尔型(bool)也称为逻辑型,在C++语言中用于处理逻辑数据。布尔型数据占用一个字节存储单元,取值只有true(真)和false(假)两个。为了向下兼容C语言,C++语言中也将0值看成false(假),将非0值看成true(真)。

布尔型变量的定义、初始化、赋值方式与其他基本数据类型相同,下面的代码段说明了布尔变量的使用方法:

```
bool b1 = true, b2;                    //定义布尔型变量 b1 和 b2,并将 b1 初始化为真值
b2 = b1;                               //将 b1 赋值给 b2
```

例1.1 求[1,100]区间内的绝对素数。绝对素数的条件是:一个素数,如果颠倒后还是素数,这个素数就称为绝对素数。例如:13是素数,31也是素数,所以13和31都是绝对素数。

```
/* Name:ex0101.cpp
   布尔型数据使用示例。
*/
#include <iostream>
using namespace std;
bool isprime(int n);                   //判断 n 是否素数函数
int revint(int n);                     //求取 n 的颠倒数函数
int main()
{
  int n,ren;
  for(n = 1;n <= 100;n ++)
  {
    ren = revint(n);
```

```
        if(isprime(n)&&isprime(ren))
            cout<<n<<endl;
    }
    return 0;
}
int revint(int n)
{
    int rn=0;
    while(n)
    {
        rn=10*rn+n%10;
        n/=10;
    }
    return rn;
}
bool isprime(int n)
{
    int i;
    if(n<=1)
        return false;
    else if(n==2)
        return true;
    for(i=2;i<n;i++)
        if(n%i==0)
            return false;
    return true;
}
```

上面程序中，函数 isprime 使用布尔型作为函数的返回值类型，当参数 n 是素数时返回 ture，否则返回 false。

1.1.3 数据的输入输出

C++程序中，也可以使用 C 语言的数据输入输出方法，即通过使用 scanf 函数和 printf 函数实现数据的输入输出，但 C++语言更提倡使用标准输入输出对象 cin 和 cout 来实现程序中的数据输入和输出。

1. 输出流对象 cout

cout 对象是 C++语言提供的标准输出对象（亦称为输出流对象），其作用是使用标准

输出设备(显示器)输出程序中的信息。cout 对象必须与流插入运算符(<<)一起使用,使用多个流插入运算符可以将多个输出数据传送给 cout 对象。在使用输出流对象的 C ++ 程序中必须包含下面两条语句:

```
#include <iostream>
using namespace std;
```

例 1.2 cout 流对象使用示例。

```
/* Name: ex0102.cpp
   输出流对象 cout 使用示例。
*/
#include <iostream>
using namespace std;
int main()
{
   cout <<"欢迎进入 C ++ 程序设计广阔天地" <<endl;
   cout <<2016 <<'\n';
   cout <<"100.5 +200 * 3 =" <<100.5 +200 * 3 <<endl;

   return 0;
}
```

上面程序演示了使用输出流对象 cout 配合插入运算符进行数据输出的情况,其中插入的 endl 与换行符'\n'意义相同,都表示换行的意思。

2. 输入流对象 cin

cin 对象是 C ++ 语言提供的标准输入对象(亦称为输入流对象),其作用是使用标准输入设备(键盘)为程序中的数据对象提供数据。cin 对象必须与流提取运算符(>>)一起使用,使用多个流提取运算符可以分别为多个数据对象输入数据。在使用输入流对象的 C ++ 程序中必须包含下面两条语句:

```
#include <iostream>
using namespace std;
```

例 1.3 cin 流对象使用示例。

```
/* Name: ex0103.cpp
     输入流对象 cin 使用示例。
*/
#include <iostream>
using namespace std;
int main()
{
   int a,b;
```

```
    double c,d;
    char s1[100],s2[100];
    cout <<"? a: ";
    cin >>a;
    cout <<"? b: ";
    cin >>b;
    cout <<"? c&d: ";
    cin >>c >>d;
    cout <<"? s1: ";
    cin >>s1;
    cout <<"? s2: ";
    cin >>s2;
    cout <<"a =" <<a <<",b =" <<b <<endl;
    cout <<"c =" <<c <<",d =" <<d <<endl;
    cout <<"s1: " <<s1 <<endl;
    cout <<"s2: " <<s2 <<endl;

    return 0;
}
```

上面程序演示了使用 cin 对象配合提取运算符进行数据输入的方法，在每个数据输入语句前面都使用输出流对象 cout 配合插入运算符进行输入数据提示信息显示。程序一次运行情况如下所示：

```
? a: 100                          //以下是数据输入部分
? b: 200
? c&d: 123.456 0.002 354
? s1: string1
? s2: string2
a =100,b =200                     //以下是数据输出部分
c =123.456,d =0.002 354
s1: string1
s2: string2
```

3. 格式化数据输出

使用输出流对象 cout 进行数据输出时，无论什么类型的数据，都能够自动按照正确的默认格式进行输出。如果要求程序按照某种规定的格式显示输出数据，则需要在实现输出的 C ++语句中嵌入格式控制符，达到控制输出数据格式的目的。C ++语言的格式控制符在头文件 iomanip 中定义，要使用格式控制符必须在程序中包含该头文件。C ++语言常用格式控制符及其意义见表 1.1。

表 1.1　C++常用格式控制符

控制符	功能描述
dec	设置基数为 10 进制
hex	设置基数为 16 进制
oct	设置基数为 8 进制
endl	插入换行符,并刷新流
ends	插入空字符
setfill(c)	设置填充字符为 c
setprecision(n)	设置输出数据有效位数(含整数部分和小数部分)为 n 位
setw(n)	设置输出数据域宽为 n 个字符
setiosflags(ios::fixed)	设置浮点数显示数据
setiosflags(ios::scientific)	设置指数(科学技术法)显示数据
setiosflags(ios::left)	设置输出数据左对齐
setiosflags(ios::right)	设置输出数据右对齐
setiosflags(ios::skipws)	设置忽略输出数据的前导空格
setiosflags(ios::uppercase)	设置 16 进制数据大写字母(A ~ F)输出
setiosflags(ios::lowercase)	设置 16 进制数据小写字母(a ~ f)输出

在表 1.1 所列出的格式控制符中,最常使用的是 setw 和 setprecision。

setw 为每个输出的数据项指定输出宽度,即用多少个字符位置来显示数据。当输出数据宽度小于 setw 指定的宽度时,空出部分(默认在左边)用空格填充。当输出数据的宽度大于 setw 指定的宽度时,数据按照实际要求输出。setw 仅对紧跟在其后的输出数据项有效,对每一个输出数据项都要用单独的 setw 进行域宽指定。

setprecision 用于设定输出数据的有效位数,有效位数包含了数据的整数部分和小数部分。当输出数据的有效位数小于指定的有效位数时,输出数据按本身实际数据输出。当输出数据的有效位数大于指定的有效位数时,则对输出数据在指定的有效位数后一位进行四舍五入后输出。

例 1.4　编程序求出所有的“水仙花数”,并求出这些数据的平均值。要求显示时每个“水仙花数”占 6 个字符宽度,平均值显示 4 位有效数据。

```
/* Name: ex0104.cpp
   格式控制符使用示例。
*/
#include <iostream>
#include <iomanip>
using namespace std;
int main()
```