

国家精品课程主讲教材配套用书

高等学校计算机程序设计课程系列教材

# C++程序设计教程

## 学习指导(第2版)

罗建军 朱丹军 顾刚 卫颜俊 编著  
冯博琴 审



高等教育出版社  
Higher Education Press

国家精品课程主讲教材配套用书  
高等学校计算机程序设计课程系列教材

# C++ 程序设计教程学习指导

## (第2版)

罗建军 朱丹军 顾刚 卫颜俊 编著  
冯博琴 审

高等教育出版社

## 内容提要

本书是与罗建军等编著的《C++ 程序设计教程(第2版)》(高等教育出版社出版)配套的习题与实验指导书。

全书包括三部分,第一部分是习题解答,给出了《C++ 程序设计教程(第2版)》各章习题的标准答案,并对每个题目的要点和编程时可能出现的问题进行了分析和讲解;第二部分是 C++ 语言的上机指南,介绍了几种常用 C++ 编译环境的使用方法;第三部分是案例分析,通过 3 个案例对结构化编程和面向对象编程的要素进行总结,从而使读者加深对解题思路的理解,掌握使用 C++ 编程解决实际问题的方法。

本书可作为高等学校理工类各专业 C++ 程序设计语言课程的辅导教材或参考书,也可各类工程技术人员进行 C++ 编程练习和上机训练提供指导。

## 图书在版编目(CIP)数据

C++ 程序设计教程学习指导/罗建军等编著. —2 版.

—北京:高等教育出版社,2007.8

ISBN 978-7-04-022190-9

I. C… II. 罗… III. C 语言-程序设计-高等学校-教学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字(2007)第 110350 号

|      |     |      |     |      |     |      |     |
|------|-----|------|-----|------|-----|------|-----|
| 策划编辑 | 张 龙 | 责任编辑 | 彭立辉 | 封面设计 | 于文燕 | 责任绘图 | 尹 莉 |
| 版式设计 | 陆瑞红 | 责任校对 | 胡晓琪 | 责任印制 | 宋克学 |      |     |

出版发行 高等教育出版社  
社 址 北京市西城区德外大街 4 号  
邮政编码 100011  
总 机 010-58581000

经 销 蓝色畅想图书发行有限公司  
印 刷 北京地质印刷厂

开 本 787×1092 1/16  
印 张 15.75  
字 数 380 000

购书热线 010-58581118  
免费咨询 800-810-0598  
网 址 <http://www.hep.edu.cn>  
<http://www.hep.com.cn>  
网上订购 <http://www.landaco.com>  
<http://www.landaco.com.cn>  
畅想教育 <http://www.widedu.com>

版 次 2004 年 8 月第 1 版  
2007 年 8 月第 2 版  
印 次 2007 年 8 月第 1 次印刷  
定 价 20.00 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号 22190-00

## 第2版前言

自2005年本书第1版出版以来,作为与《大学C++程序设计教程》配套的辅导与实验教材,西安交通大学的万余名学生使用了本书。在教学活动中我们以各种形式听取了学生大量的意见和建议,同时随着主教材的修订和更名[更名为《C++程序设计教程(第2版)》],本书亦相应更名为《C++程序设计教程学习指导(第2版)》,并对相关内容做了较大的改动。

除了修正原书中的错误外,第2版主要做了以下修订和调整工作:

(1) 根据《C++程序设计教程(第2版)》内容和章节的修订,对“习题解答”部分的相关内容进行了调整,给出了各章新增习题的参考答案。

(2) 将各章习题的参考答案使用标准C++重新进行了编码和调试,不再使用C语言保留的头文件形式。

(3) 为便于组织实验教学,增加了第21章(上机实验内容与安排),一共设计了16个实验。

(4) 主教材第1版中的部分内容以案例的形式调整到本书的第三部分。

需要强调的是,这里给出的仅是参考答案,解题的方法都不是唯一的。编写本教材的目的是希望能对读者有所启发,起到抛砖引玉的作用,使读者通过上机实践,真正理解程序设计的基本思想,正确灵活地使用语言中的各种要素,熟练掌握各种集成化开发环境,从而获得用程序设计解决实际问题的经验和技巧。

国家精品课程“计算机程序设计”的课程网站为 <http://programming.xjtu.edu.cn>。该网站提供与本书同步的全交互性网上辅导答疑讨论区、最新教学辅助课件、全程教学录像、上机实验指导及示例程序源码、优秀学生作业、课程大纲、教学计划等资源,所有内容均免费开放,并由专人负责不断更新,可供教师教学和学生自学使用。

本书的修订得到了西安交通大学冯博琴教授的指导,并由他审核了书稿,在此表示感谢。参加本书修订工作的主要人员有朱丹军、罗建军和顾刚。由于编者水平所限,不足之处在所难免,恳请教师和读者指正。

编著者

2007年4月

## 第 1 版前言

学习过计算机程序设计语言的人往往会有这样的感觉,看书或听老师讲课时,各种语言要素和编程方法的叙述都非常简单,老师讲的内容基本上都能理解,但需要自己面对问题去编程时却又无从下手,上机调试时更是错误百出,花四五个小时都调不通一个简单的程序,因此感到程序设计课程太难。

计算机程序设计语言是一门实践性很强的课程,与其他课程相比,它更加注重学生的实际动手编程能力。有再高的理论水平,编写不出符合要求、能正确运行的程序都毫无用处。上机实践是学习程序设计的重要教学环节,只有通过上机实践,才能真正理解程序设计的基本思想,正确灵活地使用语言中的各种要素,熟练掌握各种集成化开发环境,从而获得用程序设计解决实际问题的经验和技巧。

近年来,西安交通大学计算机教学实验中心在计算机基础教育的理论和实践等方面进行了一系列探索和革新,在程序设计课程教学中,摒弃了重教轻学,重理论轻实践的弊端,更新教学理念,将教学重点放在培养学生的编程能力上,教学模式也从传授知识转为培养能力。罗建军等编著的《大学 C++ 程序设计教程》(高等教育出版社,2004 年 8 月)和《Visual C++ 程序设计案例教程》(高等教育出版社,2004 年 8 月)也是这一系列改革成果的一部分,其目标是使学生掌握使用 C++ 设计应用程序的基本技能,了解面向对象和结构化程序设计的方法,能够编写、调试和运行实用、规范、可读性好的 C++ 程序,并通过多个案例掌握 Visual C++ 的基本使用方法。本书是与《大学 C++ 程序设计教程》配套的辅导与实验教材,旨在帮助读者在上机实践的过程中少走一些弯路,多得到一些启发。

本书包括 3 个部分。

第一部分是习题解答,给出了《大学 C++ 程序设计教程》各章习题的参考答案。为了便于读者学习和理解解题思路,本书不只是简单地给出了源程序,而是首先对题目的编程要点进行分析,再给出程序清单和运行结果,最后对编程时可能出现的问题或得出的结果进行解释,使读者不仅知其然,还能够知其所以然。

第二部分是 C++ 语言上机指南,简单地介绍了几个 C++ 编译环境的上机使用方法。在《大学 C++ 程序设计教程》一书中,我们强调了所有的程序都是标准 C++ 程序,即所谓的“与平台无关”。因此,这里除了介绍了目前常用的 Visual C++ 集成开发环境外,还介绍了一个基于 GCC C/C++ 编译器的集成开发环境 Quincy 2002(可从与本书配套的精品课程网站上下载),相信会给读者带来一个新的选择,对理解什么是“与平台无关”有所帮助。

第三部分是 C++ 案例分析,对应于结构化编程和面向对象编程,通过两个案例对 C++ 语言本身的要素进行了总结,读者可以观察到解决一个问题由简单到复杂、由仅

## II 第1版前言

---

具有演示功能到具有实用性的过程,也可以加深对使用 C++ 编程解决实际问题的方法和解题思路的理解。

需要指出的是,无论是习题解答还是对案例的分析,解题的方法都不是唯一的,这里给出的仅是参考答案,希望能对读者有所启发,起到抛砖引玉的作用。读者应该可以自行设计出更加简洁、高效的程序。

本书的构思和编写得到西安交通大学冯博琴教授(博士生导师、国家级教学名师、教育部非计算机专业计算机课程教学指导分委员会主任委员)的多方指导,并由他审核了书稿,在此表示感谢。参加本书编写工作的都是西安交通大学计算机教学实验中心从事程序设计教学和科研的教师,主要人员有朱丹军(第9~13章、18~20章)、罗建军(第1~8章、17章)、卫颜俊(第14~16章)。在本书的编写过程中,作者也曾与杨琦、崔舒宁等诸多同事进行了多次交流,他们提供了很多好的解题思路,受益匪浅;研究生黄锦翻译了部分英文资料,在此一并表示感谢。由于作者学识浅陋,编写时间仓促,书中错误在所难免,希望读者不吝指教。

编著者

2005年3月

# 目 录

## 第一部分 习题解答

|                            |    |                              |     |
|----------------------------|----|------------------------------|-----|
| 第 1 章 C++ 语言简介 .....       | 3  | 第 10 章 类和对象(二) .....         | 68  |
| 第 2 章 基本数据类型与表<br>达式 ..... | 8  | 第 11 章 继承 .....              | 80  |
| 第 3 章 控制结构 .....           | 13 | 第 12 章 多态性 .....             | 94  |
| 第 4 章 数组与字符串 .....         | 23 | 第 13 章 模板与异常处理 .....         | 103 |
| 第 5 章 函数 .....             | 33 | 第 14 章 标准库和输入/输<br>出流 .....  | 111 |
| 第 6 章 指针 .....             | 39 | 第 15 章 链表 .....              | 121 |
| 第 7 章 函数与指针 .....          | 46 | 第 16 章 程序设计与软件开<br>发技术 ..... | 130 |
| 第 8 章 结构化设计方法 .....        | 55 |                              |     |
| 第 9 章 类和对象(一) .....        | 63 |                              |     |

## 第二部分 C++ 语言上机指南

|                                                        |     |                                          |     |
|--------------------------------------------------------|-----|------------------------------------------|-----|
| 第 17 章 Visual C++ 6.0 集<br>成开发环境 .....                 | 135 | 编辑器 .....                                | 140 |
| 17.1 简介 .....                                          | 135 | 17.4 Developer Studio 的文件<br>处理功能 .....  | 142 |
| 17.1.1 菜单和工具栏 .....                                    | 136 | 17.5 Visual C++ 程序的编<br>译、连接和运行 .....    | 144 |
| 17.1.2 Developer Studio<br>窗口 .....                    | 136 | 17.6 查看和修改编译、连接<br>错误 .....              | 145 |
| 17.1.3 菜单选项、快捷键和<br>工具栏 .....                          | 136 | 17.7 运行错误的判断与<br>调试 .....                | 146 |
| 17.1.4 用 Developer Studio 编<br>写和调试简单的 C++<br>程序 ..... | 136 | 17.8 基本调试手段 .....                        | 147 |
| 17.2 Visual C++ 的帮助<br>功能 .....                        | 139 | 17.9 注解在调试中的作用 .....                     | 148 |
| 17.3 Developer Studio 的文本<br>编辑器 .....                 | 140 | 17.10 条件编译 .....                         | 149 |
|                                                        |     | 17.11 Developer Studio 的跟踪<br>调试功能 ..... | 150 |

## II 目录

|        |                          |     |        |                                    |     |
|--------|--------------------------|-----|--------|------------------------------------|-----|
| 17.12  | 如何在程序中使用 MFC 类库 .....    | 152 | 18.8.3 | Editor 选项 .....                    | 174 |
| 17.13  | 使用 FileView 标签 .....     | 152 | 18.9   | Quincy 工具 .....                    | 175 |
| 17.14  | 使用 ClassView 标签 .....    | 153 | 18.9.1 | Grep .....                         | 175 |
| 第 18 章 | Quincy 2002 集成开发环境 ..... | 156 | 18.9.2 | 艺术风格 .....                         | 176 |
| 18.1   | 简介 .....                 | 156 | 第 19 章 | Visual Studio .NET 下的 C++ 编程 ..... | 178 |
| 18.2   | 创建源程序 .....              | 157 | 第 20 章 | C++ Builder 使用指南 .....             | 181 |
| 18.3   | 文件保存 .....               | 158 | 第 21 章 | 上机实验内容与安排 .....                    | 185 |
| 18.4   | 编辑源程序 .....              | 159 | 实验 1:  | C++ 集成开发环境的使用 .....                | 185 |
| 18.4.1 | 输入文本 .....               | 159 | 实验 2:  | 基本数据类型与表达式 .....                   | 186 |
| 18.4.2 | 匹配花括号 .....              | 160 | 实验 3:  | 控制结构 .....                         | 186 |
| 18.5   | 程序的编译、连接和运行 .....        | 160 | 实验 4:  | 数组与字符串 .....                       | 187 |
| 18.6   | 调试程序 .....               | 162 | 实验 5:  | 函数 .....                           | 187 |
| 18.6.1 | 单步调试 .....               | 162 | 实验 6:  | 指针 .....                           | 187 |
| 18.6.2 | 断点 .....                 | 164 | 实验 7:  | 函数与指针 .....                        | 188 |
| 18.6.3 | 检查和修改变量 .....            | 164 | 实验 8:  | 结构化设计方法 .....                      | 188 |
| 18.6.4 | 观察变量 .....               | 166 | 实验 9:  | 类和对象(一) .....                      | 189 |
| 18.6.5 | 停止调试程序 .....             | 168 | 实验 10: | 类和对象(二) .....                      | 189 |
| 18.7   | 项目 .....                 | 168 | 实验 11: | 继承 .....                           | 190 |
| 18.7.1 | 建立一个项目 .....             | 168 | 实验 12: | 多态性 .....                          | 190 |
| 18.7.2 | 向项目中添加文件 .....           | 169 | 实验 13: | 模板与异常处理 .....                      | 191 |
| 18.7.3 | 从项目中删除文件 .....           | 171 | 实验 14: | 标准库和输入/输出流 .....                   | 191 |
| 18.7.4 | 创建项目的可执行程序 .....         | 171 | 实验 15: | 链表 .....                           | 192 |
| 18.8   | 设置选项 .....               | 171 | 实验 16: | 程序设计与软件开发技术 .....                  | 192 |
| 18.8.1 | Build 选项 .....           | 171 |        |                                    |     |
| 18.8.2 | Run 选项 .....             | 173 |        |                                    |     |

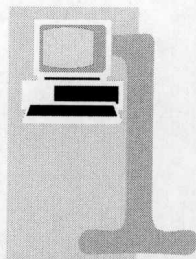
## 第三部分 C++ 案例分析

|        |                         |     |        |                    |     |
|--------|-------------------------|-----|--------|--------------------|-----|
| 第 22 章 | 案例分析 1: 求 $\pi$ 值 ..... | 195 | 第 24 章 | 案例分析 3: 哈希查找 ..... | 222 |
| 第 23 章 | 案例分析 2: 集合类 .....       | 206 |        |                    |     |

# 第一部分

## 习题解答





## C++ 语言简介

## 第1章

**习题 1-1:** 修改例 1-1 的 Hello World! 程序,使其能够在计算机屏幕上显示:“I am a student, and I like programming!”。

**要点分析:** 仿照例题中的程序结构和语法形式,按照题目要求的形式进行输出即可。

**程序清单:**

```
// Exercise 1-1: 屏幕上显示: I am a student, and I like programming!
#include <iostream>           //包含基本输入/输出库文件
using namespace std;         //使用名字空间 std
int main( )                  //主函数名
{
    cout << "I am a student, and I like programming!" << endl; //屏幕显示语句
    return 0;                //表示程序顺利结束
}
```

**输出结果:**

I am a student, and I like programming!

**小结:** 这是一个最简单的 C++ 程序。通过对该习题的编写、调试和运行,读者可以了解 C++ 程序的基本结构和要素,熟悉实验环境和方法。

**习题 1-2:** 修改例 1-3 的加法计算器程序,编写一个乘法计算器程序。

**要点分析:** 确定程序所需的变量数量和类型,定义所需的 3 个变量(分别对应乘数、被乘数和结果),提示用户输入后,进行相应的计算并输出结果。

**程序清单:**

```
// Exercise 1-2: 乘法计算器
#include <iostream>
using namespace std;
int main( )
{
    // ... (code for multiplication calculator) ...
    return 0;
}
```

```

double a, b, c;
cout << "please input two numbers: ";
cin >> a >> b;
c = a * b;
cout << a << " * " << b << " = " << c << endl;
return 0;
}

```

输入与输出:

Please input two numbers: 2.7 4.2

2.7 \* 4.2 = 11.34

小结:

这是一个简单的求值程序。通过该例子,可以了解标识符、关键字、输入/输出等 C++ 程序的基本要素的用法,在使用某一个变量前,必须对该变量进行类型说明。

习题 1-3:

修改例 1-4 的生日卡程序,使其能够输入和显示日期。

要点分析:

确定存储用户输入信息的数据类型并按照要求输出。仿照例题中收信人、发信人名字的定义方法,用字符型数组存储发信日期,按照生日卡的形式进行输出。

程序清单:

```

// Exercise 1-3:显示生日卡
#include <iostream>
using namespace std;
int main( )
{
    char name1[41], name2[41], date[15]; //增加一个数组存放日期
    cout << endl << "Please input your friend's name: ";
    cin >> name1;
    cout << endl << "Please input your name: ";
    cin >> name2;
    cout << endl << "Please input the date(year-month-day): ";
    cin >> date;
    cout << endl << "===== " << endl;
    cout << "My dear " << name1 << ", " << endl;
    cout << "Happy birthday to you!" << endl;
    cout << "yours," << endl;
    cout << " " << name2 << endl;
    cout << " " << date << endl;
    cout << "===== " << endl;
    return 0;
}

```

输入与输出:

```

Please input your friend's name: Zhu
Please input your name: Luo
Please input the date(year - month - day): 2005 - 1 - 25

=====
My dear Zhu,
    Happy birthday to you!

                                yours,
                                Luo
                                2005 - 1 - 25

=====

```

小结: 这是一个简单的输入/输出例子, 注意字符串的构造和显示方法。

习题 1-4: 参考例 1-5, 使用梯形法计算下式定积分的值。

$$\int_{-1}^1 (\sin x + e^x) dx$$

其中, 积分区域等分数可取为 200, 并比较计算结果和手算结果。

要点分析: 按照梯形法求定积分公式的要求, 定义所需的若干变量, 然后将一般求和部分直接带入, 将  $\Sigma$  和式部分转换为循环结构即可。

程序清单:

```

// Exercise 1-4: 计算定积分
#include <iostream>
#include <cmath> // 包含标准数学函数的 cmath 函数库
using namespace std;
int main( )
{
    double a, b; // 双精度类型变量: 积分的下限和上限
    double h; // 双精度类型变量: 积分步长
    double sum; // 双精度类型变量: 工作变量, 最后为积分值
    int n; // 整型变量: 积分区间等分数
    int i; // 整型变量: 循环工作变量
    // 根据题意确定积分的下限、上限和积分区间等分数
    a = -1.0;
    b = 1.0;
    n = 200;
    h = (b - a) / n; // 计算小区间长度
    // 为工作变量赋初值: 先计算无须循环运算的部分

```

```

sum = (sin(a) + exp(a) + sin(b) + exp(b))/2;
// 循环计算公式中的  $\Sigma$  和式
for(i=1;i<n;i=i+1)
    sum = sum + sin(a+i*h) + exp(a+i*h);
// 完成计算,变量 sum 中存放积分结果
sum = sum * h;
// 输出计算结果
cout << "The result is " << sum << endl;
return 0;
}

```

输出结果:

The result is 2.35042

小结: 除了顺序结构以外,C++还提供了循环结构,用于解决重复运算问题。实际上,正是这种结构,使计算机的计算能力得到充分发挥。

习题 1-5: 仿照例 1-6,编写一个计算矩形面积的程序。

要点分析: 将求矩形面积的代码单独编写为一个函数 area(),其两个形式参数分别是矩形的长和宽。在主函数中通过函数名对该函数进行调用,同时只要将所要求的矩形的相应参数数据代入即可得到所需的结果。

程序清单:

```

// Exercise 1-5: 计算矩形面积
#include <iostream>
using namespace std;
double area(double length, double width)
{
    double a;
    a = length * width;
    return a;
}
int main()
{
    cout << "长为 3, 宽为 4 的矩形面积是: " << area(3, 4) << endl;
    cout << "请输入待计算的新的矩形的长和宽: ";
    double l, w;
    cin >> l >> w;
    cout << "长为 " << l << ", 宽为 " << w << " 的矩形面积是: " << area(l, w)
    << endl;
    return 0;
}

```

输入与输出:

长为 3, 宽为 4 的矩形面积是: 12

请输入待计算的新的矩形的长和宽: 3.5 6.3

长为 3.5, 宽为 6.3 的矩形面积是: 22.05

小结:

这是一个简单的函数编写和调用的例子。通过该例子, 可以进一步理解 C++ 程序的一般结构, 了解模块化程序设计的思路。

基础类库

例 2-1: 求三角形的面积。该程序通过输入三角形的三条边长, 计算出三角形的面积。如果输入的三条边长不能构成一个三角形, 则输出 "Error data!"。

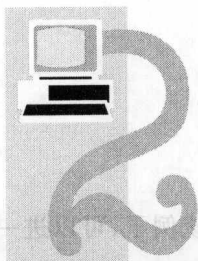
例 2-1: 求三角形的面积

```

\\ Exercise 2-1: 求三角形的面积
#include <iostream>
#include <cmath>
using namespace std;

int main()
{
    double a, b, c, s, area;
    cout << "Please input a, b, c: ";
    cin >> a >> b >> c;
    if(a+b >= c && a+c >= b && b+c >= a)
        cout << "Error data!" << endl;
    else
    {
        s = (a+b+c)/2;
        area = sqrt(s*(s-a)*(s-b)*(s-c));
        cout << "area = " << area << endl;
    }
    return 0;
}

```



## 第2章

## 基本数据类型与表达式

**习题 2-1:** 为例 2-2 添加数据检验部分。给出三边长, 检验其是否能构成一个三角形的方法是检查是否任意两边的和均大于第三边。如果检验不合格, 输出信息“Error data!”。

**要点分析:** 关系表达式和逻辑表达式的构造。在从用户处得到三边长后, 检查是否任意两边的和均大于第三边以确定其是否能构成一个三角形, 然后使用海伦公式求出对应的三角形面积。

**程序清单:**

```
// Exercise 2-1: 求三角形面积
#include <iostream>
#include <cmath>
using namespace std;
int main( )
{
    double a, b, c, s, area;
    cout << "please input a, b, c =? ";
    cin >> a >> b >> c;
    if(a + b <= c || a + c <= b || c + b <= a)
        cout << "Error data!" << endl;
    else
    {
        s = (a + b + c) / 2;
        area = sqrt(s * (s - a) * (s - b) * (s - c));
        cout << "area = " << area << endl;
    }
    return 0;
}
```

输入与输出:

```
please input a, b, c = ? 1 2 3
```

```
Error data!
```

小结:

数据检验可避免用户随意输入可能引起的错误,保证了程序的正确性。

习题 2-2:

输入两个角度值  $x$ 、 $y$ , 计算如下式子的值。

$$\frac{\sin(|x| + |y|)}{\sqrt{\cos(|x| + |y|)}}$$

要点分析:

按照题目给出的公式,确定所需的变量并构造相应的表达式。

程序清单:

```
// Exercise 2-2: 计算求值
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    double x, y, z;
    cout << "请输入两个角度值:";
    cin >> x >> y;
    double t = fabs(x) + fabs(y);
    z = sin(t) / sqrt(cos(t));
    cout << "z = " << z << endl;
    return 0;
}
```

输入与输出:

```
请输入两个角度值:0.1 1.2
```

```
z = 1.86302
```

小结:

这是一个简单的数学计算问题,只要声明恰当的变量并严格按照公式形式进行程序语句的转化即可。

习题 2-3:

编写一个程序,要求完成以下要求:

(1) 提示用户输入任意的 3 个小数。

(2) 显示这 3 个小数。

(3) 将这 3 个小数相加,并显示其结果。

(4) 将结果按四舍五入方法转换成整数并显示。

要点分析:

数据类型的转换。

程序清单:

```
// Exercise 2-3: 小数输入、计算及显示
```