



普通高等教育计算机类核心课程群教改系列教材

C++与面向对象程序设计 实践教程

左伍衡 胡同森 主 编
张克明 强 彦 张应博 副主编



科学出版社
www.sciencep.com

TP312/3547

2010



普通高等教育计算机类核心课程群教改系列教材

C++与面向对象程序设计 实践教程

左伍衡 胡同森 主 编

张克明 强 彦 张应博 副主编

科学出版社

北 京

内 容 简 介

“C++语言程序设计”或“面向对象程序设计”是一门实践性要求非常强的课程。作为该课程的全面而系统的实践教材,本书由浅入深、循序渐进地介绍了 C++语言的主要概念、知识点及其简单应用和综合应用。全书内容主要包括基础实践(第1章至第9章)和综合实践(第10章至第12章)两大部分,其中基础实践篇每章包含知识要点、典型例题、典型习题与上机实验四个模块;综合实践篇包括数值计算、数据管理、图形处理三章,均为综合性实验,可作为课程设计、大型实验素材和综合实训的指导。附录部分包括主流开发环境(平台)使用介绍、上机报告与综合实践报告样例,以方便广大读者参考、模仿和学习。

本书既可作为相关课程的辅助教材,也可作为相关课程的课程设计、大型实验以及实训/培训的主教材,还可供广大 C++爱好者参考。

图书在版编目(CIP)数据

C++与面向对象程序设计实践教程/左伍衡, 胡同森主编. —北京: 科学出版社, 2010

(普通高等教育计算机类核心课程群教改系列教材)

ISBN 978-7-03-026656-9

I. ①C… II. ①左…②胡… III. ①C 语言—程序设计—高等学校—教学参考资料 IV. ①TP312

中国版本图书馆 CIP 数据核字(2010)第 019476 号

责任编辑: 陈晓萍 / 责任校对: 耿 耘

责任印制: 吕春珉 / 封面设计: 东方人华

科学出版社 出版

北京东黄城根北街 16 号

邮政编码: 100717

<http://www.sciencep.com>

铭浩彩色印务有限公司 印刷

科学出版社发行 各地新华书店经销

*

2010 年 2 月第 一 版 开本: 787×1092 1/16

2010 年 2 月第一次印刷 印张: 16 1/4

印数: 1—3 000 字数: 382 000

定价: 26.00 元

(如有印装质量问题, 我社负责调换〈环伟〉)

销售部电话 010-62134988 编辑部电话 010-62135120-8003

版权所有, 侵权必究

举报电话: 010-64030229; 010-64034315; 13501151303

前 言

C++语言具有全面兼容 C 和支持面向对象的特点。由于面向对象方法的突出优点，目前它已经成为应用最广泛的面向对象的程序设计语言之一。针对以培养应用型人才为主要目标的普通高等院校相关专业的程序设计语言教学的实际情况，结合多年的一线教学经验积累，我们特别编写了本书。

本书在体系结构的确定和内容的选择上，坚持了定位明确、由浅入深、循序渐进、点面结合的方针，书写上力求简单明白，通俗易懂，所以本书的可读性、实用性和先进性俱佳，既适合教学，又适合自学。本书各章节都提供了典型例题，以提高读者的编程能力和技巧。书中所有程序示例均在 Visual C++ 6.0 下进行了调试并运行通过。本书分为基础实践篇和综合实践篇两大部分。其中基础实践篇主要按照课程内容分为九章，每章包含知识要点、典型例题、典型习题与上机实验四个模块。

第 1 章，绪论：概要介绍了 C++ 的发展历程、特点，C++ 程序结构及编程环节。

第 2 章，数据类型、表达式与语句：概要介绍了 C++ 语言的基础知识，如数据类型、表达式和基本语句及其使用。

第 3 章，函数与程序结构：概要介绍了 C++ 语言的函数和程序结构的相关概念及其运用。

第 4 章，数组、结构体与指针：总结了数组、结构体与指针等常用的用户自定义类型的知识点及其运用。

第 5 章，类和对象：概述了面向对象设计方法的核心概念——类和对象及其成员以及如何利用类来解决具体问题。

第 6 章，继承与派生：概述了类的继承特性，总结了不同继承方式下的基类成员的访问控制问题、构造函数和析构函数以及多重继承问题。

第 7 章，多态性：概述了多态性的概念及类型、函数重载、运算符重载、虚函数、纯虚函数以及抽象类等知识点及其简单运用。

第 8 章，模板：概述了函数模板、类模板以及 STL 相关概念及其简单运用。

第 9 章，流与异常处理：概述了输入流、输出流及异常处理相关概念及其简单运用。

综合实践篇主要根据题目的应用领域类型分为三大类：数值计算、数据管理和图形处理。

基础实践篇中的上机实验和综合实践篇的综合实践任务都根据难易程度提供必做题和选做题两种类型，以供不同学习基础的学生自行选择。

附录部分包括主流开发环境（平台）使用介绍、上机报告与综合实践报告样例，以方便学生参考、模仿和学习。

本书由左伍衡、胡同森、张克明、强彦、张应博等共同编写，其中第 8、9 章由西安外事学院张克明编写；第 10 章由太原理工大学赵涓涓编写；第 11 章由太原理工大学强彦编写；第 12 章由大连理工大学城市学院张应博编写；附录由浙江工业大学之江学院赵



端阳、袁鹤编写；其余部分由浙江工业大学的左伍衡、胡同森共同编写。浙江工业大学之江学院冯志林、曹平、刘邦明、陈珊珊、张晓娇老师以及王一波、王侃、张帆、黄翊翔和蒋湧同学参与了本书的部分章节编写、程序录入和校对工作。在本书的编写过程中，编者参考了同类书籍、面向对象程序设计课程自考历年试题、ACM 竞赛试题以及其他网络资料和文档。在此，向这些书籍、资料的作者表示诚挚的谢意。在本书编写过程中，得到了浙江工业大学各级领导、科学出版社以及浙江大学城市学院周苏的大力支持与帮助，在此表示衷心感谢。

本书先后获得浙江工业大学校级重点教材建设项目和浙江省重点教材建设项目资助，在此，要特别感谢 2009—2011 年浙江省高等学校本科计算机科学与技术及计算机基础课程教学指导委员会主任委员何钦铭教授、副主任委员陈庆章教授、胡维华教授、凌云教授和秘书长陈天洲教授以及其他委员等多位评审专家对本书的充分肯定和支持。

本书主要面向“C++语言程序设计”或“面向对象程序设计”等课程的初学者。本书既可作为相关课程的辅助教材，也可作为相关课程的课程设计、大型实验以及实训/培训的主教材，还可供广大 C++ 爱好者参考。

由于编者水平有限，书中难免有不当和欠妥之处，敬请各位专家和读者不吝批评指正。

左伍衡

2009 年 12 月

目 录

前言

基础实践篇

第 1 章 绪论	3
1.1 知识要点	3
1.2 典型例题	5
1.3 典型习题	7
1.4 上机实验	8
第 2 章 数据类型、表达式与语句	11
2.1 知识要点	11
2.2 典型例题	14
2.3 典型习题	16
2.4 上机实验	26
第 3 章 函数与程序结构	28
3.1 知识要点	28
3.2 典型例题	31
3.3 典型习题	33
3.4 上机实验	36
第 4 章 数组、结构体与指针	39
4.1 知识要点	39
4.2 典型例题	49
4.3 典型习题	56
4.4 上机实验	70
第 5 章 类和对象	73
5.1 知识要点	73
5.2 典型例题	79
5.3 典型习题	82
5.4 上机实验	89
第 6 章 继承与派生	92
6.1 知识要点	92
6.2 典型例题	95
6.3 典型习题	100
6.4 上机实验	105



第7章 多态性	107
7.1 知识要点	107
7.2 典型例题	113
7.3 典型习题	117
7.4 上机实验	125
第8章 模板	127
8.1 知识要点	127
8.2 典型例题	133
8.3 典型习题	138
8.4 上机实验	140
第9章 流与异常处理	141
9.1 知识要点	141
9.2 典型例题	146
9.3 典型习题	153
9.4 上机实验	156

综合实践篇

第10章 数值计算	161
10.1 典型案例分析	161
10.2 综合实践任务	180
第11章 数据管理	182
11.1 典型案例分析	182
11.2 综合实践任务	199
第12章 图形处理	201
12.1 典型案例分析	201
12.2 综合实践任务	208

附 录

附录A 编程环境使用简介	211
A.1 Borland C++ Builder 集成编程环境	211
A.2 MinGW Developer Studio 集成编程环境	216
A.3 Microsoft Visual C++集成编程环境	219
附录B 上机实验报告样例	225
附录C 综合实践报告样例	231
参考文献	251

基础实践篇



基础实践目的

本篇是学习完“C++程序设计”或“面向对象程序设计”课程各章节后进行的一次复习、总结和提高。其目的在于加深学生对“C++程序设计”或“面向对象程序设计”课程各章节中的主要知识点、基本概念的理解与掌握及其简单综合应用，不断提高学生阅读与分析、调试与编写程序的能力。考虑到学生在学习基础和编程能力等方面存在较大差异，基础实践应充分体现“重视基础、循序渐进、因材施教”的教学模式，以学生为认知主体，充分调动学生学习的积极性和能动性，重视学生对基本概念与原理的理解和掌握，加强对学生阅读与分析、调试与编写程序的能力培养。



基础实践总体要求

- 建议学生及时复习课本各章节相关内容。
- 建议学生消化与吸收知识要点与典型例题中的内容。
- 独立完成典型习题作业，并根据自身情况自行选择上机实验任务。
- 根据上机实验题目要求编写相应程序，并上机调试与测试。
- 最后要求提交完整的上机实验报告（格式可参考附录 B 中相关内容）。

第 1 章 绪 论

实践要求:

- 了解 C++语言的发展简史及其主要特点。
- 了解 C++程序的主要特点及其运行过程。
- 熟悉 C++语言编程环境或开发平台的使用。

重点: C++程序的主要特点及其运行过程。

难点: C++语言编程环境的使用。

1.1 知 识 要 点

1.1.1 C++语言的发展简史

C 语言是 20 世纪 70 年代发展起来的。在 Unix 操作系统中, 90%的代码是用 C 语言编写的。作为一种广为流行的结构化程序设计语言, C 语言具有灵活方便、运算符丰富、目标代码效率高、可移植性好、模块化能力强等优点。然而, 随着程序规模与复杂性的增长, 程序中的数据结构变得与这些数据的操作同样重要。在大型结构化程序中, 一个数据结构可能被许多过程处理, 修改此数据结构将影响到所有这些过程。例如, 当程序超过 50000 行, 开发人员达几十人时, 系统维护工作量大, 系统整体性能难以保证。1983 年, 美国 AT&T 公司与贝尔 (Bell) 实验室推出 C++, 克服了上述缺点。Borland 公司先后推出 Turbo C++、Borland C++和 C++ Builder 不同版本, Microsoft 公司也陆续推出 Visual C++的不同版本。C++语言是一门还在发展的语言。

1.1.2 C++语言的特点

C++语言是目前应用最广泛的高级程序设计语言之一。C++语言是对 C 语言的扩充 (或称为 C 语言的超集), 继承了 C 语言高效、灵活等特点, 完善了 C 语言的类型检查、代码重用、数据抽象机制, 扩充了对面向对象程序设计的支持, 具体特点如下。

1) C++语言支持数据封装

封装是指把对象属性和操作结合在一起, 构成独立的单元, 它的内部信息对外界是隐蔽的, 不允许外界直接存取对象的属性, 只能通过有限的接口与对象发生联系。类是数据封装的工具, 对象是封装的实现。C++类中包含私有、公有和保护成员。类的访问控制机制体现在类的成员中可以有公有成员、私有成员和保护成员。对于外界而言, 只需要知道对象所表现的外部行为, 而不必了解内部实现细节。封装体现了面向对象方法的“信息隐蔽和局部化原则”。

2) C++语言中通过发送消息来处理对象之间的交互

消息指对象之间在交互通信中所传送的信息。消息由三部分构成: 消息名、接收消



息的对象标识和参数。一个对象向另一个对象发送消息请求某项服务，接收消息的对象响应该消息，进行所要求的服务，并把操作的结果返回给请求服务的对象。

3) C++语言支持继承性

继承性是指子类（派生类）可以自动拥有父类（基类）的全部属性和服务。父类和子类是一般与特殊的关系。在定义一个子类时，可以把父类所定义的内容作为自己的内容，并加入若干新的内容。

4) C++语言支持多态性

多态性是指在基类中定义的属性和服务被子类继承后，可以具有不同的数据类型并表现出不同的行为。当一个对象接收到一个请求进行某项服务的消息时，将根据对象所属的类，动态地选用该类中定义的操作。

1.1.3 C++程序的组成

一个完整 C++ 程序包括注释部分、编译预处理部分（宏定义、文件包含和条件编译）和程序正文部分（类型定义、常变量定义、函数定义）。其中 C++ 注释有以下两种类型。

(1) 一对符号 “/” 与 “*/” 之间的内容称为多行注释。它可以占多行，是从 C 语言中继承来的一种注释形式。

(2) 一行中符号 “//” 之后的内容也称为单行注释。它只能占一行，是 C++ 语言特有的一种注释形式。

C++ 的编译预处理命令以 “#” 开头。C++ 提供了三类编译预处理命令：宏定义、文件包含和条件编译。

一个 C++ 程序至少且仅包含一个 `main()` 函数，也可以包含一个 `main()` 函数和若干个其他函数。其他函数可以是系统提供的库函数，也可以是用户根据需要自定义的函数。

1.1.4 C++程序的编辑、编译、连接和运行

不同开发平台所支持的 C++ 程序调试、运行环境不同，但一般都要经历以下四个基本环节。

1) 编辑

编辑是将编写好的 C++ 源程序（如在 Visual C++ 环境中，C++ 源程序文件后缀名为 `.cpp`）输入到计算机中，生成磁盘文件的过程。

2) 编译

编译器的功能是将程序的源代码转换成为机器代码的形式，称为目标代码（文件后缀名为 `.obj`）。

3) 连接

将用户程序生成的多个目标代码文件和系统提供的库文件中的某些代码连接在一起，由连接器生成一个可执行文件（文件后缀名为 `.exe`）。

4) 运行

一个 C++ 的源程序经过编译、连接后生成了可执行文件。只有可执行文件才能在计算机上运行。

目前提供的大多是集成开发环境（Integrated Development Environment, IDE），程序

的编辑、编译、连接与运行均可在同一环境中完成，并且采用菜单和快捷按钮方式操作，这对于提高程序的编写与调试效率有非常大的帮助。

1.2 典型例题

例 1.1 显示“I am a student”字符串，并通过简单的程序了解和熟悉主流 C++ 开发环境中进行程序编写、编译、调试、显示结果的过程。

解答：该源程序非常简单，代码如下所示：

```
#include<iostream>
using namespace std;
int main()
{
    cout<<"I am a student"<<endl;
    return 0;
}
```

下面以 Microsoft 公司的 Visual C++ 6.0 开发环境示范 C++ 程序编写、编译、调试、显示结果的过程，其他环境使用介绍请参考附录 A。

编辑过程如下。

1) 编辑

编辑是将 C++ 源程序输入计算机的过程。在 Visual C++ 6.0 主界面下，单击“文件”菜单，选择“新建”菜单项，即可出现“新建”对话框，如图 1.1 所示。在弹出窗口中的“文件”标签页下，选择 C++ Source File。在选项卡右面的“文件”编辑框里输入文件名，如“exa”，在“目录”编辑框里输入文件路径，如“d:\test”。单击“确定”按钮，将出现编辑窗口，如图 1.2 所示。

在图 1.2 所示的右端编辑窗口里输入相应 C++ 源程序并保存文件。

2) 编译

编译是使用系统提供的编译器将文本文件 exa.cpp 生成机器语言的过程，结果为 exa.obj，由于没有得到系统分配的绝对地址，还不能直接运行，如图 1.3 所示。选择“编

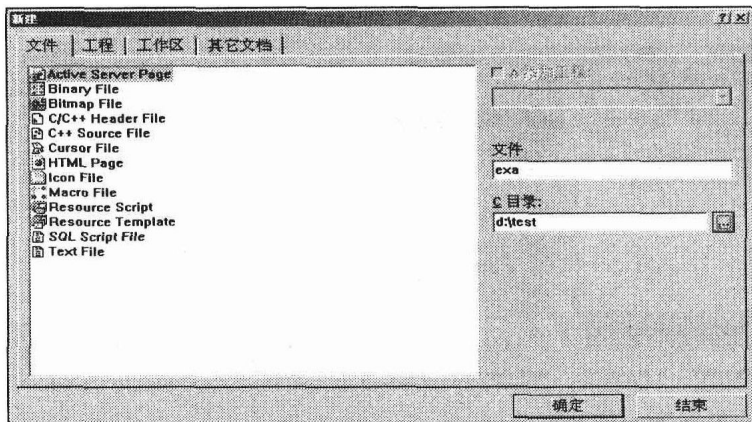


图 1.1 “新建”对话框

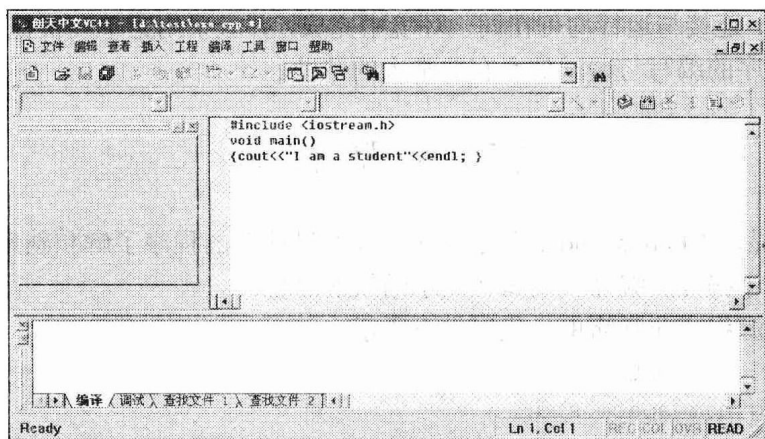


图 1.2 编辑窗口

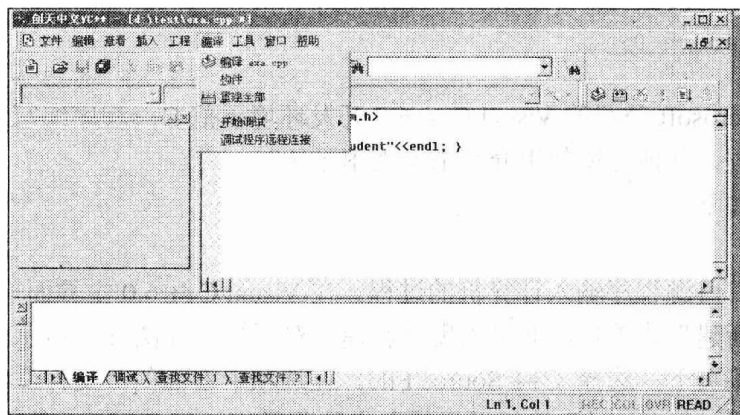


图 1.3 编译

译”菜单下的“编译 exa.cpp”，系统将完成对当前源程序的编译。按提示建立默认工程空间。编译无错时，将源程序 exa.cpp 编译为目标文件 exa.obj。

3) 连接

连接是将目标文件 exa.obj 转换为可执行文件 exa.exe 的过程。如图 1.4 所示，选择“编译”菜单下的“构件”exa.exe。连接无错时，将目标文件 exa.obj 连接为 exa.exe。

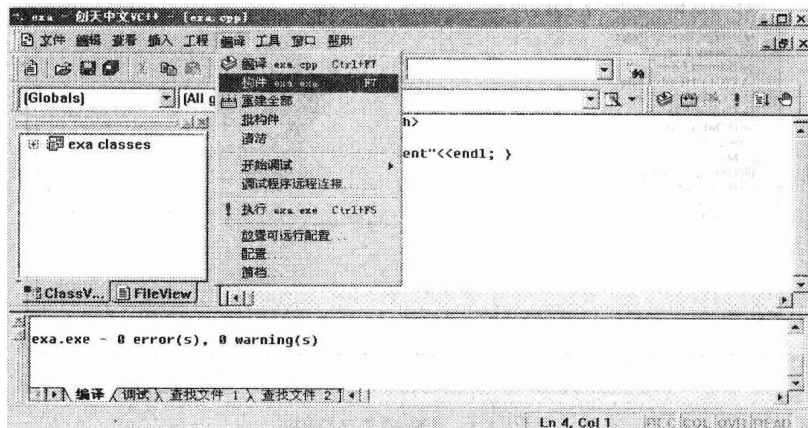


图 1.4 连接

4) 运行

运行是执行可执行文件 `exa.exe`，屏幕输出运行结果。如图 1.5 所示，选择“编译”菜单下的“执行 `exa.exe`”。运行结果如图 1.6 所示。

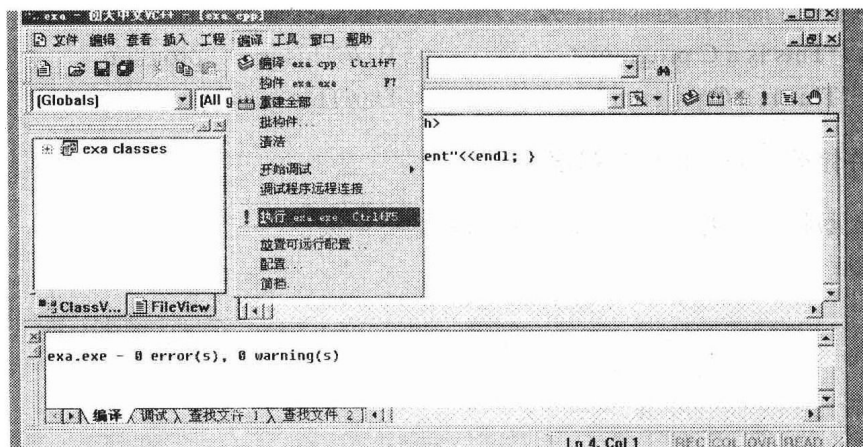


图 1.5 运行程序

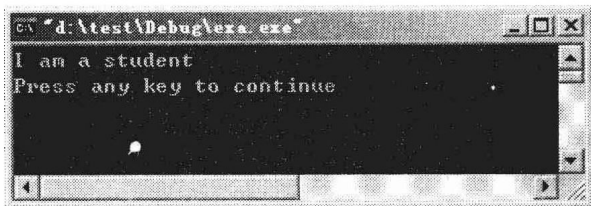


图 1.6 运行结果

1.3 典型习题

一、选择题

1. 在一个 C++ 程序中，`main` 函数的位置（ ）。
 - A. 必须在程序的开头
 - B. 必须在程序的后面
 - C. 可以在程序的任何地方
 - D. 必须在其他函数中间
2. 用 C++ 语言编制的源程序要变为目标程序必须要经过（ ）。
 - A. 解释
 - B. 汇编
 - C. 编辑
 - D. 编译
3. C++ 程序基本运行单位是（ ）。
 - A. 数据
 - B. 字符
 - C. 函数
 - D. 语句
4. C++ 程序中的语句必须以（ ）结束。
 - A. 冒号
 - B. 分号
 - C. 空格
 - D. 花括号
5. 执行 C++ 程序时出现的“溢出”错误属于（ ）错误。
 - A. 编译
 - B. 连接
 - C. 运行
 - D. 逻辑



6. 面向对象的()是一种信息隐蔽技术, 目的在于将对象的使用者与设计者分开。不允许使用者直接存取对象的属性, 只能通过有限的接口与对象发生联系。

- A. 多态性 B. 封装性 C. 继承性 D. 重用性

7. C++的合法注释是()。

- A. /*This is a C program/* B. // This is a C program
C. "This is a C program" D. //This is a C program//

二、分析并修改下列该程序的错误

若输入数据为: 2,5 (回车), 请写出其运行结果。

```
main()
{
    int sum=0;
    int a,b;
    cout<<"input a,b:";
    cin>>a>>b;
    sum=a+b;
    cout<<sum<<endl;
}
```

三、阅读程序, 写出下列程序的运行结果

```
#include<iostream>
using namespace std;
void main()
{
    int a(2),b(3),c(4);
    cout<<"c1="<<c<<endl;
    c=a+b;
    cout<<"c2="<<c<<endl;
}
```

1.4 上机实验

实验名称: C++集成开发环境

一、实验目的

- (1) 熟悉 C++的集成开发环境, 学习运行一个 C++程序的步骤。
- (2) 了解几种运算符的使用规则。
- (3) 掌握变量命名规则, 学会变量的定义和使用。

二、实验内容

1. 必做题

- (1) 使用一种 C++ 开发环境对以下 C++ 程序进行编辑、编译、连接和运行。

```
#include<iostream>
using namespace std;
void main()
{
    cout<<"Welcome to C++ Programming!"<<endl;
}
```

- (2) 对以下 C++ 程序进行编辑、编译、连接和运行，并试着加上必要的注释内容。

```
#include<iostream>
using namespace std;
int main()
{
    int a,b,c;
    int add(int x,int y);
    cin>>a>>b;
    c=add(a,b);
    cout<<"a+b="<<c<<endl;
    return 0;
}

int add(int x,int y)
{
    int z;
    z=x+y;
    return(z);
}
```

2. 选做题

- (1) 编写程序。从键盘输入两个正整数，输出其和与差。例如，输入 7、61，输出为 68、-54。

- (2) 给出下列程序的运行结果。

```
#include<iostream>
using namespace std;
int main()
{
    int a=32,b;
    double c=2.7,d;
    char e='D',f;
    b=a+c;
    d=a+c;
    f=a+e;
}
```



```
    cout<<"b="<<b<<endl;
    cout<<"d="<<d<<endl;
    cout<<"f="<<f<< endl;
    cout<<b+d+f<< endl;
    return 0;
}
```

三、实验要求

- (1) 结合上课内容，写出并调试程序，要给出测试数据和实验结果。
- (2) 整理上机步骤，总结经验和体会。
- (3) 完成实验报告。