

应用型高等院校经管类系列实验教材·计算机

程序设计基础实验 ——C语言程序设计

王泽 鲜征征 黄承慧 / 主编

Cheng Xu She Ji Ji Chu Shi Yan



经济科学出版社
Economic Science Press

应用型高等院校经管类系列实验教材·计算机

编委 (PI) 目 录 附录 参考文献

王泽 鲜征征 黄承慧 (主编)

ISBN 978-7-302-50040-0

应用型高等院校经管类系列实验教材·计算机

ISBN 978-7-302-50040-0

程序设计基础实验

——C语言程序设计

王泽 鲜征征 黄承慧 / 主编

ChengXu Sheji JiChu ShiYan

经济科学出版社

北京市西城区德胜门内大街2号

邮编：100089

(二) 本系列实验教材由清华大学出版社出版，ISBN 978-7-302-50040-0



经济科学出版社
Economic Science Press

图书在版编目 (CIP) 数据

程序设计基础实验: C 语言程序设计 / 王泽, 鲜征征,
黄承慧主编. —北京: 经济科学出版社, 2010. 7

(应用型高等院校经管类系列实验教材·计算机)

ISBN 978 - 7 - 5058 - 9640 - 6

I. ①程… II. ①王…②鲜…③黄… 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 (2010) 第 128998 号

程序设计基础实验

——C 语言程序设计

王 泽 鲜征征 黄承慧 主编

经济科学出版社出版、发行 新华书店经销

社址: 北京市海淀区阜成路甲 28 号 邮编: 100142

教材编辑中心电话: 88191354 发行部电话: 88191540

网址: www.esp.com.cn

电子邮件: bailiujie518@126.com

北京密兴印刷厂印装

787 × 1092 16 开 9 印张 160000 字

2010 年 8 月第 1 版 2010 年 8 月第 1 次印刷

ISBN 978 - 7 - 5058 - 9640 - 6 定价: 18.00 元

(图书出现印装问题, 本社负责调换)

(版权所有 翻印必究)

总序

实践教学是高等教育本质的必然要求，是践行应用性人才培养的必经之路，是地方行业性教学型本科院校办学的重要特征。近几年来，各高校经济与管理类专业实验教学已经逐步开展，把实验教学作为教学改革的抓手、知识融合的平台以及联系社会的桥梁，然而如何进一步完善实验教学体系、提高实验实践教学水平与质量已经成为各高校亟待解决的问题。应用型高等院校经管类系列实验教材以提高高等院校经济与管理类专业实验教学的建设水平为目的，以实验教材建设为突破口，探讨高等院校经济与管理类实验教材的新方向、新思路、新内容、新模式。

本系列实验教材的编写紧紧围绕“知行合一，能力为尚，积淀特色，共享协作”的地方行业性教学型经济与管理类实验教学理念，贯彻以现代教育技术为基本手段，以实验资源共享与应用为条件，强化理论教学与实践教学互动与互补，“实践与理论相结合”和在“做中学”的指导思想，强调实验教材建设与实验课程建设、实验项目建设、实验教师队伍建设以及深化实验教学改革相结合，力图通过系列教材建设规范实验教学内容和实验项目，促进实验教学质量的提高。

(一) 本系列实验教材内容与教学方式符合实验教学规律和要求。具体表现在以下几个方面：

1. 实验教材以实验项目为章节，按如下体例编写：实验目的和实验要求；实验的基本原理；实验仪器、软件和材料或实验环境；实验方法和操作步骤；实验注意事项；数据处理和实验结果分析；实验报告。当然，对于不同的课程，根据其本身的学科特点，实验教材的编写体例并不完全一致。

2. 增加综合性、设计性、创新性实验项目的比例，并逐步将科研成果项目转化为教材的实验项目。

3. 与当前流行的实验平台软件或硬件及教材内容紧密结合，符合一般软件要求。

4. 充分体现以学生为主体，明确实验教学的内涵。实验教学过程体现以学生操作为主，教师辅导为辅，少量时间教师讲解，大部分时间学生操作的特点。

5. 按实验教学规律分配学时，并且有多余的实验项目供学生利用开放实验室自主学习。

6. 内容精练，主次分明，详略得当，文字通俗易懂，图表与正文密切配合。

(二) 本系列实验教材遵循实验教学规律，体现时代特色，总体来说，具有以下四个特点：

1. 与现代典型案例相结合。以培养应用型人才为原则，根据实验教学大纲，注重理论联系实际，教材具有较强的实践性、新颖性、启发性和适用性，有利于培养学生的实践能力和创新能力。

2. 建设形式新颖。实验教材分为纸质实验教材和网络资源的形式；纸质教材实验报告

尝试做成活页形式，或做成可撕下的带切割线形式；在纸质教材出版，配套建有供学生实验前和实验后学习使用的网络资源。

3. 实验内容创新。对于实验教材编写内容上的创新，一是凸显应用型人才培养特色实验项目，提高了综合性、设计性、创新性实验项目的比例；二是将教师的科研成果转化为本科学生实验教学项目。

4. 编写程序严格。对实验教材的申请立项的实验教材经由学院领导及专家进行立项审查；实验教材初稿经由相关同行专家给出鉴定，最终审核后，送交出版社评审出版。

本系列教材得到各方面人士的指导、支持和帮助，尤其是得到中国经济信息学会实验经济学与经济管理实验室专业委员会的专家，广东金电集团等多家业界人士，以及各高校同行老师们的支持和帮助，我们在此表示由衷的感谢。本系列实验教材尚处于探索阶段，作为一种努力和尝试，存在诸多不足之处，竭诚希望得到广大同行及相关专家的批评指正。

应用型高等院校经管类系列实验教材编委会

2009年12月

前 言

程序设计基础实验是计算机科学与技术等计算机相关专业低年级学生开设的一门专业基础实验课程，实验内容既要深入浅出，又要体现以培养应用型人才为专业教学特色。在积极倡导以现代教育技术为基本手段，以实验教学为中心的前提下，我们认为有必要编写此教材。

要学好程序设计基础，不仅要学习 C 语言程序设计的基本概念、方法、语法规则，更重要的是实践。只有通过实践体验，不断积累编程经验，才能真正提高程序设计能力。作为程序设计基础教程的辅助教材，通过本实验教材内容的实践，可以加强学生对 C 语言概念、方法和内容的巩固；有了本实验教材，可增强学生对课程实践重要性的认识深度和投入力度。对编程的思路、方法和技巧起指导、示范作用。

本书是集众多长期从事 C 语言教学工作的一线教师的经验和体会，并参考大量的国内外有关资料编写而成。本书共分四部分，一共设置了 21 个实验，各部分的特点综述如下：

第一部分基础实验包括 9 个实验，内容包括 C 语言程序开发环境和上机步骤、基本数据类型、运算符和表达式、控制结构、函数、数组、指针、结构体和共用体、编译预处理、文件等。通过这部分的实验教学，使读者熟悉 C 语言的基本语法结构、程序结构、编程思路及解题的基本思路。

第二部分综合实验包括银行存款大小写转换和数组、函数、指针的综合应用。通过这部分实验教学，加深读者对 C 语言的理解，真正能用 C 语言这个强有力的编程工具去解决实际问题。

第三部分课外实验包括 9 个实验，内容安排上由浅入深，供学有余力的读者练习。

第四部分特色实验内容以银行开户、存款、取款、转账、查询等为主要功能模块，设计实现一个银行常用业务管理信息系统。

为了帮助读者上机练习，本书附录中收录了 C 语言常见编程错误和 Visual C++ 6.0 调试方法，另附有实验教学中使用的实验报告模板样本，以规范文档格式。本书具有基础性、实用性和系统性，同时充分考虑了和其他教材内容的兼容性，适用于各类高等院校尤其是财经类院校计算机专业程序设计基础 C 语言程序设计课程的实验教学。

本书由王泽、鲜征征、黄承慧编写，全书由王泽主编和统稿。本书得到了北方交通大学实验教学中心王占刚博士，广东金融学院计算机系蔡焕夫教授、陈国君教授、肖伟才老师，美国阿拉巴马州立大学王家来博士等相关老师的大力支持和协助，在此表示衷心的感谢。

受编者水平所限，书中难免存在疏漏之处，恳请广大读者提出宝贵意见。

电子邮件地址：wzebox@gdutf.edu.cn。

目 录

第一部分 基础实验

实验一	C 语言运行环境及运行 C 程序	(1)
实验二	数据类型、运算符和表达式	(7)
实验三	C 语言的控制流程	(15)
实验四	C 程序函数调用	(21)
实验五	C 程序调用中的指针参数	(27)
实验六	数组在 C 程序中的使用	(32)
实验七	结构体、联合体在 C 程序中的使用	(37)
实验八	C 程序中的字符串处理	(42)
实验九	C 程序中的文件处理	(47)

第二部分 综合实验

实验十	银行存款金额大小写转换	(51)
实验十一	数组、函数和指针的综合应用	(59)

第三部分 课外实验

实验十二	计算到期银行存款本息	(69)
实验十三	简单计算器	(72)
实验十四	学习成绩字母制评价模型	(74)
实验十五	矩阵乘法	(78)
实验十六	绘制余弦函数曲线	(82)
实验十七	字符串比较	(84)
实验十八	奇数魔方矩阵	(87)
实验十九	约瑟夫斯 (Josephus) 问题	(90)
实验二十	Life 游戏	(93)

第四部分 特色实验

实验二十一 银行信息管理系统设计与实现	(99)
附录 A C 语言编程常见错误	(116)
附录 B Visual C ++ 6.0 的调试	(123)
附录 C ASCII 表	(128)
附录 D 实验报告参考样本	(130)
参考文献	(133)

第一部分

基础实验

实验一

C 语言运行环境及运行 C 程序

一、实验目的

1. 掌握在 Microsoft Visual C++6.0 环境中编辑、编译、连接和运行一个 C 语言源程序文件的过程。
2. 通过运行简单的 C 程序，初步了解 C 语言程序的特点。
3. 初步了解输入函数 `scanf` 和输出函数 `printf` 的使用方法。

二、实验环境

1. Windows XP/2000 操作系统。

2. Microsoft Visual C++ 6.0 软件。

三、相关知识

1. Visual C++ 简介。

(1) Visual C++ 是 Microsoft 公司的 Visual Studio 开发工具箱中的一个 C++ 程序开发包。Visual Studio 提供了一整套开发 Internet 和 Windows 应用程序的工具, 包括 Visual C++, Visual FoxPro, Visual Basic, Visual InterDev, Visual J++ 以及其他辅助工具, 如代码管理工具 Visual SourceSafe 和联机帮助系统 MSDN。Visual C++ 包中除包括 C++ 编译器外, 还包括所有的库、例子和为创建 Windows 应用程序所需要的文档。

(2) Visual C++ 软件包中的 Developer Studio 是一个集成开发环境 (IDE), 它集成了各种开发工具和 VC 编译器。程序员可以在不离开该环境的情况下编辑、编译、调试和运行一个应用程序。

(3) Developer Studio 中还提供向导和 MFC 类库, 这些都可以帮助程序员快速而正确地开发出应用程序。向导是一个通过一步步的帮助引导你工作的工具。Developer Studio 中包含三个向导, 用来帮助程序员开发简单的 windows 程序, 它们是: AppWizard, 它支持三类程序, 基于视图/文档结构的单文档应用、基于视图/文档结构的多文档应用程序和基于对话框的应用程序, 也可以生成最简单的控制台应用程序 (DOS 下用字符输入输出的程序)。ClassWizard 用来定义 Appwizard 所创建的程序中的类。利用 ClassWizard 在项目中添加类、为类增加处理消息的函数等。ActiveX Control Wizard 用于创建一个 ActiveX 控件的基本框架结构。ActiveX 控件是用户自定义的控件。它支持一系列定义的接口, 可以作为一个可再利用的组件。

(4) MFC 库是 Visual C++ 开发环境所带的类库, 利用类库中的类, 可以快速建立应用程序。使用 MFC 库开发应用程序可以减少很多工作量。

2. 项目开发过程。

(1) 在一个集成的开发环境中开发项目非常容易。建立一个项目的第一步是利用编辑器建立程序代码文件, 包括头文件、代码文件、资源文件等。然后, 启动编译程序, 编译程序首先调用预处理程序中的预处理命令 (如 #include, #define 等), 经过预处理程序处理的代码将作为编译程序的输入。编译对用户程序进行词法和语法分析, 建立目标文件, 文件中包括机器代码、连接指令、外部应用以及从该源文件中产生的函数和数据名。此后, 连接程序将所有的目标代码和用到的静态连接库的代码连接起来, 为所有的外部变量和函数找到其提供地点, 最后产生一个可执行文件。

(2) 可执行文件分为 Debug 和 Release 两个版本, Debug 版本用于程序的开发过程, 可以供调试程序使用。而 Release 版本作为最终的发行版本, 没有调试信息, 具有某种程度的优化。在编程的过程中, 可以采用 Debug 版本, 这样便于调试。

四、实验内容

1. 熟悉 Microsoft Visual C++6.0 编程环境，输入并运行一个 C 程序。
2. 调试运行程序并改正其中的错误。

五、实验步骤

1. 输入并运行一个简单的 C 程序，从而熟悉 Microsoft Visual C++6.0 编程环境，操作步骤如下：

(1) 启动 Microsoft Visual C++6.0 的方法：选择【开始】|【程序】|【Microsoft Visual Studio 6.0】|【Microsoft Visual C++6.0】图标，启动 Microsoft Visual C++6.0，主窗口如图 1-1 所示。

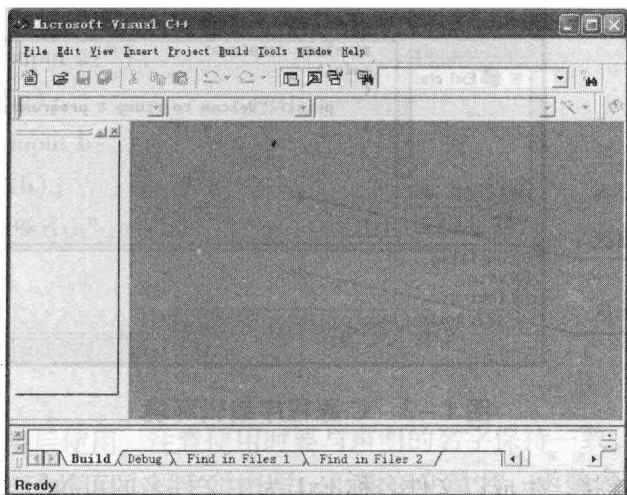


图 1-1 Microsoft Visual C++6.0 主窗口

(2) 选择菜单 File 中的 New，弹出一个对话框，单击此对话框的左上角的 File 选项卡，选择 C++ Source File 选项，如图 1-2 所示。

使用默认的文件存储路径则不必更改 Location（目录）文本框，但如果想在其他地方存储源程序文件，则需要对话框右半部分的 Location（目录）文本框中输入文件的存储路径，也可以单击右侧的“省略号…”来选择存储的目标文件夹。

然后，在右上方的 File 文本框中输入准备编辑的源程序文件的名字（如 Ex1.c）。

注意：如果不写后缀 c，系统会默认指定后缀为 .cpp，表示要建立的是 C++ 源程序。因此命名时要加上后缀 c，单击 OK 按钮后，就可以在编辑区域输入程序了。

(3) 在文件编辑区窗口中输入源程序，如图 1-3 所示。

(4) 单击图 1-3 工具栏上的 Build 按钮  或选择菜单 Build 中的 Compile Ex1.c，系统

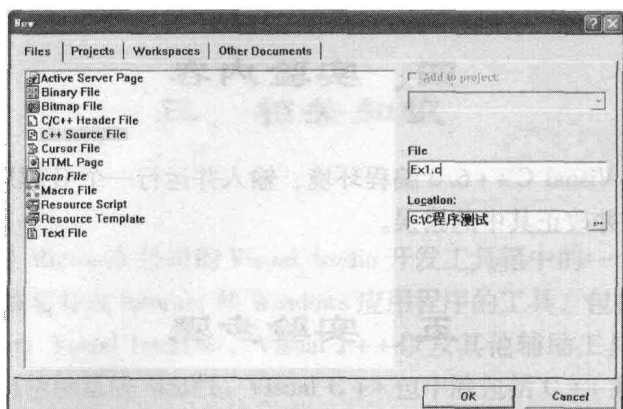


图 1-2 新建文件对话框

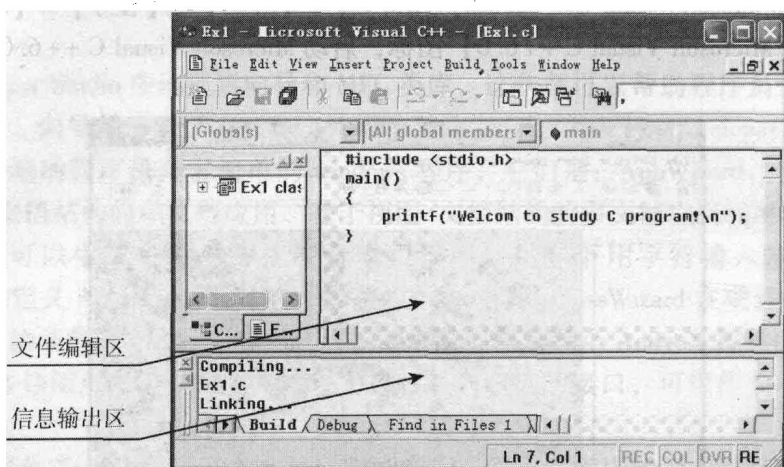



图 1-3 C 源程序编辑环境

对源程序进行编译和连接，生成以文件名称 Ex1 为主文件名的可执行文件 Ex1.exe。在编译和连接的过程中，系统发现程序有语法错误，则在图 1-3 中的下部分输出窗口中显示错误信息，包括错误的性质、原因和出错的位置。单击键盘上面的功能键 F4 或双击某条错误信息，在编辑区域右侧出现一个箭头，提示程序错误的位置。据此对源程序进行相应的修改，并重新编译和连接，直到程序没有错误为止。

(5) 单击工具栏上的 Execute Program 按钮图标或选择菜单 Build 中的 Execute Ex1.exe，执行编译后的程序。运行成功后，程序将在一个新打开的 DOS 窗口中运行并显示结果。在程序运行结果的后面显示一条提示信息“Press any key to continue”，这是 Visual C++ 自动加上的，此时按任意键将关闭 DOS 窗口，系统返回 Microsoft Visual C++6.0 主窗口。

2. 输入下列代码，观察程序运行结果。

```
#include <stdio.h>
main()
```

```
printf(" Hello everybody!\n");
```

修改上面程序，使其输出“祖国，祝您生日快乐！”。

3. 输入下列代码，观察程序运行结果。

```
#include <stdio.h>
main()
{
    /* 输出 2 个整数相加的结果 */
    printf("%d\n", 1 + 1);
}
```

修改上面程序，使其输出 123 + 456 的值。

4. 从键盘输入 2 个整数，输出两个数的和，程序代码如下，调试程序并改正其中的错误，并分析程序中每一行语句的作用。

```
#include <stdio.h>
main()
int a,b;
printf(" Please input a:");
scanf("%d",&a);
printf(" Please input b:");
scanf("%d",&b);
printf(" a + b = %d\n", a + b)
```

六、实验分析

1. 变量必须先声明后使用，注意使用时要与声明的名字保持一致。
2. 关键字及预处理命令用小写字母书写，不能在变量名、函数名、关键字中插入空格和空行。
3. 程序中以分号作为语句的结束符。
4. 一行只写一条语句。
5. 程序中用 `/*` 和 `*/` 包含起来的内容，称为注释，方便读者提高程序的可读性，C 编译程序不对其进行语法检查，`/*` 和 `*/` 中间不能有空格。
6. 输出函数 `printf()`，不要写成 `print()` 或者 `Printf()`。
7. 函数 `printf()` 或 `scanf()` 中的格式控制字符串要加上双引号。
8. 函数 `scanf()` 中的变量要加上取地址运算符 `&`。
9. 一个 C 源程序有且仅有一个 `main` 作为名字的函数，C 程序总是从 `main` 函数开始执行，并且与其位置无关，`main` 里面的语句用一对 `{}` 括起来。
10. C 程序中，输入函数格式：`scanf("输入数据格式", &变量名)`；输出函数格式：`printf("输出格式", 变量或表达式)` 或 `printf("输出字符串")`，函数 `scanf()` 的格式转换

说明符如表 1-1。

表 1-1 函数 scanf () 的格式转换说明符

格式转换说明符	用 法	举 例
%d 或%i	输入十进制整数: scanf ("%d", &x);	键盘输入如: 25
%o	输入八进制整数: scanf ("%o", &x);	键盘输入如: 072
%x	输入十六进制整数: scanf ("%x", &x);	键盘输入如: 0x71AF
%c	输入一个字符, 空白字符 (包括空格、回车、制表符) 时结束	键盘输入如: g
%s	输入字符串, 遇到第一个空白字符 (包括空格、回车、制表符) 时结束	键盘输入如: finance
%f 或%e %%	输入实数, 以小数或指数形式输入均可 输入一个百分号%	键盘输入如: 234.54 或 2.3454e2 键盘输入如: %

七、实验报告

实验结束后, 需要整理并书写实验报告, 实验报告格式见附录 D, 其内容包括:

1. 实验题目。
2. 实验内容。
3. 写出程序源代码。
4. 写出程序运行结果。
5. 对实验结果进行分析、总结。

数据类型、运算符和表达式

一、实验目的

1. 通过对表达式运算结果的分析，学习数据类型的概念，熟悉如何定义一个整型、字符型和实型的变量，以及对它们赋值的方法。
2. 通过输入数据到变量中参加表达式的运算，学习变量的编程方法。
3. 学会正确使用逻辑运算符及逻辑表达式。
4. 掌握常用运算符的优先级与结合性。
5. 学会检查给定程序的错误并通过调试信息纠正错误。
6. 进一步掌握输入函数 `scanf` 和输出函数 `printf` 的使用方法。
7. 进一步掌握 C 程序的编辑、编译、连接和运行的过程。

二、实验环境

1. Windows XP/2000 操作系统。
2. Microsoft Visual C ++ 6.0 软件。

三、相关知识

1. 计算机中数的表示。

在计算机中，任何数据都是用一定位数的二进制组合来表示，每个二进制位（bit）只能取值 0 或 1。例如， $(01000000010010001111010111000011)_2$ 或 $(4048F5C3)_{16}$ 是一个 32 位的二进制数。它可以代表整数 1 078 523 331，也可以用来表示其他数据（如字符、汉字、声音、图像、指令代码、网络协议等），称为数据的二进制编码。为了让程序里的每个数据都与计算机中一定长度（位数）的二进制数码组合相对应，C 语言将数据分为若干种不同

的数据类型，每种数据类型以其特有的长度、二进制编码以及定义在其上的一组运算的这三个要素而区别于其他类型。

2. 数据类型。

(1) C 语言的基本数据类型有整型和浮点型两种。前者为整型编码（原码和补码），后者为浮点编码。两种类型都具有 +（加）、-（减）、*（乘）、/（除）四种基本运算，但定义有所不同（如整数除法“ $3 / 4 = 0$ ”，而浮点除法“ $3.0 / 4.0 = 0.75$ ”）。除此之外，整型数中还定义%（取余数）运算；两种类型都定义有 =（赋值）、++（增量）、--（减量）以及由算术和赋值运算构成的复合运算等。

(2) 在实表达式“ $3 / 4 * 5$ ”中，3 和 4 为整型，它们的商也是整型，而整型是没有小数位的，因此， $3 / 4 = 0.75$ 中小数部分被截去，结果为 $3 / 4 * 5 = 0$ ，而在表达式“ $3.0 / 4.0 * 5$ ”中，3.0 和 4.0 为浮点型，它们的商也是浮点型，包含小数位，因此 $3.0 / 4.0 = 0.75$ ， $3.0 / 4.0 * 5 = 3.75$ 。又例如表达式“ $3\ 000\ 000 * 1\ 000$ ”中，3 000 000 和 1 000 都是整型，它们的积也是整型，而整型数在 32 位操作系统 Windows XP 和编译系统 Microsoft Visual C++ 6.0 下占 32 位，32 位二进制所能表达的最大正整数（原码）是 $(01111111111111111111111111111111)_2$ ，即十进制的 2 147 483 647，而 $3\ 000\ 000 * 1\ 000 = 3\ 000\ 000\ 000$ 超出了这个值，致使最高位为 1，根据负整数的表示方法和补码原理，它被解释为负整数 -1 294 967 296；而表达式 $3\ 000\ 000.0 * 1\ 000.0$ 中，3 000 000.0 和 1 000.0 都是浮点型，它们的积也是浮点型。而浮点数在计算机内部是采用“浮点编码”，分指数和尾数两部分表示的，32 位浮点型所能表示的实数的大小可达到 10^{38} 数量级（但其有效数字只能达到 7 位十进制），这样大的数值范围表示 3 000 000 000 自然没有问题。

3. 根据其长度（二进制位数）的不同，整型和浮点型又分别被分为若干个子类型（如表 2-1 和表 2-2 所示），其中数据范围和数量级范围是由其长度和二进制编码决定的。

表 2-1 整型的子类型

类型保留字	长度（字节）	二进制编码	数据范围	定义的运算
char	1	原码 + 补码	-128 ~ 127	+, -, *, /, %, ++, --
short	2	原码 + 补码	-32768 ~ 32767	+, -, *, /, %, ++, --
long	4	原码 + 补码	-2147483648 ~ 2147483647	+, -, *, /, %, ++, --
int	在 16 位系统中同 short；在 32 位系统中同 long			+, -, *, /, %, ++, --

表 2-2 浮点型的子类型

类型保留字	长度（byte）	二进制编码	数量级范围	有效数字	运算
float	4	浮点编码	$10^{-38} \sim 10^{38}$	7 位十进制	+, -, *, /, ++, --
double	8	浮点编码	$10^{-308} \sim 10^{308}$	15 位十进制	+, -, *, /, ++, --

4. 常用运算符的优先级与结合性见表 2-3。

表 2-3

常用运算符的优先级与结合性

优先级顺序	运算符种类	附加说明	结合方向
1	单目运算符	逻辑非! 按位取反~ 求负- ++ -- 类型强制转换等	右→左
2	算术运算符	* / % 高于 + -	左→右
3	关系运算符	< <= > >= 高于 == !=	左→右
4	逻辑运算符	除逻辑非之外, && 高于	左→右
5	赋值运算符	= += -= *= /= %= &= ^= != <<= >>=	右→左
6	逗号运算符	,	左→右

例如：设有语句 `int a=3`；执行语句 `a+=a-=a*a` 后，变量 `a` 的值为 `-12`，语句运算过程说明如下：

(1) * 号运算符优先级高于赋值运算符 `+=`，`-=` 的优先级，赋值运算符 `+=`，`-=` 优先级相等，先将语句 `a+=a-=a*a` 右边 “=” 号右侧表达式用括号括起来，即 `a -= (a*a)`。

(2) 再将 “=” 号左侧内容 “`a-`” 移到 “=” 号右侧，即 `a - (a*a)`。

(3) 在 “=” 号左侧补上目标变量名，即 `a = a - (a*a)`，此时， $a = 3 - (3 * 3) = -6$

(4) 然后，计算 `a += -6`，按照步骤 (2) 和 (3)，可得 $a = a + (-6) = -6 + (-6) = -12$ 。

C 运算符的优先级与结合性的复杂性可通过将先需要计算的表达式用圆括号括起来的方法来解决。如对于表达式：`year % 4 == 0 && year % 100 != 0 || year % 400 == 0`，只有明确了各种运算符的优先级才能正确对其进行计算，但如果将其写成如下形式：`(year % 4 == 0) && (year % 100 != 0) || (year % 400 == 0)`，即使忘记各种运算符的优先级，也能对其进行正确计算，在 C 语言中，圆括号 () 的优先级最高。

四、实验内容

1. 分析下列用 C 表达式计算 `128 * 2 + 1024` 值的程序。

```
#include <stdio.h>
main()
{
    printf("%d", 128 * 2 + 1024);
}
```

2. 编程计算下列表达式的值并输出到屏幕上。

(1) $\frac{3}{4} \times 5$

(2) $3\ 000\ 000 \times 1\ 000$

(3) $\frac{3.0}{4.0} \times 5$

(4) $3\ 000\ 000.0 \times 1\ 000.0$

(5) 2π