



全国高等农林院校“十一五”规划教材

C语言 程序设计

上机指导与习题解答

滕桂法 主编

 中国农业出版社

全国高等农林院校“十一五”规划教材

C 语言程序设计

上机指导与习题解答

■ 滕桂法 主编

中国农业出版社

图书在版编目 (CIP) 数据

C 语言程序设计上机指导与习题解答/滕桂法主编.

北京: 中国农业出版社, 2007. 1

全国高等农林院校“十一五”规划教材

ISBN 978-7-109-11025-0

I. C… II. 滕… 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 (2006) 第 149540 号

中国农业出版社出版

(北京市朝阳区农展馆北路 2 号)

(邮政编码 100026)

责任编辑 许 坚

中国农业出版社印刷厂印刷 新华书店北京发行所发行

2007 年 1 月第 1 版 2007 年 1 月北京第 1 次印刷

开本: 820mm×1080mm 1/16 印张: 6.5

字数: 146 千字

定价: 10.50 元

(凡本版图书出现印刷、装订错误, 请向出版社发行部调换)

内容简介

本教材是与滕桂法教授主编的《C 语言程序设计》一书配套使用的教学参考书。内容包括：Turbo C 2.0 集成开发环境下编辑、编译、调试和运行程序的方法；《C 语言程序设计》一书各章习题的参考答案及部分习题的详细解答；C 语言程序设计课程的上机实验安排，提供了学习本课程应当进行的 15 个实验（教材中的一章内容对应一或两个实验）；最后给出了一个利用 C 语言实现通讯录管理的综合实验，该实验涉及 C 语言中使用的大多数操作。作为附录还给出了 Turbo C 2.0 常用命令和常见错误提示信息。

教材中的习题参考答案全部上机调试通过。本教材实验、习题等练习内容丰富，概念清晰，具有启发性和综合性，和理论教学配合紧密，实用性强，是学习 C 语言的一本必备参考书。

本教材不仅可以作为《C 语言程序设计》的教学参考书，而且可以作为其他 C 语言教材的参考书，既可作为高等院校计算机专业或其他专业配套教材使用，也可以作为参加计算机等级考试者或其他计算机应用人员的参考书或培训教材。

C 语言程序设计

编写人员名单

主 编 滕桂法

副 主 编 王 芳 何国斌 常淑惠

编写人员 (按姓氏笔画排序)

王 芳 冯 勇 吕显强

杨断利 何国斌 张 隆

张慧英 滕桂法

C 语言程序设计

前言

C 语言程序设计

学习程序设计语言，上机实验是必不可少的实践环节。只有通过充分的实际上机操作，才能实现学习程序设计语言的目的——掌握其语法规则、掌握程序设计方法和提高程序开发能力。对于使用灵活、简洁的 C 语言，更需要通过大量的上机编辑、编译、调试和运行的实践来真正掌握它。

本教材共有 15 个实验，对应于《C 语言程序设计》教材，每章有 1~2 个实验。每个实验包括：实验目的、知识要点、实验内容和步骤、部分习题解答四部分，基本上覆盖了所有知识点，帮助读者通过上机实践更好地领会教材中的内容。为帮助读者更好地理解 and 掌握程序设计方法，提高程序设计开发能力，本教材在最后一次实验中通过一个小型的通讯录管理系统给出一个完整的、较综合和实用的例子，以便读者可将前面所学内容进行综合应用。这些实验可根据实际情况分配学时，建议每个实验 2 学时，综合实验可适当增加学时。在附录中给出了 Turbo C 2.0 常用命令、常见错误提示信息。

应该说明的是，本教材所给出的习题中程序的答案并不是惟一正确的解答，只是提供了一种参考方案。对于同一个题目可以编写出多个程序，我们给出的只是其中的一种。希望读者多尝试并进行比较，千万不要只是照抄照搬。另外，学习 C 语言除了课堂讲授以外，必须保证有不少于课堂讲授学时的上机时间。但是学时又有限，不能安排过多的统一上机实验，所以希望学生有效地利用课堂上机实验机会，尽快掌握 C 语

言开发程序的能力,为以后的学习打下一个坚实的基础。

本教材由滕桂法任主编,王芳、何国斌、常淑惠任副主编,参加本教材编写的有:吕显强、张慧英、张隆、杨断利、冯勇等。全书由滕桂法统稿并定稿。

由于编者水平有限,教材中难免有不足或错误之处,敬请广大读者批评指正。

编 者

2006 年 11 月

目 录

C 语言程序设计

前言

Turbo C 2.0 集成开发环境下的使用方法	1
实验 1 熟悉 C 环境	6
实验 2 变量与表达式	9
实验 3 选择结构程序设计	16
实验 4 循环结构程序设计	18
实验 5 数组的使用	24
实验 6 自定义函数 (一)	29
实验 7 自定义函数 (二)	30
实验 8 指针 (一)	35
实验 9 指针 (二)	38
实验 10 结构体与其他构造数据类型	48
实验 11 链表	50
实验 12 文件 (一)	61
实验 13 文件 (二)	63
实验 14 编译预处理与多模块可执行程序生成	69
实验 15 综合实验	75
附录 1 Turbo C 2.0 常用命令	88
附录 2 Turbo C 2.0 常见错误提示信息	89

Turbo C 2.0 集成开发环境下的使用方法

Turbo C 2.0 是一个集成开发环境，它把 C 程序编辑、编译、连接、调试和运行等操作集成在一个软件中，具有使用方便、简单易用的特点。

1. 安装和启动 Turbo C 2.0

运行安装文件 tc2.exe，进入 Turbo C 2.0 安装界面，如图 1 所示。在目标文件夹中可以更改安装文件夹，也可以单击浏览按钮选择安装文件夹。通常，安装文件夹选择“C:\TC”。设置完成后，单击安装按钮就开始安装 Turbo C 2.0。

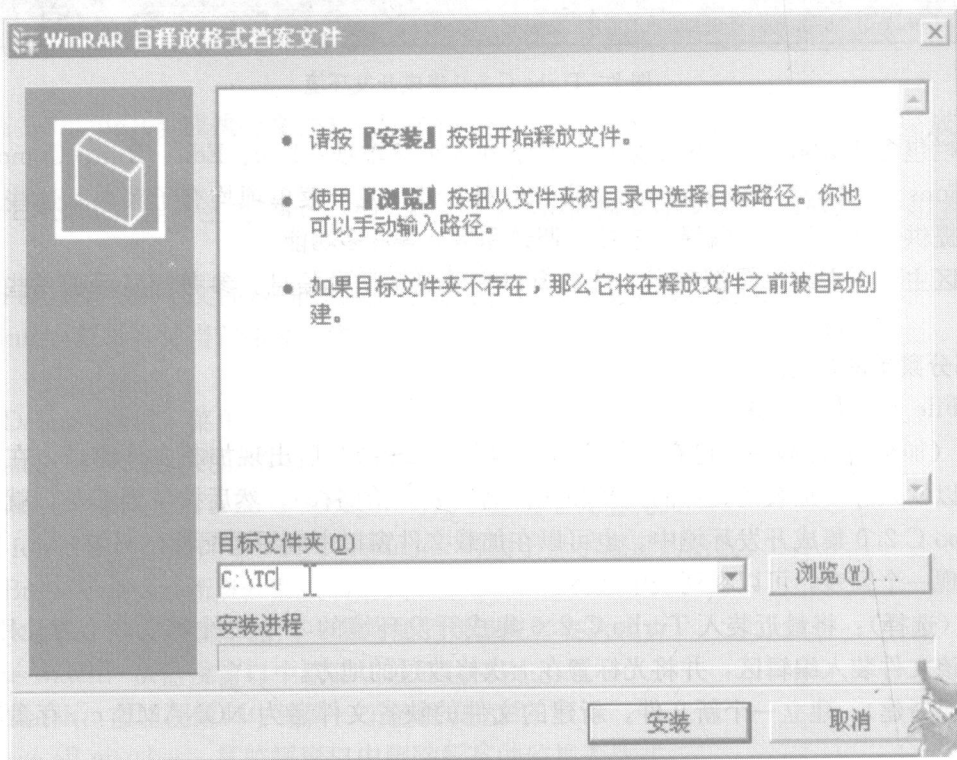


图 1 Turbo C 2.0 的安装界面

运行 C:\TC\TC.EXE 文件就可以启动 Turbo C 2.0。启动后，Turbo C 2.0 的集成开发环境如图 2 所示。

Turbo C 2.0 的集成开发环境中，从上到下依次为：标题栏、菜单栏、编辑区、信息窗口和参考行。

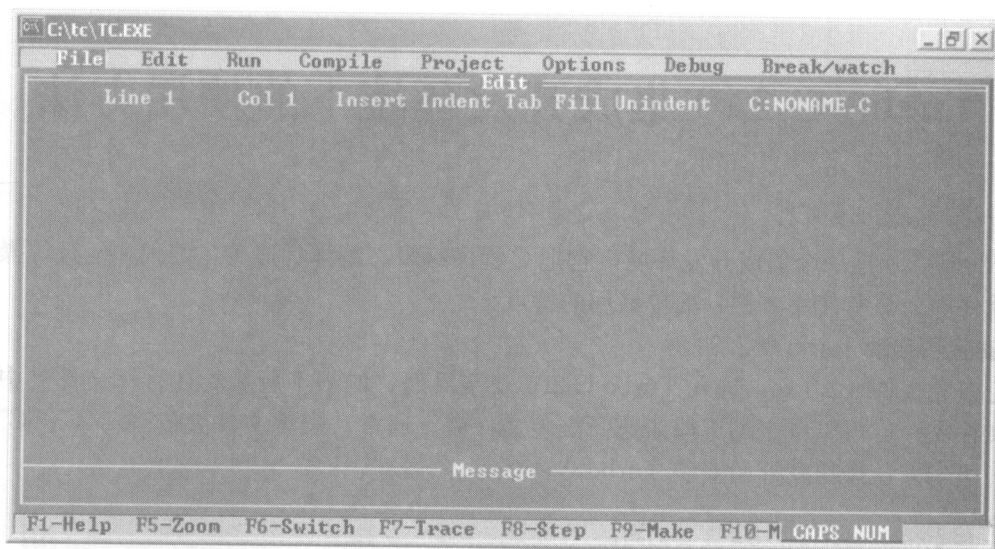


图 2 Turbo C 2.0 集成开发环境

菜单栏包含 Turbo C 2.0 的主菜单, 有下列 8 个菜单项: File、Edit、Run、Compile、Project、Options、Debug、Break/watch。除 Edit 外, 其他几个菜单项均有子菜单。这些主菜单及其子菜单提供了文件管理、编译、连接、调试和运行等许多功能。

编辑区主要用来编写 C 源程序。信息窗口主要显示调试信息。参考行显示了一些常用的功能键。

2. 部分菜单的功能

(1) File (文件) 菜单

Load (加载): 加载一个已存在的文件。执行 Load 命令后出现加载文件窗口, 在加载文件窗口中可以输入想要加载的文件的路径和文件名 (或只给路径), 然后按下回车键, 就可把文件装入 Turbo C 2.0 集成开发环境中。也可以在加载文件窗口中应用通配符, 例如 *.c, 按下回车键后, 出现一个列表, 可以在列表中选择。

Pick (选择): 将最近装入 Turbo C 2.0 集成开发环境的 8 个文件列成一个表让用户选择, 选择后将该文件装入编辑区, 并将光标置在上次修改过的地方。

New (新建): 建立一个新文件。新建的文件的缺省文件名为 NONAME.c, 存盘时可以更改文件名。

Save (存盘): 将编辑区中的文件存盘。如果编辑区中的文件原来没有保存过 (文件名是 NONAME.c), 将询问是否更改文件名。

Write to (换名存盘): 由用户给出文件名, 将编辑区中的文件存盘。如果同名文件已存在, 那么询问要不要覆盖同名文件。

Os shell (暂时退出): 暂时退出。从 Turbo C 2.0 集成开发环境转换到 DOS 提示符下, 此时可以运行 DOS 命令, 若想回到 Turbo C 2.0 集成开发环境中, 只要在 DOS 状态下键入 Exit

即可。

Quit (退出): 退出 Turbo C 2.0 集成开发环境。

(2) Edit (编辑) 菜单

Edit 菜单被执行后, 光标出现在编辑区, 此时用户可以进行文本编辑。

(3) Run (运行) 菜单

Run (运行程序): 运行当前编辑区的文件。如果对上次编译后的源代码未做过修改, 那么直接运行; 否则先进行编译、连接后才运行。

Program reset (程序重启): 中止当前的调试, 释放分配给程序的内存空间。

Go to cursor (运行到光标处): 调试程序时使用。选择该项可使程序运行到光标所在行。光标所在行必须为一条可执行语句, 否则提示错误。

User screen (用户屏幕): 在屏幕上显示程序运行的结果。按任意键后返回 Turbo C 2.0 集成开发环境。

(4) Compile (编译) 菜单

Compile to OBJ (编译生成目标文件): 将一个 C 源文件编译生成 .OBJ 目标文件, 完成后显示报告。

Make EXE file (生成执行文件): 生成一个扩展名为 .EXE 的可执行文件, 完成后显示报告。

Link EXE file (连接生成执行文件): 把当前 .OBJ 目标文件及库文件连接在一起生成可执行文件。

Build all (建立所有文件): 重新编译所有文件, 并生成可执行文件。

Get info: 获得有关当前路径、源文件名、源文件字节大小、编译中的错误数目、可用空间等信息。

(5) Debug (调试) 菜单

Evaluate: 计算表达式的结果。

Call stack: 在运行 Turbo C debugger 时用于检查堆栈情况。

Find function: 在运行 Turbo C debugger 时用于显示规定的函数。

(6) Break/watch (断点及监视表达式) 菜单

Add watch: 向监视窗口插入一个监视的变量或者表达式。

Delete watch: 从监视窗口中删除当前的监视表达式。

Edit watch: 在监视窗口中编辑一个监视表达式。

Remove all watches: 从监视窗口中删除所有的监视表达式。

Toggle breakpoint: 对光标所在的行设置或清除断点。

3. 执行菜单命令的方法

通常, 有三种方法执行菜单命令:

(1) 使用组合键 只要同时按下 Alt 键和主菜单的第一个字母, 就可进入该主菜单。例如, 同时按下 Alt+F 可进入 File 菜单。进入一个主菜单后, 按下子菜单的第一个大写字母, 就可执行该子菜单命令。按下 Esc 键可以从子菜单退回到主菜单或从子菜单的列表中退出。

(2) 使用方向键和回车键 进入一个主菜单后, 可以用上下方向键选择子菜单项, 然后按下回车键, 可执行该子菜单命令。进入一个主菜单后, 也可以用左右方向键选择其他的主菜单。

(3) 使用功能键 Turbo C 2.0 给一些常用的菜单定义了功能键, 在编辑状态下可以直接执行菜单命令。子菜单的功能键显示在子菜单的右边, 如图 3 所示。Load 命令的功能键为 F3, 即按 F3 键即可进入该子菜单, 而不需要先进入 File 菜单再选择这个子菜单。

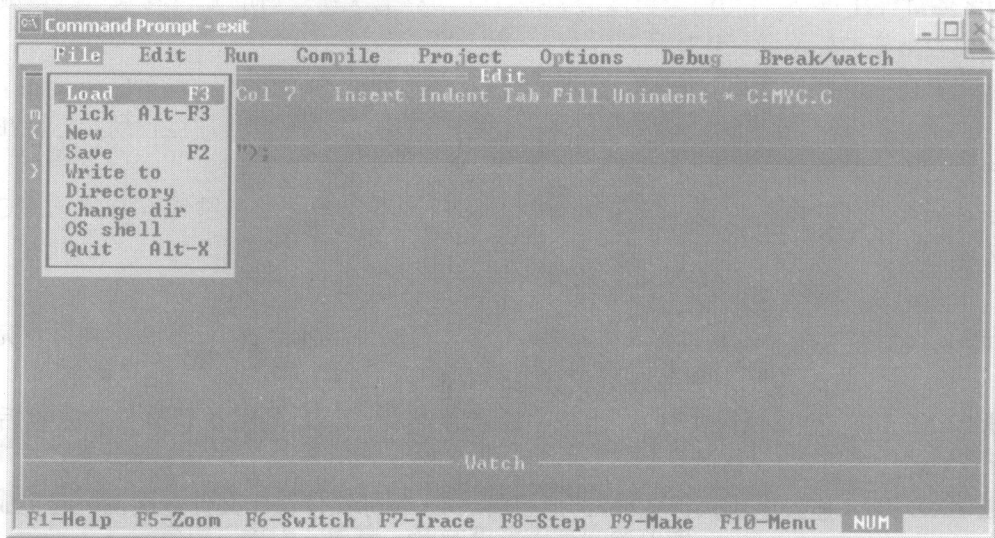


图 3 File 菜单的子菜单

4. 编辑方法

编辑区主要用来编写 C 源程序, 其编辑方法与一些文本编辑软件的使用方法基本相同。一些常用的编辑命令如下:

Ctrl+Y: 删除光标所在的一行。

Ctrl+T: 删除光标所在处的一个词。

Ctrl+KB: 设置块开始。

Ctrl+KK: 设置块结尾。

Ctrl+KV: 块移动。

Ctrl+KC: 块复制。

Ctrl+KY: 块删除。

Ctrl+KH: 块取消。

Ctrl+F1: 如果光标所在位置是 Turbo C 2.0 库函数, 那么获得有关该函数的帮助信息。

Ctrl+OL: 自动缩进开关。

5. 使用 Turbo C 2.0 集成开发环境开发 C 程序的一般步骤

一般地, 使用 Turbo C 2.0 集成开发环境开发 C 程序分为以下几个步骤:

①新建一个文件或者加载一个已存在的文件。

②在编辑区主要用来编辑文件内容。

③文件存盘。C 源程序文件的扩展名为 .c。

④编译、连接、生成可执行文件。

这时, Turbo C 2.0 集成开发环境调用 C 语言编译器, 将源程序生成二进制形式的目标程序。目标程序文件以 .OBJ 为扩展名存入磁盘。再将目标程序与它所用到的库函数文件和其他目标程序连接起来, 生成可执行文件。可执行文件以 .EXE 为扩展名存入磁盘。

⑤运行 .EXE 文件。

⑥查看程序的运行结果。

⑦退出 Turbo C 2.0 集成开发环境。

6. 程序基本调试方法

复杂的程序或者有错误的程序, 都需要进行调试。Turbo C 2.0 集成开发环境提供了一些调试功能。这里介绍基本的调试方法。

将光标移到要观察的变量下, 执行 Break/watch 主菜单下的 Add watch 子菜单, 把变量取到信息窗口中, 执行 Run 主菜单下的 Step over 或者 Trace into 子菜单, 会有一个光条指示当前执行的程序语句, 信息窗口中变量的值也会随之发生变化。单步执行过程 (Step over) 中, 可以随时执行 Run 主菜单下的 User screen 子菜单, 来查看用户屏幕, 也可以将光标定位到一个程序段的末尾, 执行 Run 主菜单下的 Go to cursor 子菜单, 这时程序直接运行到光标处, 可以在信息窗口中查看这段程序运行结束后所选变量的值。

实验 1 熟悉 C 环境

一、实验目的

1. 熟悉 Turbo C 2.0 集成开发环境。
2. 了解 C 程序的基本构成和一般开发过程。

二、知识要点

1. C 语言是一种结构化语言

C 语言以函数为模块组织程序。通常把某一特定的操作编辑成一个函数，调用这个函数就可以进行这个操作。多个函数通过互相调用组成了一个 C 程序。模块化和结构化使得 C 程序层次清晰明了，易于调试和维护。

2. C 程序的组成

一般的，一个 C 程序由以下几个部分组成：预处理命令、自定义数据类型、自定义函数、main 函数和注释。其中，main 函数是主函数。一个 C 程序有且只能有一个 main 函数。C 程序总是从 main 函数开始执行的。

3. printf 和 scanf 函数

printf 函数的作用是向终端输出一个或多个任意类型的数据。它的一般格式是：

printf(格式控制,输出表列)

其中，格式控制是一个字符串，一般包括普通字符、格式说明符和转义字符。

普通字符的作用是输出一些提示性的信息，例如语句“printf("Power = %d\n",iz);”中的“Power =”就是普通字符。普通字符按照原样输出。

格式说明符的作用是把数据按照指定的类型输出，常见的几种格式说明符有：

%d: 以带符号的十进制数形式输出整数。

%f: 以小数形式输出实数。

%c: 以字符形式输出一个字符。

%s: 输出字符串。

经常用到的一个转义字符是“\n”，它的作用是在输出数据之后换行。

输出表列是一个或者多个字符串、表达式或者变量。当输出项多于一个时，输出项之间用逗号分开。

scanf 函数的作用是从键盘输入数据赋给相应的变量。它的一般格式是：

scanf(格式控制,地址表列)

其中, 格式控制与 printf 函数相同。地址表列给出变量的地址。语句 “scanf(“%d%d”, &ix, &iy);” 中的 “&” 就表示取变量的地址, 这个语句的作用是把从键盘输入的第一个数据赋给 ix, 把从键盘输入的第二个数据赋给 iy。

4. 开发 C 程序的步骤

开发一个 C 程序的一般过程包括四个步骤: 编辑、编译、连接和执行。

C 源程序经过编译、连接、最后生成可执行文件, 可执行文件可以在操作系统中运行。

三、实验内容和步骤

1. 熟悉 Turbo C 2.0 集成开发环境。

运行 C:\tc\TC.EXE 文件启动 Turbo C 2.0, 熟悉 Turbo C 2.0 的菜单使用方法和编辑方法。

2. 编写第 1 章例题, 查看运行结果。

- (1) 新建一个文件。
- (2) 在编辑区编辑文件内容。
- (3) 文件存盘。
- (4) 编译、连接、生成 .EXE 文件。
- (5) 运行 .EXE 文件。
- (6) 查看程序的运行结果。
- (7) 返回 Turbo C 2.0 集成开发环境。

3. 编写第 1 章习题第 4 题和第 5 题, 查看运行结果。

四、第 1 章部分习题解答

1. 选择题:

- (1) C (2) C (3) D (4) C

2. 填空题:

- (1) printf("Please input two integers:\n");
- (2) scanf("%d%d\n", &ix, &iy);
- (3) 求和
- (4) printf("The sum is:\n");

4. 编程题:

- (1) 编写一个 C 程序, 显示以下信息:

Happy New Year!

参考程序如下:

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

```
printf("Happy New Year!\n");  
}
```

程序运行结果如下:

```
Happy New Year!
```

(2) 编写一个 C 程序, 输入 3 个整数并且计算它们的和。

参考程序如下:

```
#include <stdio.h>  
void main()  
{  
    int ix,iy,iz,isum;  
    printf("Please input three integers:\n");  
    scanf("%d%d%d",&ix,&iy,&iz);  
    isum = ix + iy + iz;  
    printf("Sum = %d", isum);  
}
```

程序运行后, 输入第一个整数、空格、第二个整数、空格、第三个整数、回车, 输出三个整数之和。

程序运行结果如下:

```
28 36 48  
Sum = 112
```

实验 2 变量与表达式

一、实验目的

1. 了解 C 语言数据类型的含义，并掌握基本数据类型。
2. 初步掌握常量和变量的应用。
3. 掌握基本运算符的优先级、结合性及其应用。
4. 掌握表达式的概念、运算规则及数据类型的转换规则。
5. 掌握基本输入输出函数的格式及其应用。
6. 熟悉常用函数的使用。

二、知识要点

本次实验主要涉及变量与常量的概念、C 语言的基本数据类型、各种运算符和表达式、基本输入输出函数以及一些常用函数。其知识要点包括：

1. 常量的使用

常量在整个程序的运行过程中，其值保持不变。常量包括直接常量和符号常量，直接常量又包括数值常量、字符常量和字符串常量。符号常量应在编译预处理处进行定义，格式如下：

```
#define 符号常量名 常量表达式
```

2. 变量的命名规则及定义

变量是指在程序运行的过程中其值可以变化的量。变量必须先定义后使用。

变量定义时其名称要符合 C 语言标识符的命名规则——由字母、数字和下画线构成，只能由字母或下画线开头，不能使用 C 语言提供的关键字，并且要注意字母的大小写，C 语言大小写敏感，例如：Sum 和 sum 就不是同一个变量名。变量可以在定义的同时赋初值，也可以通过赋值语句进行赋值。变量定义的一般格式为：

```
数据类型 变量名表；
```

3. C 语言的基本数据类型

C 语言有整型、实型、字符型和枚举类型等基本数据。本章主要介绍了前三种基本数据类型：整型、实型和字符型（char）。整型又分为基本整型（int）、短整型（short）和长整型（long），每一种又分有符号（signed）和无符号（unsigned）两种；实型分为单精度型（float）和双精度型（double）。

不同的数据类型所占的存储空间、数据的范围和有效位数均有所不同，读者使用时一定要根