

计算机科学与技术系列丛书

高级程序设计语言C

(下)——实训教程

GAOJI CHENGXU SHEJI YUYAN C (XIA) — SHIXUN JIAOCHENG

■ 李晓燕 主编



教育部直属师范大学
华中师范大学出版社

计算机科学与技术系列丛书

高级程序设计语言 C (下)——实训教程

主 编:李晓燕

副主编:潘雪峰 董国刚

编 者:(以姓氏笔画为序)

李 远 李晓燕 郭 迟

符明丽 曾秀莲 董国刚

廖自能 潘雪峰 魏妮妮

华中师范大学出版社

内 容 简 介

高级程序设计语言 C(下)——实训教程,共分 12 章:第 1 章主要介绍集成软件 Turbo C,第 2 章数据及其类型实训,第 3 章运算符和表达式实训,第 4 章基本语句实训,第 5 章选择语句实训,第 6 章循环语句实训,第 7 章数组实训,第 8 章函数实训,第 9 章结构体与共用体实训,第 10 章指针实训,第 11 章编译预处理实训,第 12 章文件实训。该实训教程包括了丰富的高级程序设计语言 C 上机实例和练习。它与高级程序设计语言 C(上)一起可作为普通高校(民办高校)、二级分校和高职高专计算机专业和非计算机专业的本科和专科教材。

新出图证(鄂)字 10 号

图书在版编目(CIP)数据

高级程序设计语言 C. 下,实训教程/李晓燕主编. —武汉:华中师范大学出版社, 2006. 1

(计算机科学与技术系列)

ISBN 7-5622-3227-X/TP·43

I. 高… II. 李… III. C 语言—程序设计—高等学校: 技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2005)第 058041 号

书 名: 高级程序设计语言 C(下)——实训教程

主 编: 李晓燕

选题策划: 华中师范大学出版社第二编辑室 电话: 027-67867362

出版发行: 华中师范大学出版社©

地 址: 武汉市武昌珞瑜路 100 号 邮编: 430079

销售电话: 027-67867076 67863040 67867371 67861549

邮购电话: 027-67861321 传真: 027-67863291

网址: <http://www.ccnp.com.cn> 电子信箱: hscbs@public.wh.hb.cn

经 销: 新华书店湖北发行所

印 刷 者: 石首市印刷厂

督 印: 姜勇华

责任编辑: 裴媛媛

责任校对: 罗 艺

封面设计: 罗明波

开本/规格: 787 mm×960 mm 1/16 印 张: 13.25 字 数: 250 千字

版次/印次: 2005 年 8 月第 1 版 2006 年 1 月第 2 次印刷

印 数: 3 501—9 500

定 价: 20.60 元

敬告读者: 欢迎举报盗版, 请打举报电话 027-67861321。

本书如有印装质量问题, 可向承印厂调换。

前 言

《高级程序设计语言 C》分上、下两册。上册主要介绍高级程序设计语言 C 的基本概念和基本语句,以及运用这些基本概念和基本语句进行程序设计的基本方法和技巧。下册是实训教程,共分 12 章:第 1 章 Turbo C 基本操作和 C 程序实训;第 2 章数据及其类型实训;第 3 章运算符和表达式实训;第 4 章基本语句实训;第 5 章选择语句实训;第 6 章循环结构实训;第 7 章数组实训;第 8 章函数实训;第 9 章结构体与共用体实训;第 10 章指针实训;第 11 章编译预处理实训;第 12 章文件实训。

其中,第 1 章和第 2 章由廖自能编写,第 3 章和第 4 章由符明丽编写,第 5 章由曾秀莲编写,第 6 章由郭迟编写,第 7 章和第 8 章由李远编写,第 9 章由魏妮妮编写,第 10 章由潘雪峰编写,第 11 章和第 12 章由董国刚编写。全书由主编李晓燕教授统稿和审定。

《高级程序设计语言 C(下)——实训教程》,包括了丰富的上机实例和练习;且每道练习题都给出了参考答案。这些实例和练习,对于提高读者设计程序的能力和水平是十分必要的,通过本教程实训,将为读者参加计算机等级考试和程序员考试打下比较坚实的基础。

本实训教程在编写过程中,得到了作者所在高校和华中师范大学出版社有关领导和同志的支持和帮助,在此一并致谢。

由于作者水平有限,实训教程中可能出现的错误和不妥之处请批评指正。

编 者

2005 年 5 月



目 录

第 1 章 Turbo C 基本操作及 C 程序实训	(1)
1.1 Turbo C 基本操作	(1)
1.2 C 程序实训	(10)
第 2 章 数据及其类型实训	(16)
第 3 章 运算符和表达式实训	(22)
3.1 算术运算符和算术表达式实训	(22)
3.2 赋值运算符和赋值表达式实训	(26)
3.3 关系、逻辑运算符和关系、逻辑表达式实训	(28)
3.4 逗号、条件运算符和逗号、条件表达式实训	(30)
3.5 位运算符及相应表达式实训	(33)
第 4 章 基本语句实训	(36)
4.1 赋值语句实训	(36)
4.2 数据输入输出实训	(37)
第 5 章 选择语句实训	(46)
5.1 if 条件语句实训	(46)
5.2 switch 多分支选择语句实训	(55)
第 6 章 循环结构实训	(62)
第 7 章 数组实训	(92)
7.1 一维数组实训	(92)
7.2 二维数组实训	(96)
第 8 章 函数实训	(102)
8.1 值传递实训	(102)
8.2 地址传递实训	(108)
第 9 章 结构体与共用体实训	(115)
9.1 结构体实训	(115)
9.2 共用体实训	(133)



第 10 章 指针实训	(141)
10.1 指针概念实训	(141)
10.2 数组指针实训	(144)
10.3 字符串指针实训	(152)
10.4 结构体指针实训	(160)
10.5 函数指针及指向函数的指针实训	(165)
10.6 返回指针值的函数实训	(168)
10.7 指针数组和指向指针的指针实训	(175)
第 11 章 编译预处理实训	(182)
第 12 章 文件实训	(192)
参考文献	(205)



第1章 Turbo C 基本操作及 C 程序实训

1.1 Turbo C 基本操作

1.1.1 Turbo C 的安装

C 语言是一种高级语言,用 C 语言编写的源程序是不能直接运行的。而 Turbo C 是一个集 C 语言源程序编辑、编译、连接、运行及调试于一体,用可视菜单驱动的集成软件环境。Turbo C(以下简称 TC)具有方便、直观、易用的界面和丰富的库函数。

1. TC 的安装步骤

(1) 将 TC 的系统盘插进软驱,找到名为 INSTALL 的安装程序。

(2) 双击 INSTALL 图标,并根据提示信息进行相应的操作。

安装成功后用户就可以在硬盘上看到有个 TC 的目录形成,同时在 TC 目录下还有两个子目录“INCLUDE”和“LIB”。

2. TC 的启动

若是在 DOS 命令状态下,先进入 TC 子目录,键入 TC 后按回车键,即: C:\TC>TC↵就进入了 TC 集成开发环境,屏幕显示主菜单和版本信息。

若在 WINDOWS 环境下,找到安装目录下的 TC.EXE,双击即可。此时屏幕上就会出现一个全屏窗口,这时按任意键,相关版本信息消失,出现菜单屏幕,如图 1.1 所示。

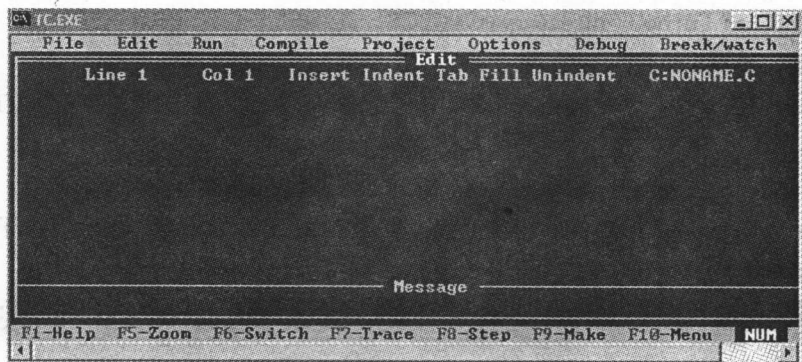


图 1.1 TC 集成环境主屏幕



3. TC 的操作界面

从图 1.1 上可以看到 TC 集成环境的上部有一行主菜单(Main Menu),其中包括下面 8 个菜单项:

File Edit Run Compile Project Options Debug Break/Watch

以上 8 个菜单项分别表示:文件操作、编辑、运行、编译、项目文件、选项、调用、中断/观察。

主菜单的下面为编辑窗口(Edit Window),在该区域内用户可以进行源程序的编辑、修改和调试等工作。

4. TC 的设置

TC 在编译过程中要知道包含文件、库文件、输出文件和 TC 的子目录所在的位置,TC 的设置就是设置它们的目录路径。选中“Options”下拉菜单中的“Directories”选项后按回车键,打开目录设置子窗口,如图 1.2 所示设置目录路径。

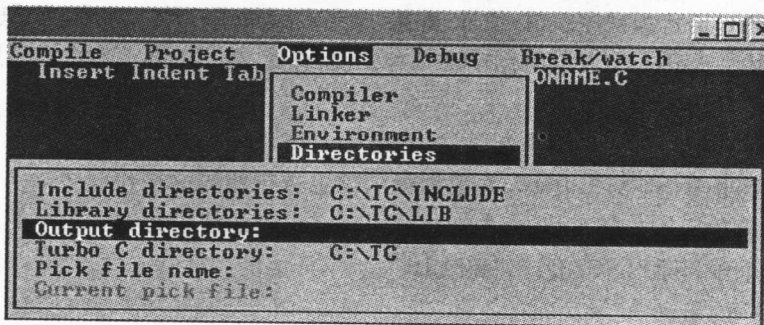


图 1.2 设置目录路径子窗口

操作步骤:

<F10> → “Options” → “Directories” → “Include directories:” → “Include directories:C:\TC\INCLUDE” → “Library directories:” → “Library directories:C:\TC\LIB” → “Output directory:” → “Output directory:C:\TC” → “Turbo C directory:” → “Turbo C directory:C:\TC”

设置完成后,保存配置文件:

<F10> → “Options” → “Save Options” → “Config File:C:\TC\TCCONFIG.TC”

1.1.2 Turbo C 的菜单项的描述

Turbo C 的菜单栏由多个菜单组成,除了 Edit 编辑命令外,其他均含子菜单。

Turbo C 的菜单栏如图 1.3 所示。



图 1.3 TC 菜单栏



1. File 菜单

File 文件菜单主要包含有关对文件进行操作的选项,如图 1.4 所示。各菜单项的描述如表 1.1 所示。

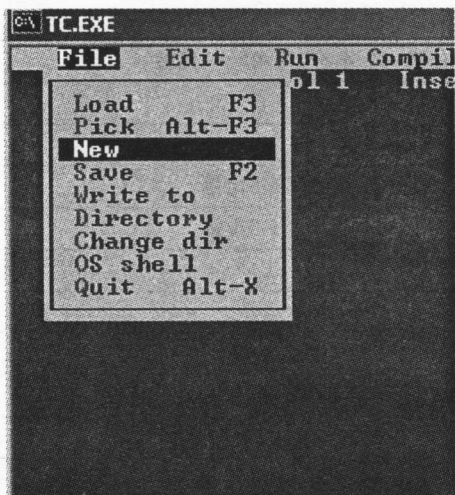


图 1.4 File 菜单

表 1.1 File 菜单项的描述

菜单项	描 述	快捷键
Load	装入已存盘的 C 文件,在主屏幕窗口进行编辑	F3
Pick	将最近装入的 8 个文件列表供用户选择,用户选中其中一个文件打开	Alt-F3
New	编辑一个新文件	
Save	将编辑区中的文本存盘	F2
Write to	写盘,将当前编辑区的文件写到用户指定的文件名下	
Directory	显示目录和一组文件供用户选择	
Change dir	从当前目录转到指定盘符和目录	
OS shell	暂时退出 Turbo C 返回 DOS,键入 EXIT 返回 Turbo C	
Quit	退出 Turbo C 返回 DOS 操作系统	Alt-X

2. Edit 菜单

Edit 编辑菜单项是编辑命令,执行该命令进入编辑窗口,提供文本编辑功能。选中 Edit 后按回车键,进入编辑窗口,这是一个文本编辑程序,它提供四类编辑功能键供用户使用。各菜单项的描述如表 1.2 所示。

表 1.2 Edit 菜单项的描述

光标移动描述			
按 键	描 述		
Ctrl-S	光标左移一个字符,可用“←”代替		
Ctrl-D	光标右移一个字符,可用“→”代替		
Ctrl-A	光标左移一个字		
Ctrl-F	光标右移一个字		
Ctrl-E	光标上移一行,可用“↑”代替		
Ctrl-X	光标下移一行,可用“↓”代替		
Ctrl-W	屏幕上滚		
Ctrl-Z	屏幕下滚		
Ctrl-R	上翻一页,可用 PgUp 代替		
Ctrl-C	下翻一页,可用 PgDn 代替		
块操作描述			
按 键	描 述		
Ctrl-K B	定义块首		
Ctrl-K K	定义块尾		
Ctrl-K T	标出一个单字		
Ctrl-K C	把已定义的块复制到当前光标位置		
Ctrl-K V	把已定义的块移动到当前光标位置		
Ctrl-K Y	删除已定义的块		
Ctrl-K R	从磁盘读文件到当前光标位置		
Ctrl-K W	将已定义的块复制到文件上		
Ctrl-K H	隐藏/显示已定义的块		
Ctrl-K P	打印当前已定义的块		
其他描述			
按 键	描 述	选 项	描 述
Ctrl-QF	查找	B	向后寻找
Ctrl-QA	查找且替换	G	在源程序中寻找
Ctrl-L	重复最后一次查找	L	在块中寻找
Ctrl-U	or Esc	n	寻找 n 次
Ctrl-K	设置标志	N	不询问
Ctrl-Q	寻找标志	u	不区分大小写
		W	寻找一个字

3. Run 运行菜单

Run 运行菜单项能够自动编译、连接并运行当前编译中的程序,如图 1.5 所示。各菜单项的描述如表 1.3 所示。

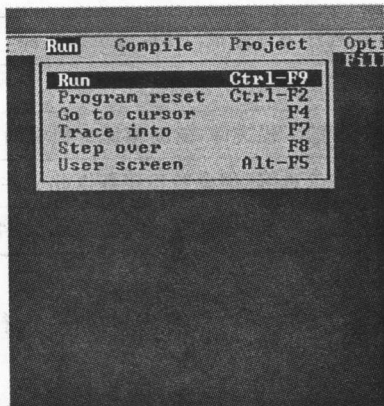


图 1.5 Run 菜单

表 1.3 Run 菜单项的描述

菜单项	描 述	快捷键
Run	运行当前编辑区中的文件或运行指定的主文件	Ctrl-F9
Program reset	重新启动程序,中止当前的调试,释放程序战局的空间	Ctrl-F2
Go to cursor	程序运行到光标处停止	F4
Trace into	运行跟踪至函数的内部	F7
Step over	单步执行	F8
User screen	切换至用户窗口	Alt-F5

4. Compile 菜单

Compile 菜单项编译源程序成为目标文件或可执行文件,如图 1.6 所示。各菜单项的描述如表 1.4 所示。

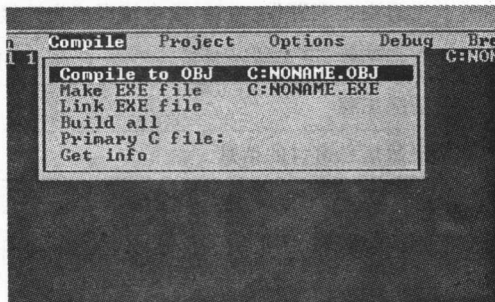


图 1.6 Compile 菜单



表 1.4 Compile 菜单项的描述

菜单项	描 述
Compile to OBJ	将 C 源程序编译生成. OBJ 文件
Make EXE file	将 C 源程序编译生成. EXE 可执行文件
Link EXE file	将 C 目标文件. OBJ 和函数库连接
Build all	构造所有的工程文件, 重新编译工程中的文件, 生成可执行的文件
Primary C file:	设置 C 主文件名, 编译和连接都以此文件名为依据
Get info	获取当前编辑区中文件信息, 如路径、文件名及文件大小等信息

5. Project 工程菜单

Project 工程菜单项指定工程名称, 设置中止程序运行的错误类型, 并清除工程, 如图 1.7 所示。各菜单项的描述如表 1.5 所示。

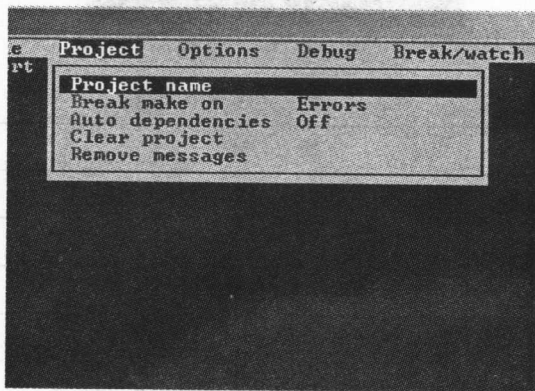


图 1.7 Project 菜单

表 1.5 Project 菜单项的描述

菜单项	描 述
Project name	定义工程名称, 工程名作为. EXE 和. MAP 文件的名称
Break make on Errors	设置在出现警告、错误、致命错误或在连接之前停止编译
Auto dependencies Off	自动关联, 设置编译时是否检查. C 和. OBJ 文件的日期和时间
Clear project	清除工程
Remove messages	清楚信息窗口的信息

6. Options 菜单

Options 菜单项选择编译开关并定义宏, 设置路径、控制开关、保存内容, 如图 1.8 所示。各菜单项的描述如表 1.6 所示。

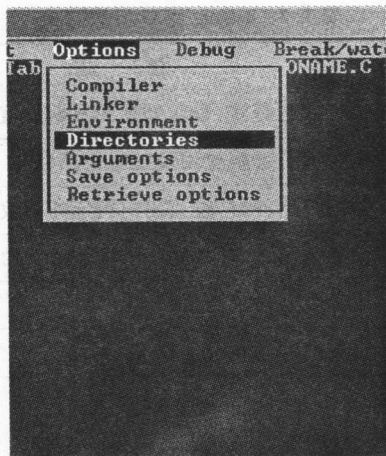


图 1.8 Options 菜单

表 1.6 Options 菜单项的描述

菜单项	描 述
Compiler	包括编译模式、宏定义、代码生成、优化、源代码处理、出错处理和命名
Linker	供用户设置与连接有关选择项
Environment	允许改变 Turbo C 工作环境的设置
Directories	用于设置编译、连接、帮助、输出文件的路径
Arguments	设置命令行参数
Save options	保存配置文件
Retrieve options	装入以前保存过的配置文件

7. Debug 菜单

Debug 菜单项允许跟踪调试,管理错误信息,如图 1.9 所示。各菜单项的描述如表 1.7 所示。

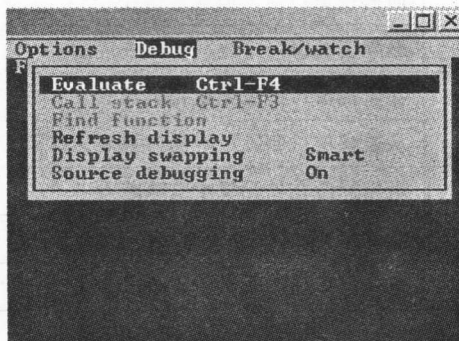


图 1.9 Debug 菜单



表 1.7 Debug 菜单项的描述

菜单项	描 述	快捷键
Evaluate	当程序运行时,查看变量和数据的值	Ctrl-F4
Call stack	调试检查堆栈使用及运行情况	Ctrl-F3
Find function	当程序运行时,查找用户指定的函数	
Refresh display	刷新显示,恢复被输出窗口重写后的编辑窗口	
Display swapping Smart	设置交换显示窗口	
Source debugging On	设置调试资源	

8. Break/watch 菜单

Break/watch 菜单项设置、编辑、删除监视表达式,并设置、清除断点,如图 1.10 所示。各菜单项的描述如表 1.8 所示。

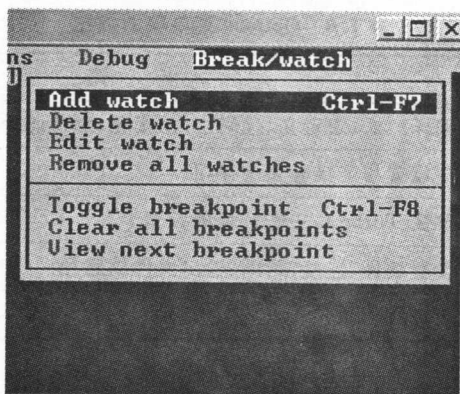


图 1.10 Break/watch 菜单

表 1.8 Break/watch 菜单项的描述

菜单项	描 述	快捷键
Add watch	向监视窗口插入监视表达式	Ctrl-F7
Delete watch	删除监视窗口中当前的监视表达式	
Edit watch	在监视窗口中编辑监视表达式	
Remove all watches	从监视窗口删除所有的监视表达式	
Toggle breakpoint	对光标所在处设置或清除断点	Ctrl-F8
Clear all breakpoints	清除所有的断点	
View next breakpoint	将光标移到下一个断点处	

1.1.3 C 语言源程序的编译、连接

在主菜单中 Compile 的下拉菜单里选择 Make EXE file 项,按回车键(快捷键 Ctrl+F9),则 TC 自动完成对当前编辑窗口中编辑的源程序的编译,连接生成可执行文件. EXE。

若源程序有语法错误,系统将在屏幕中央的“Compiling”窗口底端提示“Errors:Press any key”(错误:按任意键)字样,如图 1.11 所示。

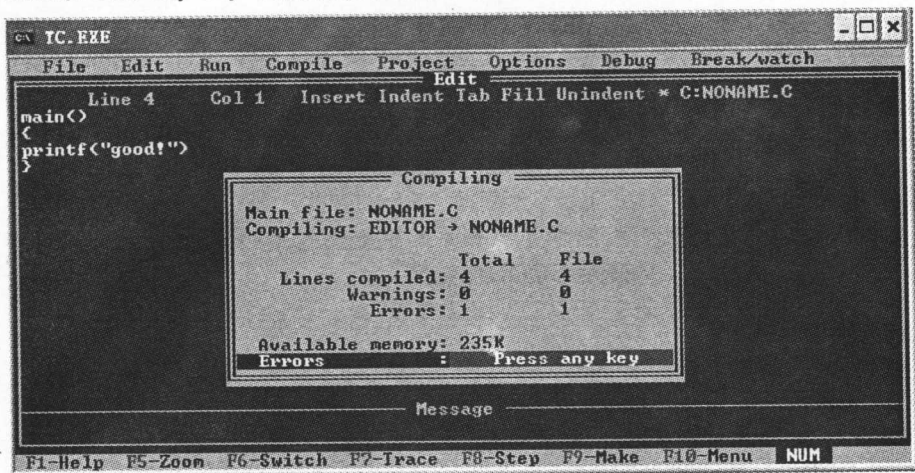


图 1.11 源程序的编译、连接

这时按空格键,屏幕下端的消息中显示出错的信息,并且光带会停在第一条错误消息上,而此时编辑窗口中也有一条光带,再按空格键,光标将会停留在错误信息的位置,这时用户可以逐条对错误位置进行修改,再次编译,连接,直至无误。

1.1.4 运行,查看,编辑新程序

1. 若要运行当前窗口中正在编辑的源程序,可在主菜单中 Run 的下拉菜单里选择 Run 项,按回车键,或快捷键 Ctrl+F9,程序运行完毕后,将再次回到编辑窗口中。

2. 在主菜单中 Run 的下拉菜单里选择 User screen 项,按回车键或快捷键 Alt+F5,查看输出窗口,查看完后,按任意键即可再次回到编辑窗口中,如发现结果不对,可再次对源程序进行修改,修改完毕后重新编译,连接,运行,直到得到正确结果。

3. 在主菜单中 File 的下拉菜单里选择 New 项,按回车键,即可重新出现一个新的编辑窗口,这时用户可以输入新的源程序,重复以上的步骤即可。



1.2 C 程序实训

1.2.1 上机实例

【实例 1】 编写程序,显示以下字符串:

```
*****  
VERY GOOD!  
*****
```

解:程序如下:

```
#include "stdio.h"  
main()  
{printf("*****\n");  
  printf("  VERY GOOD! \n");  
  printf("*****\n");  
}
```

运行结果:

```
*****  
VERY GOOD!  
*****
```

【实例 2】 编程计算 $7+8-2$ 。

解:根据题意,可以采用顺序结构编写程序。程序如下:

```
#include "stdio.h"  
main()  
{int a,b,c,d,s;          /* 定义 5 个整型变量 a,b,c,d,s */  
  a=7;b=8;  
  c=2;                    /* 分别对 a,b,c 赋值 */  
  d=a+b;                  /* 计算 7+8 赋值给 d */  
  s=d-c;  
  printf("%d\n",s);       /* 输出 s 的值 */  
}
```

【实例 3】 随机输入两个整数,输出其最大值。

解:程序如下:

```
#include "stdio.h"  
main()
```



```
{
    int x,y,max;
    scanf("%d,%d",&x,&y);    /* 输入 x,y 的值 */
    if(x>y)
        max=x;
    else
        max=y;
    printf("max=%d\n", max);
}
```

【实例 4】 输入某班 50 名同学的 5 门功课成绩,编程输出每位同学的平均分、最高分和最低分。

解: 分析此题,因数据较大,可以采用二维数组来存储数据,用 for 循环来重复输入所有数据。程序如下:

```
#include "stdio.h"
main()
{
    int i,j;
    float sum,max,min;
    float average,score[50][8];
    for(i=0;i<50;i++)
        for(j=0;j<5;j++)
            scanf("%f",&score[i][j]);
    for(i=0;i<50;i++)
    {
        sum=0;max=min=score[i][0];
        for(j=0;j<5;j++)
        {
            sum+=score[i][j];
            if(score[i][j]>max)
                max=score[i][j];
            if(score[i][j]<min)
                min=score[i][j];
        }
        average=sum/5.0;
        score[i][5]=average;
        score[i][6]=max;
    }
}
```