

C YUYAN
CHENGXUSHEJI
XIANGMUHUA
JIAOCHENG

应用型本科院校“十二五”规划教材



C语言程序设计

项目化教程

主 编 樊为民 唐红雨



南京大学出版社

应用型本科院校“十二五”规划教材

C语言程序设计

项目化教程

主 编 樊为民 唐红雨



南京大学出版社

图书在版编目(CIP)数据

C 语言程序设计项目化教程 / 樊为民, 唐红雨主编.
—南京: 南京大学出版社, 2015. 11
应用型本科院校“十二五”规划教材
ISBN 978-7-305-15975-6

I. ①C… II. ①樊… ②唐… III. ①C 语言—程序设计—高等学院—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 242077 号

出版发行 南京大学出版社
社 址 南京市汉口路 22 号 邮编 210093
出 版 人 金鑫荣

丛 书 名 应用型本科院校“十二五”规划教材
书 名 C 语言程序设计项目化教程
主 编 樊为民 唐红雨
责任编辑 单 宁 编辑热线 025-83596923

照 排 南京理工大学资产经营有限公司
印 刷 江苏凤凰通达印刷有限公司
开 本 787×1092 1/16 印张 16.25 字数 406 千
版 次 2015 年 11 月第 1 版 2015 年 11 月第 1 次印刷
ISBN 978-7-305-15975-6
定 价 36.00 元

网 址: <http://www.njupco.com>
官方微博: <http://weibo.com/njupco>
官方微信号: njupress
销售咨询热线: (025)83594756

* 版权所有, 侵权必究

* 凡购买南大版图书, 如有印装质量问题, 请与所购
图书销售部门联系调换

目 录

第 1 章 C 语言概述	1	3.5 小型案例	44
1.1 任务 1——认识 C 语言程序 及 VC++6.0	1	3.6 小 结	47
1.2 C 语言的发展及特点	1	习 题	48
1.3 初步认识 C 语言程序	3	第 4 章 选择结构程序设计	50
1.4 软件开发方法	6	4.1 任务 4——有节约要求的水 费问题	50
1.5 C 语言程序上机调试步骤和 方法	7	4.2 if 语句	52
1.6 小 结	10	4.3 switch 语句	59
习 题	10	4.4 小型案例	61
第 2 章 C 语言程序设计的初步 知识	12	4.5 小 结	62
2.1 任务 2——关于银行利息的 计算	12	习 题	64
2.2 C 语言的数据类型	14	第 5 章 循环结构程序设计	67
2.3 标识符	14	5.1 任务 5——公司员工薪水 计算	67
2.4 常 量	15	5.2 while 语句	69
2.5 变 量	19	5.3 do-while 语句	70
2.6 运算符与表达式	21	5.4 for 语句	73
2.7 自增(自减)运算符及 C 语言 运算符的优先级别	25	5.5 循环嵌套	76
2.8 不同类型数据间的混合运算 (包含数据类型转换)	27	5.6 goto 语句以及用 goto 语句 构成循环	81
2.9 小 结	28	5.7 break 语句和 continue 语句	82
习 题	28	5.8 小型案例	84
第 3 章 顺序结构程序设计	30	5.9 小 结	88
3.1 任务 3——将英里转换为 公里	30	习 题	89
3.2 C 语句	32	第 6 章 数 组	93
3.3 数据的输出	35	6.1 任务 6——用冒泡法对 10 个数排序	93
3.4 数据的输入	40	6.2 一维数组	95
		6.3 二维数组	97
		6.4 字符数组	102

6.5 小型案例·····	110	9.9 小 结·····	179
6.6 小 结·····	113	习 题 ·····	179
习 题 ·····	113	第 10 章 构造数据类型 ·····	184
第 7 章 函 数 ·····	116	10.1 任务 10——输出一批学生的 基本信息·····	184
7.1 任务 7——从身份证号码中 间提取出生日期、性别、年龄 等信息·····	116	10.2 结构体·····	187
7.2 函数的概念·····	120	10.3 结构体与函数·····	193
7.3 函数的参数和返回值·····	122	10.4 结构体与指针·····	194
7.4 函数的参数传递方式·····	123	10.5 链 表·····	197
7.5 函数的调用·····	127	10.6 共用体·····	199
7.6 变量的作用域和存储 类型·····	129	10.7 枚举类型·····	200
7.7 函数的作用范围·····	132	10.8 typedef 类型声明 ·····	202
7.8 小型案例·····	132	10.9 小型案例·····	203
7.9 小 结·····	134	10.10 小 结 ·····	204
习 题 ·····	135	习 题·····	206
第 8 章 编译预处理 ·····	137	第 11 章 文 件 ·····	208
8.1 任务 8——求解某校园圆形 花坛的圆周长及圆面积·····	137	11.1 任务 11——磁盘文件信息 复制·····	208
8.2 宏定义·····	139	11.2 文件类型指针·····	212
8.3 文件包含命令·····	140	11.3 文件的基本操作·····	212
8.4 条件编译·····	142	11.4 文件的定位函数·····	227
8.5 小型案例·····	144	11.5 文件出错检测函数·····	229
8.6 小 结·····	145	11.6 小型案例·····	231
习 题 ·····	145	11.7 小 结·····	233
第 9 章 指 针 ·····	147	习 题·····	234
9.1 任务 9——利用指针变量 完成税率的调整并打印前 后结果·····	147	第 12 章 位运算 ·····	237
9.2 指针变量·····	149	12.1 位的运算·····	237
9.3 指针与数组·····	157	12.2 位 段·····	241
9.4 指针与字符串·····	164	12.3 小型案例·····	244
9.5 指针与函数·····	168	12.4 小 结·····	247
9.6 指向指针的指针变量·····	174	习 题·····	248
9.7 main()函数的形参和 void 指针·····	175	附 录 ·····	251
9.8 小型案例·····	177	附录 1 C 语言关键字 ·····	251
		附录 2 ·····	252
		参考文献 ·····	255

第 1 章 C 语言概述

C 语言是一种结构化语言。它层次清晰,便于按模块化方式组织程序,易于调试和维护。C 语言的表现能力和处理能力极强。它不仅具有丰富的运算符和数据类型,便于实现各类复杂的数据结构;它还可以直接访问内存的物理地址,进行位(bit)一级的操作。由于 C 语言实现了对硬件的编程操作,因此 C 语言集高级语言和低级语言的功能于一体:既可用于系统软件的开发,也适合于应用软件的开发。此外,C 语言还具有效率高、可移植性强等特点,因此广泛地移植到了各种类型的计算机上,从而形成了多种版本的 C 语言。

1.1 任务 1——认识 C 语言程序及 VC++6.0

问题

什么是计算机的程序?为什么要学习计算机程序设计语言?用 C 语言编写的程序结构是什么样子的,如何使用 C 语言开发环境 VC++6.0 编辑、编译、运行、发布一个 C 语言程序?

完成步骤

- 利用图书馆、网络资源,搜集关于程序、程序设计、编译、编辑、链接、计算机程序设计语言的基本概念。
- 查阅计算机程序设计的一般步骤和常用辅助工具。
- 研究 C 语言程序特点、C 语言程序架构、组成要素。
- VC++6.0 集成开发环境基本设置,菜单、信息出口、编辑窗口的使用。
- 输入并运行本章 1.3 节例程。

1.2 C 语言的发展及特点

1.2.1 C 语言的发展

C 语言的发展颇为有趣。它的原型是 ALGOL 60 语言。

1963 年,剑桥大学将 ALGOL 60 语言发展成为 CPL (Combined Programming Language)语言。

1967 年,剑桥大学的 Martin Richards 对 CPL 语言进行了简化,于是产生了 BCPL 语言。

1970 年,美国贝尔实验室的 Ken Thompson 将 BCPL 进行了修改,并为它起了一个有趣的名字“B 语言”。意思是将 CPL 语言煮干,提炼出它的精华。并且他用 B 语言写了第一个 UNIX 操作系统。

1973 年,B 语言也被人“煮”了一下,美国贝尔实验室的 Dennis M. Ritchie 在 B 语言的基础上最终设计出了一种新的语言,他取了 BCPL 的第二字母作为这种语言的名字,这就是 C 语言。

为了使 UNIX 操作系统推广,1977 年,Dennis M. Ritchie 发表了不依赖于具体机器系统的 C 语言编译文本《可移植的 C 语言编译程序》。

1978 年,Brian W. Kernighan 和 Dennis M. Ritchie 出版了名著《The C Programming Language》,从而使 C 语言成为目前世界上最流行的高级程序设计语言。

1988 年,随着微型计算机的日益普及,出现了许多 C 语言版本。由于没有统一的标准,使得这些 C 语言之间出现了一些不一致的地方。为了改变这种情况,美国国家标准研究所(ANSI)为 C 语言制定了一套 ANSI 标准,成为现行的 C 语言标准 3。C 语言发展迅速,而且成为最受欢迎的语言之一,主要是因为它具有强大的功能。许多著名的系统软件如 DBASE III PLUS、DBASE IV 都是由 C 语言编写的。用 C 语言加上一些汇编语言子程序,就更能显出 C 语言的优势了,如 PC-DOS、WORDSTAR 等就是用这种方法编写的。

1.2.2 C 语言的特点

1. 简洁紧凑、灵活方便

C 语言一共只有 32 个关键字,9 种控制语句,程序书写自由,主要用小写字母表示。它把高级语言的基本结构和语句与低级语言的实用性结合了起来。

2. 运算符丰富

C 语言的运算符包含的范围很广泛,共有 34 个运算符。C 语言把括号、赋值、强制类型转换等都作为运算符处理,从而使 C 语言的运算类型极其丰富,表达式类型多样化,灵活使用各种运算符,可以实现在其他高级语言中难以实现的运算。

3. 数据结构丰富

C 语言的数据类型有整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等,能用来实现各种复杂的数据类型的运算,并引入了指针的概念,使程序效率更高。另外,C 语言具有强大的图形功能,支持多种显示器和驱动器,且计算功能、逻辑判断功能强大。

4. C 语言是结构式语言

结构式语言的显著特点是代码及数据的分隔化,即程序的各个部分除了必要的信息交流外彼此独立。这种结构化方式可使程序层次清晰,便于使用、维护以及调试。C 语言是以函数形式提供给用户的,这些函数可方便地调用,并具有多种循环、条件语句控制程序流向,从而使程序完全结构化。

5. C 语言语法限制不太严格,程序设计自由度大

一般的高级语言语法检查比较严,能够检查出几乎所有的语法错误。而 C 语言允许程

序编写者有较大的自由度。

6. C语言允许直接访问物理地址,可以直接对硬件进行操作

C语言既具有高级语言的功能,又具有低级语言的许多功能,能够像汇编语言一样对位、字节和地址进行操作,而这三者是计算机最基本的工作单元,可以用来写系统软件。

7. C语言程序生成代码质量高,程序执行效率高

一般只比汇编程序生成的目标代码效率低10%~20%。

8. C语言适用范围大,可移植性好

C语言有一个突出的优点就是适合于多种操作系统,如DOS、UNIX,也适用于多种机型。

1.3 初步认识C语言程序

为了帮助同学们对C语言程序形成完整的认识,这里准备了一个较为完整的程序。下面的程序用来演示在C语言程序设计中使用的—般程序设计方法和基本要素。编写程序,用来从文件中读取所有的数据,并输出所读取的数的平方。

```
/*
程序执行时在D盘的根目录下创建一个文件取名为a.dat,
文件中输入10个数"1 3 2 4 5 6 7 8 9 10"
程序运行时输入:Demo01.exe d:\\a.dat
*/
#include <stdio.h>      /* 包含stdio.h,malloc.h,stdlib.h三个头文件 */
#include <malloc.h>
#include <stdlib.h>

#define N 10            /* 定义一个不带参数的宏N */
#define f(x) x*x       /* 定义一个带参数的宏f(x) */

int a[N], *b;          /* 定义全局变量:数组a和指针变量b */
int input();           /* 对函数input作声明 */
void main(int argc,char *argv[]) /* 主函数 */
{
    /* 下面是主函数的函数体 */
    void output();     /* 声明output函数 */
    int n,i;           /* 定义局部变量n,i */
    if(argc!=2)        /* 如果程序运行携带了参数,不携带参数时只有文件名demo01.exe */
    {
        printf("语法:Demo01 <filename>\n"); /* 输出提示信息 */
    }
}
```



```
        return ;          /* 程序运行错误,直接退出程序 */
    }
    n=input(argv[1],a);/* 调用函数 input */

    b=(int *)malloc((sizeof(int) * n));
                                                /* 为指针变量 b 申请内存空间 */
    for(i=0;i<n;i++)      /* 循环 n-1 次 */
    {
        b[i]=f(a[i]);
                                /* 把数组 a 中每个元素都求出平方值,送到数组 b 中对应位置 */
    }
    output(b,n);          /* 调用函数 output,输出所有 b 中元素 */
}
/* 下面函数的功能是把文件中的数据逐个读出,存放到数组 a 中 */
int input(char * filename,int a[])      /* input 函数的首部 */
{
    FILE * fp;                /* 定义指向文件的指针 fp */
    int i=0;                  /* 定义整型变量 i,用来存放读取出来的整数的个数 */
    fp=fopen(filename,"r");    /* 调用系统函数 fopen 打开文件 filename */
    if(fp==NULL)              /* 如果打开文件失败,提示出错并退出程序 */
    {
        printf("\n 无法打开文件 %s",filename);/* 提示错误信息 */
        exit(0);              /* 调用系统函数 exit,退出程序 */
    }
    while(! feof(fp)&& i<N) /* 当文件读取未结束时继续读取 */
    {
        fscanf(fp,"%d",&a[i++]); /* 读取文件 fp 中的一个整数放入数组 */
    }
    fclose(fp);              /* 关闭文件 */
    return (i);              /* 返回读取到得数据的个数 */
}
/* 下面函数的功能是输出数组 c 中的 n 个元素 */
void output(int c[],int n) /* 函数 output 的首部 */
{
    int i;                    /* 定义循环用的整型变量 i */
    for(i=0;i<n;i++)          /* 循环 n 次 */
    {
        printf("%8d",c[i]);    /* 输出数组元素 c[i]的值,保留 8 位宽度 */
        if(i%10==0)           /* 每输出 10 个数据换一行 */

```

```
        printf("\n");  
    }  
}
```

程序输出结果为:

```
Demo01.exe d:\a.dat <回车>  
1 9 4 16 25 36 49 64 81 100
```

1.3.1 C语言程序的格式

C语言中格式书写比较自由,一行可以写多条语句,一条语句也可以书写在多行上。C语言识别大小写字母,如“A”和“a”是两个不同的字母。C语言语句后必须有分号,只有分号的语句为空语句。为了增强程序的可读性,应该避免在一行中书写多条语句,并使用锯齿形书写程序代码,还可以通过空行来增强可读性。

1.3.2 C语言程序的构成

C程序的基本结构是函数,一个或多个C函数组成一个C程序,若干个C语句构成C语言函数,若干个基本单词形成C语句。C语言中使用的函数有两类,一类是系统定义的函数,如printf和fclose等,称为标准库函数,可以直接在程序中使用。另一类是用户自己定义的函数,如demo01.c中的output()函数,必须由用户自己编写源程序代码。

函数的基本格式如下:

```
[函数类型]函数名([函数形参表])    /* 函数首部 */  
{                                     /* 函数体 */  
    [变量定义和声明语句;]  
    可执行语句部分;  
}
```

1.3.3 C语言程序的基本要求

1. 在整个程序文件中,函数可以出现在任意位置。主函数不一定出现在程序的开始处,但不管主函数位于程序何处,程序总是在主函数中开始,也在主函数中结束。

2. 每个程序行中的语句数量任意,既允许一行内写多条语句,也允许多条语句写在同一行上。

3. 为了对程序进行必要的描述,可以给程序进行说明,说明必须写在/*和*/之内。

1.4 软件开发方法

1.4.1 软件开发方法

20 世纪 60 年代出现了软件危机,其现象表现为软件开发费用和精度失控,软件的可靠性差,生产出来的软件难以维护。为了解决软件危机,在 60 年代末期提出了软件工程的概念,并在以后不断发展、完善。与此同时,软件研究人员也在不断探索新的软件开发方法,至今已形成八类软件开发方法,主要有 Parnas 方法、SASA 方法、面向数据结构的软件开发方法、问题分析法、面向对象的软件开发方法、可视化开发方法等。

1.4.2 算法

算法定义:算法是规则的有限集合,是为解决特定问题而规定的一系列操作。

算法的特性:算法必须具备 5 个基本的特性。

1. 有限性:有限步骤之内正常结束,不能形成无限循环。
2. 确定性:算法中间的每个代码行必须有确定含义。
3. 输入:可以有 0 个或者多个输入。
4. 输出:至少有一个输出。
5. 可行性:原则上能够精确运行,操作可以经过已经实现的基本运算通过有限次执行完成。

算法的表示方法:算法的表示方法有多种,下面介绍算法的流程图表示方法。传统的流程图由下面的几种基本框组成。使用这些框和流程线组成的流程图表示算法,形象直观,简单方便。在设计算法的时候对于整理设计思路很有帮助。

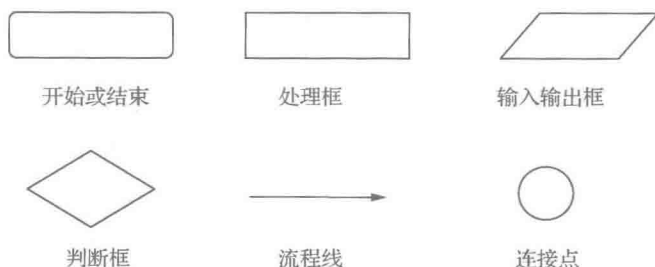


图 1-1 流程图的基本框图

1.4.3 结构化程序设计

结构化程序设计的思想最早是由著名计算机科学家 E. W. Dijkstra 提出的。1966 年,Bohm 和 Jacopin 证明了只用三种基本结构就能实现任何一个入口、一个出口的程序;1977

年,IBM公司的Mills又进一步提出:“程序应该只有一个入口和一个出口。”在长期程序设计的实践中,结构化程序设计方法不断得以完善,使之成为开发传统应用领域应用系统的主要方法之一。

结构化程序设计由三种基本结构组成,分别是:顺序结构、选择结构和循环结构。

1. 顺序结构

顺序结构是最简单的一种结构,可以由赋值语句和输入、输出语句构成。当执行由这些语句构成的程序时,将按照这些语句在程序中的先后顺序逐条执行。流程图如图1-2(a)所示。

2. 选择结构

选择结构也称分支结构。当执行该结构中的语句时,程序根据不同的条件执行不同分支中的语句。如图1-2(b)所示。

3. 循环结构

循环结构是指根据各自的条件,使同一组语句重复执行多次或者一次也不执行。循环结构有两种形式:当型循环和直到型循环。当型循环中的循环体可能一次都不执行,而直到型循环中的循环体至少执行一次。分别如图1-2(c)和(d)所示。

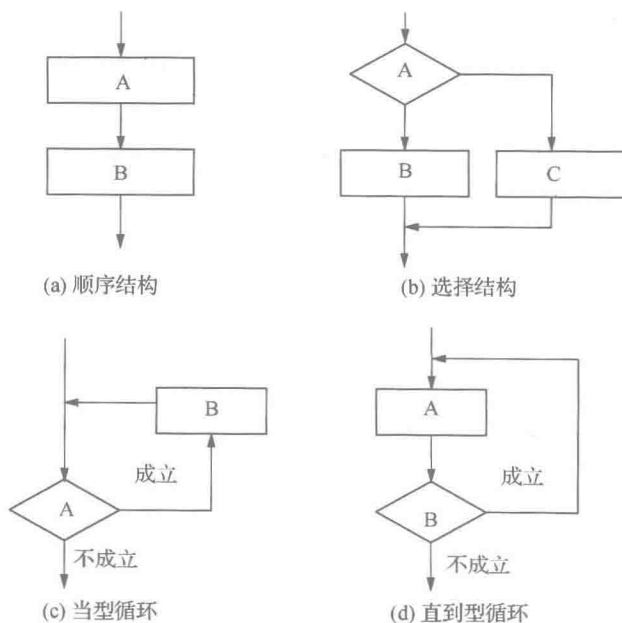


图 1-2 程序的基本结构框图

1.5 C语言程序上机调试步骤和方法

本教材使用 Microsoft Visual C++ 6.0 作为开发工具,下面对于如何创建应用程序作个说明,步骤如下:

(1) 启动 Microsoft Visual C++ 6.0。

(2) 点击菜单上的【File】|【New】出现如图 1-3 所示的新建对话框,选择【Projects】下的【Win32 Console Application】选项,在【Project name】中输入项目名称,这里输入 Demo001,点【Location】右边的浏览按钮选择文件存放的位置。然后点 OK 按钮。

(3) 接下来点 Finish 按钮,在接着出现的对话框上点 OK 按钮,就创建好了一个空项目,如图 1-4。

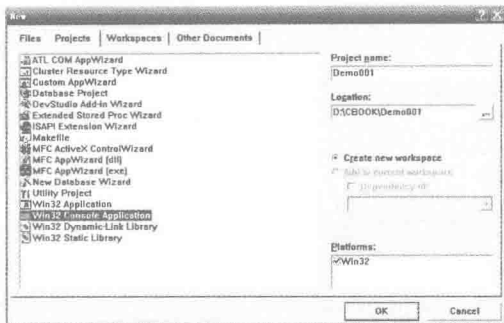


图 1-3 新建对话框

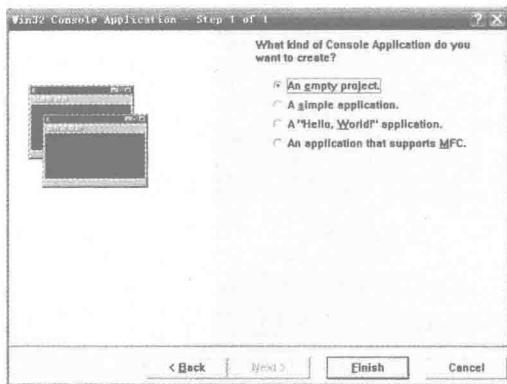


图 1-4 新建对话框结束

(4) 然后添加源代码文件。打开【New】对话框,在【Files】选项卡下选择 C++ Source File 选项,在【File】下面的文本框中输入源文件的名称,这里输入 Code01.C,然后点 OK 按钮。切记,这里输入 Code01.C 文件名时必须输入文件扩展名。如图 1-5。

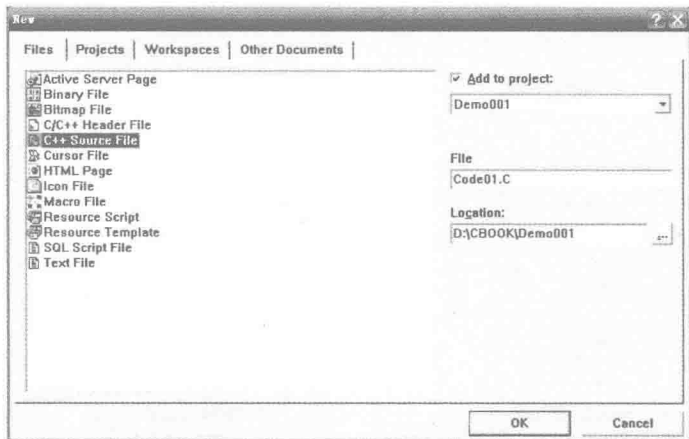


图 1-5 新建 C 源代码文件

(5) 在右边的空白区输入下面的源程序代码。

```
#include "stdio.h"
int max(int x, int y)
{
    return x>y? x:y;
}
```

```
main()
{
    int a,b;
    int maxValue;
    printf("请输入两个整数:");
    scanf("%d %d",&a,&b);
    maxValue=max(a,b);
    printf("\n%d,%d 中较大值为 %d\n",a,b,maxValue);
}
```

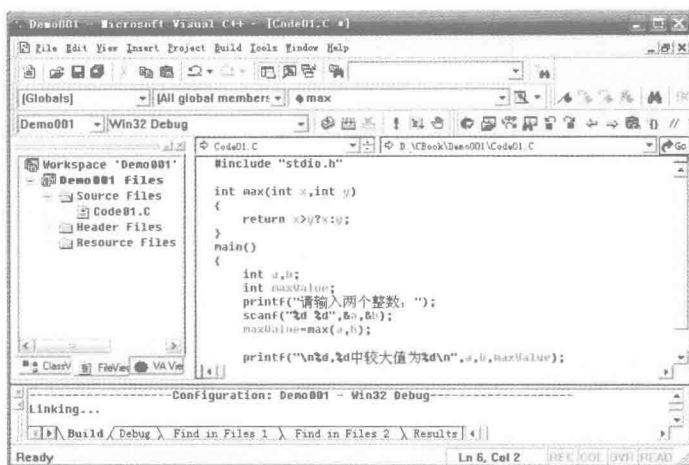


图 1-6 输入源代码

(6) 源代码输入结束后点【Build】菜单上的【Compiled Code01.C】菜单项,如果源代码有语法错误,下面的【Build】框会提示错误出现的位置。根据错误修改源代码,直到错误数为0。如图1-7。

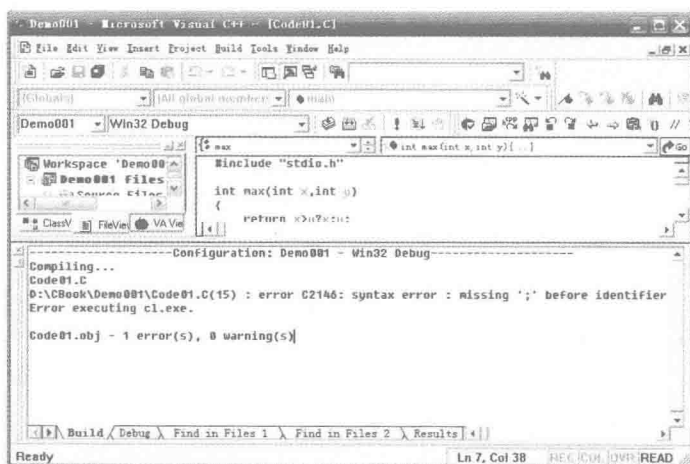


图 1-7 Build 窗口

(7) 最后点【Build】菜单上的【Build Demo001. exe】菜单项,生成应用程序。最后点击【Execute Demo001. exe】菜单项运行程序。根据屏幕提示运行程序,得到程序运行结果如下:

```
请输入两个整数:3 5 <回车>
3,5 中较大值为 5
```

1.6 小 结

本章概括介绍了 C 语言的发展及特点,通过实例分析讨论了 C 程序的格式、构成及基本要求、程序设计的基本知识,最后介绍了 C 语言上机调试的步骤和方法。需要学习的要点有:

1. C 语言是一种兼有汇编语言和高级语言特点的语言,于 20 世纪 70 年代初期由贝尔实验室研制开发。

2. C 语言是一种理想的结构化语言。

(1) 具有结构化的控制语句。

(2) 语言简洁,结构紧凑,使用方便灵活。

(3) 运算丰富,数据处理能力强。

(4) 可以直接访问物理地址,实现硬件和底层系统软件的访问。

(5) 语言生成的代码质量高。

(6) 可移植性好。

3. 函数是 C 语言程序的基本单位。一个 C 语言源程序可以由多个函数组成,其中有且只有一个名为 main 的主函数,无论 main 函数在程序的什么位置,C 语言程序总是从 main 函数开始执行。

4. 用 C 语言编写的程序称为 C 语言源程序,它必须经过编译和连接,生成可执行程序后才能执行。

习 题

一、选择题

1. 下面的说法不正确的是()。

A. 一个 C 语言程序的执行总是从该程序的 main 函数开始,在 main 函数中结束

B. main 函数必须写在一个 C 语言程序的最前面

C. 一个 C 语言程序可以包含若干个函数,但是只能有一个主函数(main)

D. C 语言程序的注释可以是中文文字信息

2. 下面说法不正确的是()。

A. C 语言是一种高级语言

B. C 语言的文件扩展名是.C

- C. 顺序结构、选择结构和循环结构之外再没有其他的程序结构
- D. 算法可以没有输出

二、填空题

1. C语言程序是由_____构成的,一个C程序中至少包含一个_____。因此,_____是C语言程序的基本单位。
2. C语言程序注释是由_____和_____所界定的文字信息组成的。
3. 函数体一般包括_____和_____。
4. C语言是一种_____化程序设计语言。

三、程序题

1. 编写程序在控制台输出下面的图案。

```
* * * * *
*      Hello China      *
* * * * *
```


第2章 C语言程序设计的初步知识

构成C语言程序的两个主要因素是数据和操作。计算机中用来处理问题的是程序,其主要操作对象是数据。当程序执行后,操作的结果又会改变数据。数据类型就是对程序所处理的数据按照其性质、表达方式、构造特点、存储范围等划分的不同种类。

本章主要介绍C语言程序中的数据类型、常量、变量、运算符、表达式、不同类型数据间的运算等内容。

2.1 任务2——关于银行利息的计算

问题

假设银行定期存款的年利率为5.65%,定期存款金额为m元,编写一个C语言程序,计算n年定期存款到期后本金和利息总和s元。分别用单精度类型和双精度类型变量完成计算,观察计算精度的变化,思考在编写程序解决实际问题时变量类型的选取。

分析

计算银行定期存款利息时,如果给定存款的本金数量m元、定期年限n年和年利率v,就可以根据银行计算本息的公式: $s=m * (1+v)^n$ 。

数据需求

数据输入

v /* 年利率 */

m /* 存款本金总金额 */

n /* 存款年限 */

数据输出

s /* 存款到期后本息总额 */

算法描述

实现银行定期存款本息计算的算法表示如图2-1所示