

● 21世纪高等院校计算机应用规划教材

C语言

程序设计实训教程

主 编 杨丽萍 刘粉香 孙勤红



南京大学出版社

21世纪高等院校计算机应用规划教材

C语言

程序设计实训教程

主 编 杨丽萍 刘粉香 孙勤红



南京大学出版社

图书在版编目(CIP)数据

C 语言程序设计实训教程 / 杨丽萍, 刘粉香, 孙勤红
主编. — 南京: 南京大学出版社, 2015. 8
21 世纪高等院校计算机应用规划教材
ISBN 978-7-305-15763-9

I. ①C… II. ①杨… ②刘… ③孙… 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(2015)第 188972 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 21 世纪高等院校计算机应用规划教材
书 名 C 语言程序设计实训教程
主 编 杨丽萍 刘粉香 孙勤红
责任编辑 刘群烨 吴 汀 编辑热线 025-83686531
照 排 南京南琳图文制作有限公司
印 刷 常州市武进第三印刷有限公司
开 本 787×1092 1/16 印张 11.75 字数 301 千
版 次 2015 年 8 月第 1 版 2015 年 8 月第 1 次印刷
ISBN 978-7-305-15763-9
定 价 27.00 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

前 言

C 语言是一门实践性强的结构化程序设计语言。学习和掌握 C 语言离不开上机练习。目前,随着高校本科教育转型的快速发展,应用型院校也迫切需要实践性强的 C 语言实训教程。

本书结合作者十多年讲授 C 语言程序设计课程的教学经验编写而成的。全书共分 10 个部分,主要内容包括:C 语言概述、数据类型、运算符、表达式、顺序结构、选择结构、循环结构、数组、函数、指针、结构体、共用体、枚举和文件等。

本书内容编排合理、循序渐进、深入浅出、讲解通俗,对重点和难点做了详细的阐述,并辅以大量的实例,各章还配有较实用的案例、真题习题。本书注重实践性并有以下特点:

(1) 适合初学者学习,通俗易懂。本书针对初学者的特点,从读者所熟悉的数学知识或其他常识知识入手,深入浅出地叙述,循序渐进地展开,尽可能通俗地讲解 C 语言各章的知识和算法。

(2) 实践、实用性强。书中示例丰富、各章末尾设置了较实用的案例,习题难易适中,覆盖面广,既包括理解语法的示例,又有增加兴趣和提高编程能力的案例。强调了编程能力训练的重要性。

(3) 本书各章末尾增加了等级考试真题,有等级考试需求的同学可以做针对性的训练。

本书适用于计算机专业和非计算机专业“C 语言程序设计”课程的实训教程,也可作为 C 语言培训和计算机等级考试辅导的教学用书,还可供自学者参考。

本书由杨丽萍、孙勤红、刘粉香等具有丰富教学经验的老师编写。孙勤红编写了实训一、实训四、实训五,杨丽萍编写了实训二、实训三、实训七、实训八、实训十,刘粉香编写了实训六、实训九。

在本书的编写过程中得到了许多老师的帮助,在此表示诚挚的谢意。由于作者水平有限,编写教材的经验不足,书中难免有疏漏和不当之处,敬请专家、同仁和读者批评指正。

作者
2015 年 6 月

目 录

实训一 初步认识 VC++ 6.0	1
1.1 Visual C++ 6.0 基本操作	1
1.2 新建、保存、修改、运行程序	2
1.3 调试器的使用	6
1.4 上机练习	8
实训二 数据类型、运算符和表达式	9
2.1 上机实验	9
2.2 上机练习	13
2.3 真题练习	13
实训三 输入输出函数,顺序结构	17
3.1 上机实验	17
3.2 上机练习	23
3.3 真题练习	24
实训四 选择结构	30
4.1 上机实验	30
4.2 上机练习	34
4.3 真题练习	36
实训五 循环结构	45
5.1 上机实验	45
5.2 上机练习	56
5.3 真题练习	57
实训六 数 组	71
6.1 上机实验	71
6.2 上机练习	77
6.3 真题练习	81

实训七 函 数	89
7.1 上机实验	89
7.2 上机练习	97
7.3 真题练习	98
实训八 指 针	112
8.1 上机实验	112
8.2 上机练习	124
8.3 真题练习	125
实训九 结构体与共用体	140
9.1 上机实验	140
9.2 上机练习	154
9.3 真题练习	157
实训十 文 件	165
10.1 上机实验	165
10.2 上机练习	172
10.3 真题练习	173
真题练习答案	178

实训一 初步认识 VC ++ 6.0

1.1 Visual C ++ 6.0 基本操作

1.1.1 启动和退出 Visual C ++ 6.0

1. 启动 VC ++ 6.0 的方法与步骤

方法一:双击桌面 VC ++ 6.0 的快捷图标。



方法二:桌面左下角【开始】菜单→【所有程序】→【Microsoft Visual C ++ 6.0】→单击 Microsoft Visual C ++ 6.0 即可打开。

方法三:双击打开一个 C 程序文件,随之打开 VC ++ 6.0。

通常第一次启动 VC ++ 6.0 后,出现的界面如图 1-1 所示。

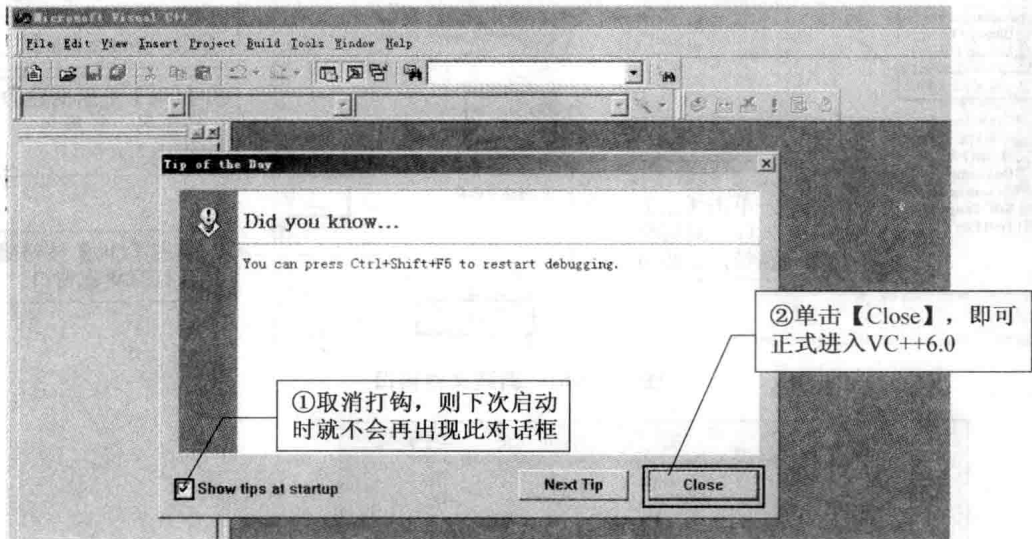


图 1-1 启动界面

2. 退出 VC ++ 6.0 的方法与步骤

方法一:单击窗口右上角的【×】。

方法二:使用快捷键【Alt】+【F4】。

方法三:选择菜单栏【File】→【Exit】。

1.2 新建、保存、修改、运行程序

在 F 盘下建立以自己姓名命名的文件夹,创建的 C 程序文件均存放在该文件夹里。

【例 1-1】 编写 C 语言程序 1-1.c,使其在屏幕上显示一句话“This is a C program.”。

操作步骤:

(1) 打开 VC++ 6.0,新建文件 1-1.c,操作步骤如图 1-2(a)~(c)所示。

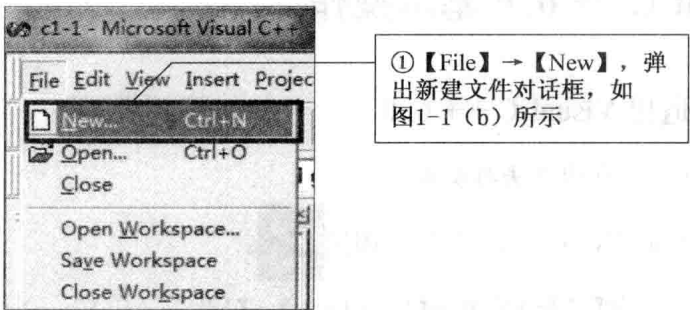


图 1-2(a) 新建文件菜单

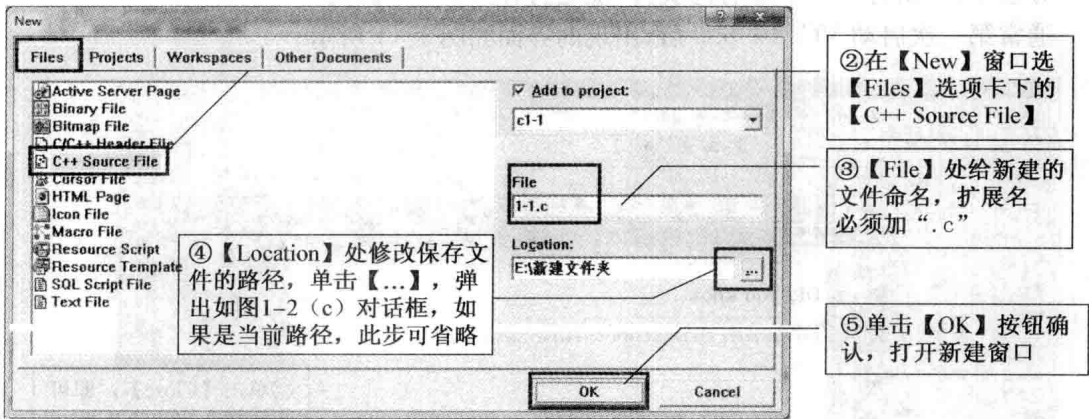


图 1-2(b) 新建文件窗口

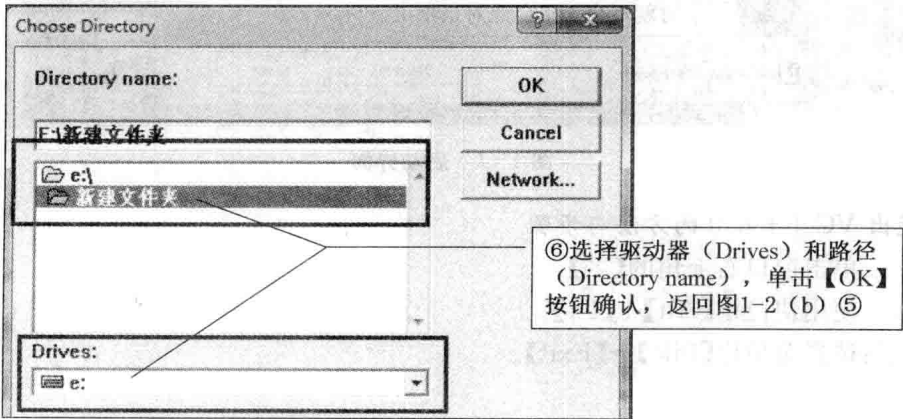


图 1-2(c) 选择保存路径

(2) 在新建窗口中,输入 C 语言代码并运行,如图 1-2(d)所示。

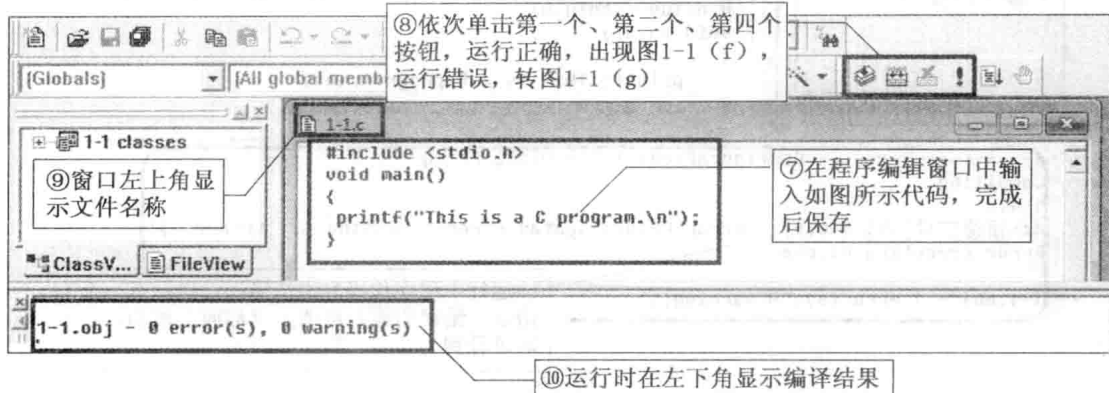


图 1-2(d) 输入代码运行

注意:

- ① 调试运行程序,也可使用菜单【Build】下的【Compile】和【Build】等命令实现。
- ② 在 Win 7 操作系统下首次运行可能会弹出一个窗口,处理方法见图 1-2(e)。



图 1-2(e) 程序兼容性处理

(3) 运行结果显示,正确执行结果如图 1-2(f)所示,如果没有出现图 1-2(f),请转至错误处理方法,参见图 1-2(g)。

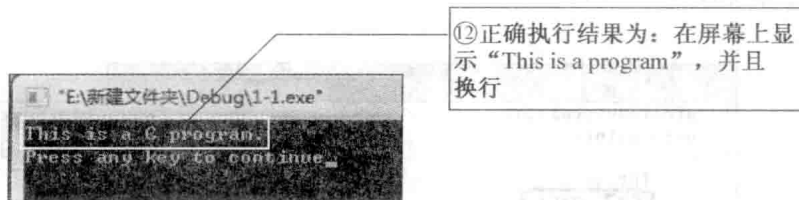


图 1-2(f) 正确执行结果

查看结果后按任意键返回编辑窗口,继续编辑或关闭。“Press any to continue”即为按任意键继续,任一程序运行时最后均会显示这一句话。

如果编写程序中有语法错误,必须将错误修改后方可运行。

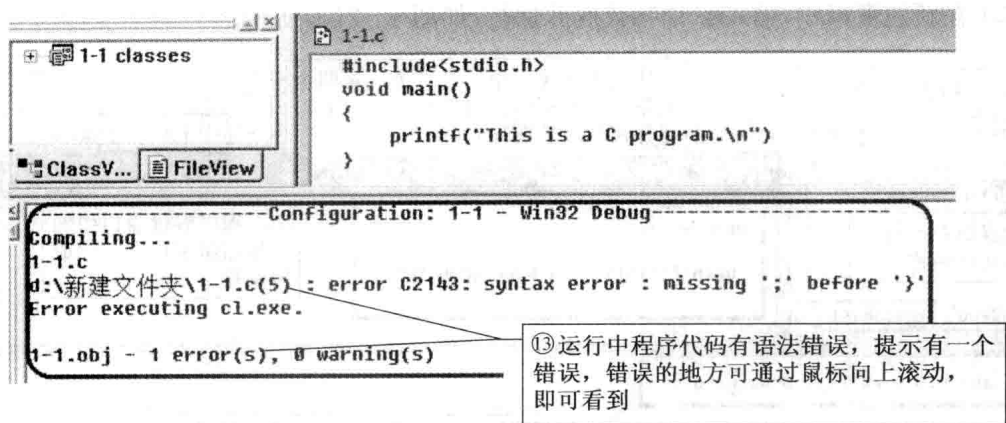


图 1-2(g) 有错误的情况处理

注意：

- ① VC++ 6.0 提示的错误不一定正确，有时可能由于一个错误而提示有多处错误，修改错误的能力随着 C 语言课程的学习以及程序的编程能力提高。
- ② 在 VC++ 6.0 中，源程序中使用的括弧、分号、反斜杠等符号均为英文的半角符号，除非是普通字符(后面学习中会讲解)，所以编写程序时一定要注意符号状态的切换。
- ③ 在运行程序过程中，产生很多同名文件，如图 1-2(h)所示，其中最为重要的为源程序(扩展名为.c)，其他文件为运行时产生的一系列编译、链接等文件。

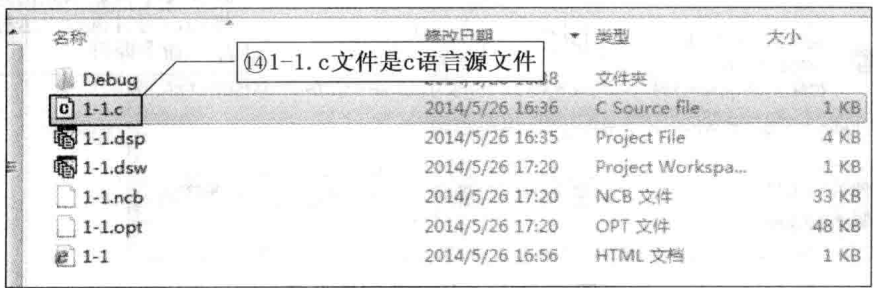


图 1-2(h) 运行程序产生的文档

【例 1-2】 打开已有文件 1-2.c，请填空实现 3+5 的值，并输出结果，操作步骤见图 1-3(a)，运行结果见图 1-3(b)。

操作步骤：

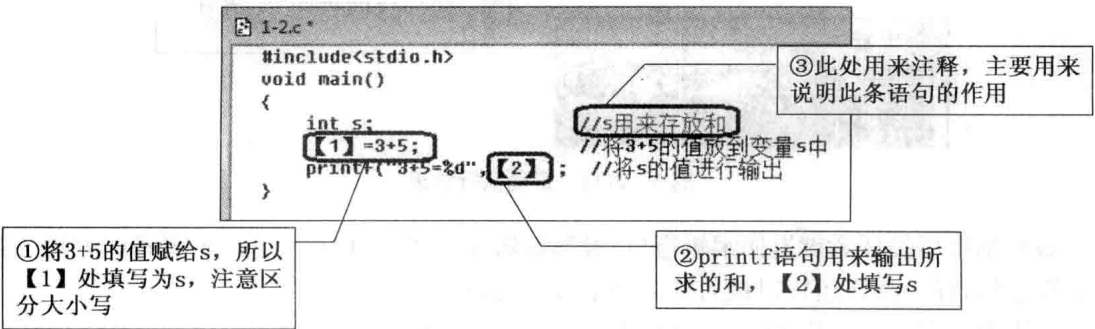


图 1-3(a) 修改程序

运行结果如图 1-3(b)所示。“3+5=”是普通字符,原样输出,“8”是变量 s 的值,随后出现的“Press any key to continue”在程序运行结束后均会出现。

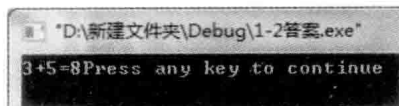


图 1-3(b) 例 1-2 运行结果

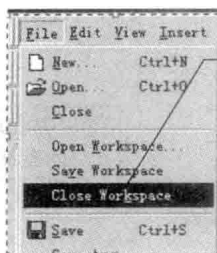
思考:

为什么例 1-1 题显示结果时换行,而本题显示结果时却直接在结果后面显示没有换行,根本原因在于“\n”。

【例 1-3】 新建文件 1-3.c,使用 do...while 语句求出 1~100 的和。

操作步骤:

(1) 完成【例 1-2】后,如果没有关闭 VC++ 6.0,需要关闭工作空间,操作方法如图 1-4 所示。



①首先需要关闭工作空间【Close Workspace】,否则运行下一个程序时就会出错

图 1-4 新建菜单

(2) 新建 1-3.c(步骤参照【例 1-1】),输入下列文本。

```
#include <stdio.h>

void main(void)
{
    int i, sum;
    i=1;
    sum=0;
    while(i <= 100)
    {
        sum += i;
        i++;
    }
    printf("%d\n", sum);
}
```

(3) 依次编译、链接、运行,最后在屏幕显示 5050。如果编写程序有错误,请改正错误后再次运行。

1.3 调试器的使用

调试器可以设置断点、逐步调试代码,还可以监视变量和内存,查看每一步执行时变量的值的变化,是在无法找到程序的逻辑错误时有效解决问题的一种工具。在学习 C 语言的过程中,如果程序能够运行,但是结果错误,而查看代码又看不出错误在何处时,可以使用调试器。

【例 1-4】 打开 1-3.c,需要完成:① 语句 i=1 处设置断点;② 取消 i=1 处的断点;③ 在 sum=0 处设置断点;④ 进入【Debug】窗口后单步执行,监视 i 和 sum 值的变化。

操作步骤:

(1) 将光标定位在 i=1 语句处,右击鼠标,出现快捷菜单,选【Insert/Remove Breakpoint】,

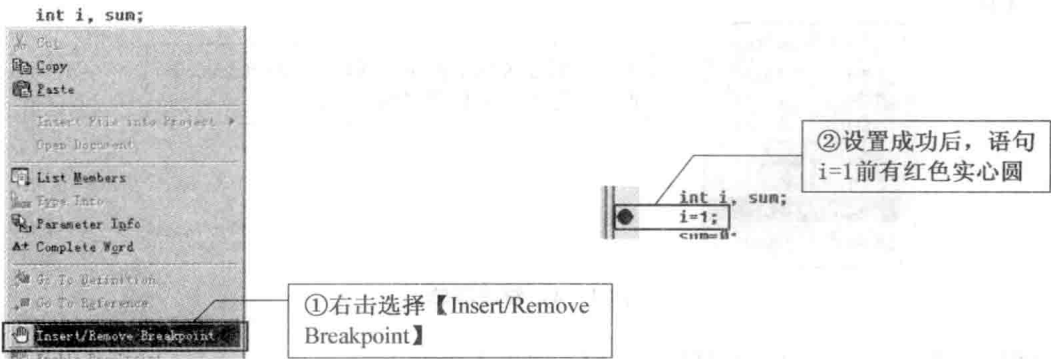


图 1-5(a) 设置断点

(2) 在语句 i=1 处再次执行【Insert/Remove Breakpoint】命令,则执行的是 Remove Breakpoint,已有断点取消。将光标定位于语句 sum=0,使用同样的方法设置断点。

(3) 进入【Debug】窗口,依次选择【Build】→【Start Debug】→【Go】,

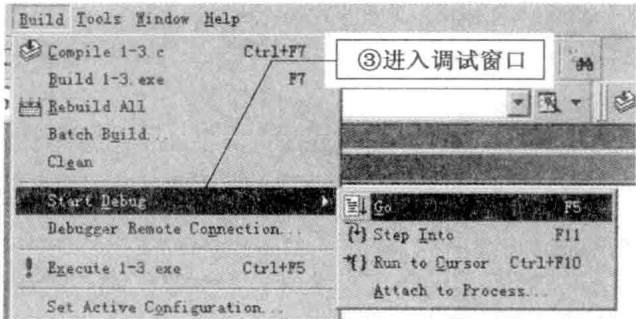


图 1-5(b) 进入 Dubeg 窗口

(4) 进入【Debug】窗口后,菜单由【Build】变为【Debug】,界面如图 1-5(c)所示。

(5) 连续不断地按【F10】键,单步执行程序,查看程序变量的值是否和预想中的一样。图 1-5(d)为 i=19,sum=190 的情况。

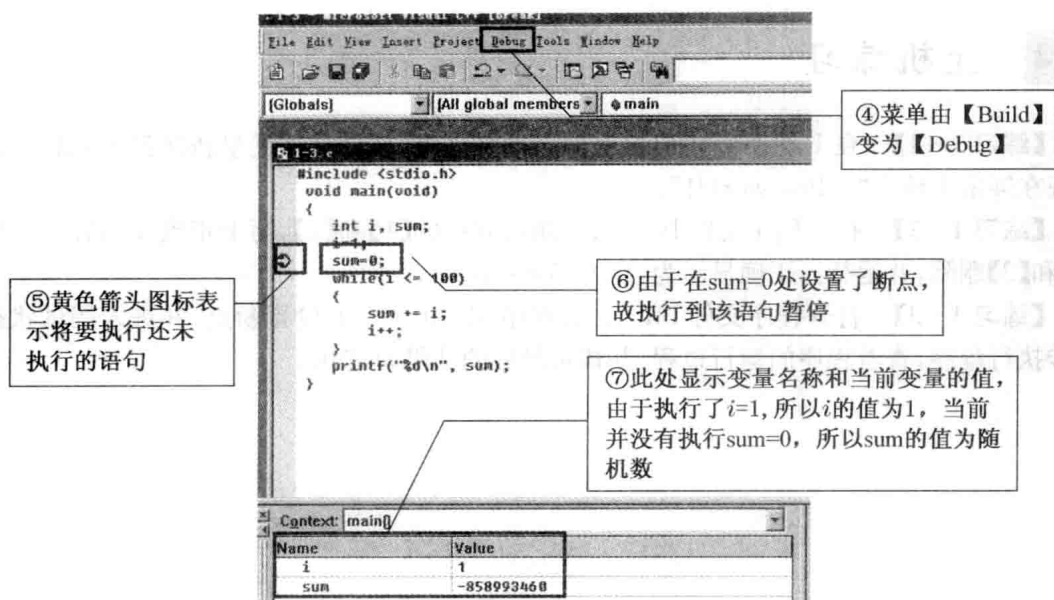


图 1-5(c) 初次 Debug 窗口

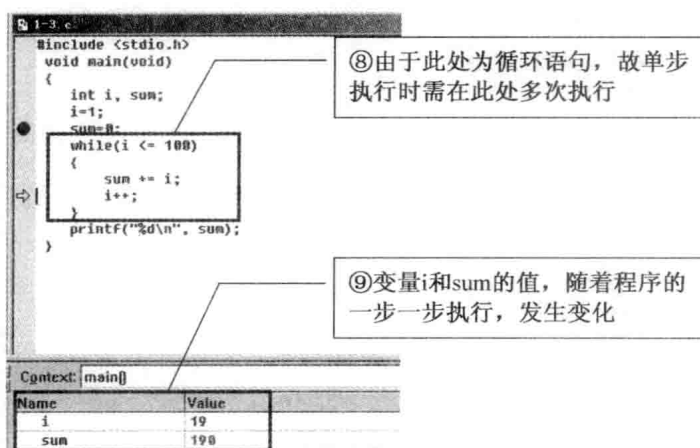


图 1-5(d) 中间执行的一个结果

如此执行下去，一直到 $i=101$ ，不再满足循环条件 $i \leq 100$ 退出循环，继续执行，完毕。

(6) 如果想退出 Debug 窗口，执行【Debug】→【Go】，或者再程序编辑框中右击选【Go】，即可恢复到 Build 界面。

说明：

本内容为学习 C 语言的第一次课，对于程序的变量定义和使用、循环的概念不了解是正常的。主要通过调试方法告诉读者，在后面的学习过程中，如何有效使用调试窗口，解决编程中出现的逻辑错误，清楚知道程序执行时执行的路径和各变量的变化情况，是否和预期一样，如果不一样，查找错误原因，修正错误。

1.4 上机练习

【练习 1-1】 在 F 盘下新建以名字命名的文件夹,在该文件夹里新建程序 lx1-1.c,实现在屏幕上输出“hello, world!”。

【练习 1-2】 打开程序文件 lx1-2.c,填写其中的**【1】**和**【2】**,写上正确地内容,并且把**【1】**和**【2】**删除,并运行。正确显示为:100-50=50。

【练习 1-3】 打开程序文件 lx1-3.c,在语句 sum=0 处设置断点,并进入调试状态,单步执行程序,查看程序的运行过程,并核对最后的结果为 2500。

实训二 数据类型、运算符和表达式

2.1 上机实验

2.1.1 数据类型

【例 2-1】 编写 C 语言程序 2-1.c, 求各基本类型所占的字节数。

```
#include<stdio.h> //包含命令,使用 printf 函数必须使用此命令
```

```
int main()
```

```
{
```

```
    printf("int:    %d\n", sizeof(int)); //输出函数
```

```
    printf("short:  %d\n", sizeof(short));
```

```
    printf("long:   %d\n", sizeof(long));
```

```
    printf("char:   %d\n", sizeof(char));
```

```
    printf("float:  %d\n", sizeof(float));
```

```
    printf("double: %d\n", sizeof(double));
```

```
    return 0; //返回给操作系统
```

```
}
```

计算类型所占字节数

运行结果: int: 4
 short: 2
 long: 4
 char: 1
 float: 4
 double: 8

【例 2-2】 编写 C 语言程序 2-2.c, 求各基本类型所占的字节数。

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    int a=3;
```

```
    short b=4;
```

```
    long c=12345;
```

```
    char d='a';
```

```
    float x=2.1;
```

```
    double y=3.4;
```

```
    printf("int:    %d\n", sizeof(a));
```

```
    printf("short:  %d\n", sizeof(b));
```

定义整型变量 a 并赋值 3

计算 a 变量所占内存字节数

```

printf("long:      %d\n", sizeof(c));
printf("char:      %d\n", sizeof(d));
printf("float:     %d\n", sizeof(x));
printf("double:    %d\n", sizeof(y));
return 0;
}

```

```

运行结果: int:      4
          short:    2
          long:     4
          char:     1
          float:    4
          double:   8

```

【分析】 编译时产生警告错误:

实训二\2-2.c(8) : warning C4305: 'initializing': truncation from 'const double' to 'float'.

提示第 8 行 `float x=2.1`; 产生初始化警告。"2.1" 在 C 语言中当作 `double` 型数处理的, 而 `x` 是 `float` 型变量, "=" 两端类型不一致, 要进行转换。这类错误可以忽略, 也可以改正。将 `float x=2.1` 改为 `float x=2.1f`。

【总结】 在 VC++ 6.0 编译器下, `int` 型数据占 4 字节, `short` 型数据占 2 字节, `long` 型数据占 4 字节, `char` 型数据占 1 字节, `float` 占 4 字节, `double` 占 8 字节。

2.1.2 常量变量

【例 2-3】 先写出下面程序的运行结果, 再编写程序 2-3.c 运行, 验证运算结果。

```

#include<stdio.h>

int main()
{
    printf("\101 \x42 C\n");
    printf("I say: \"How are you? \"\n");
    printf("\\C Program\\n");
    printf("Turbo 'C'");
    return 0;
}

```

运行结果: A B C

```

I say: "How are you?"
\C Program\
Turbo 'C'

```

【总结】 `'\ddd'` 表示八进制数字代表的字符, `'\xhh'` 十六进制代表的字符, `'\"'` 表示 `'\"'` 字符

【例 2-4】 编写程序 2-4.c, 编译链接运行。

```

#include<stdio.h>

int main()

```

```
{ int a,b,sum;
    printf("sum=%d\n",sum);
    return 0;
}
```

运行结果:sum=-858993460

【分析】 编译时产生警告错误:

实训二\2-4.c(3): warning C4101: 'a': unreferenced local variable /* 未引用局部变量 a */

实训二\2-4.c(3): warning C4101: 'b': unreferenced local variable /* 未引用局部变量 b */

实训二\2-4.c(4): warning C4700: local variable 'sum' used without having been initialized /* 局部变量 sum 没有初始化就被使用了 */

运行时产生上述结果,原因是 a,b 变量没有被赋值就使用了,sum 得到随机值并输出了。

【总结】 变量使用三部曲:先定义,后赋值,再使用。

2.1.2 表达式

【例 2-5】 编写程序 2-5.c,计算并输出一个三位整数的个位、十位、百位数字之和。例如:123 的个位为 3,十位 2,百位为 1,数字之和为 6。

【算法】 设存放数的变量为 x,个位、十位、百位数的变量分别为 b0,b1,b2。

```
b0=x%10;
b1=x/10%10;
b2=x/10/10;
#include <stdio.h>
int main()
{
    int x=123, b0, b1, b2, sum;
    b2 = x / 100;
    b1 = (x - b2 * 100) / 10;
    b0 = x % 10;
    sum = b2 + b1 + b0;
    printf("b2=%d, b1=%d, b0=%d, sum=%d\n", b2, b1, b0, sum);
    return 0;
}
```

运行结果:b2=1, b1=2, b0=3, sum=6

【总结】 学会分离整数的各个数位。后续课中常用的算法。

【例 2-6】 已知 a=2, b=3, x=3.5, y=2.5,编写程序 2-6.c,求 (float)(a+b)/2 + (int)(x) % (int)(y) 的值。

【提示】 根据变量的值可知 a,b 变量定义为整型(int),x,y 变量定义为实型(float)。

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