

21世纪高职高专教学做一体化规划教材

C语言

程序设计教学做一体化

- ★ 问题导入
- ★ 案例驱动
- ★ 理论够用
- ★ 实践丰富

主 编 滕泓虬

副主编 何受倩 邹 月 陈文兰



中国水利水电出版社

www.waterpub.com.cn

内 容 提 要

本书结合高职高专学生的培养特点,突出高职高专以技能培养为目标的特色,从实用性角度出发,较全面地介绍了C语言程序设计所涉及的知识。理论知识以必需、够用为度,重在培养编程能力,力求从实际应用的角度出发,尽量减少枯燥、实用性不强的理论概念,加强应用性和实操性强的内容。全书通过案例驱动的方式,把本门课所需的知识点融合到案例中,提高学生的兴趣和学习主动性。

全书分为11个模块,内容包括简单程序基本组成和程序调试过程,常用的数据类型、运算符和表达式,顺序程序设计方法和常用的输入、输出函数,选择程序设计方法,循环程序设计方法,数组和字符串,函数和预处理,指针,结构体,文件操作,应用程序设计实例等。

本书可作为高职高专计算机类、电子信息类各专业的教材,也可作为相关技术人员的自学参考用书。

本书电子教案读者可以从中国水利水电出版社网站和万水书苑免费下载,网址为:
<http://www.waterpub.com.cn/softdown/>和 <http://www.wsbookshow.com>。

图书在版编目(CIP)数据

C语言程序设计教学做一体化 / 滕泓虬主编. — 北京:中国水利水电出版社, 2010.8
21世纪高职高专教学做一体化规划教材
ISBN 978-7-5084-7646-9

I. ①C… II. ①滕… III. ①
C语言—程序设计—高等学校:技术学校—教学参考资料
IV. ①TP312

中国版本图书馆CIP数据核字(2010)第121435号

策划编辑:陈宏华 责任编辑:李 炎 加工编辑:李 皓 封面设计:李 佳

书 名	21世纪高职高专教学做一体化规划教材 C语言程序设计教学做一体化
作 者	主 编 滕泓虬 副主编 何受倩 邹 月 陈文兰
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路1号D座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 68367658 (营销中心)、82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京蓝空印刷厂
规 格	184mm×260mm 16开本 14印张 345千字
版 次	2010年8月第1版 2010年8月第1次印刷
印 数	0001—3000册
定 价	25.00元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换

版权所有·侵权必究

前 言

C 语言是高职计算机类、电子信息类专业的一门专业基础课,课程的教学目标是培养学生以 C 语言为编程语言进行模块化程序设计的逻辑思维能力,掌握程序设计的基本语法规则和程序调试方法,并具有初步的分析问题和解决问题的能力。

本教材编写具有以下特点:

(1) 结合高职高专学生的培养特点,突出高职高专以技能培养为目标的特色,重在编程能力培养,理论知识以必需、够用为度。力求从实际应用的角度出发,尽量减少枯燥、实用性不强的理论概念,加强了应用性和实操性强的内容。

(2) 按模块化来组织教材内容,全书分为 11 个模块,各专业可根据本专业教学要求来选择不同的模块进行教学。通过各模块的教学和实训,提高学生的程序设计、分析和调试能力。

(3) 采用“案例驱动”的组织方式编写,每一模块都是以案例作为引入点,避免大段的理论叙述,这样就有利于激发学生的兴趣,避免因内容枯燥而使学生学习热情不高的情况。每一模块所需了解的知识点,都贯穿到各案例中,通过案例的分析,更有利于对知识点的掌握。

(4) 充分体现“教、学、做”一体化的教学理念,各模块内容全部都融合到各精选案例中,在案例分析后,还安排了随堂练习,使对知识点的掌握马上得到巩固。本书设计了“通讯录管理系统”作为综合设计实例,通过综合设计,有助于串连各模块所学知识,提高综合的程序设计能力。

(5) 配备了大量的课后练习以及上机实训题目,可以作为课堂练习、课后作业和上机实训内容。

全书共分为 11 个模块,模块一:简单程序基本组成和程序调试过程,模块二:常用的数据类型、运算符和表达式,模块三:顺序程序设计方法和常用的输入、输出函数,模块四:选择程序设计方法,模块五:循环程序设计方法,模块六:数组和字符串,模块七:函数和预处理,模块八:指针,模块九:结构体,模块十:文件操作,模块十一:应用程序设计实例。

本书由广东科贸职业学院的滕泓虬任主编,何受倩、邹月、陈文兰任副主编,苏瑞娟、黄静、刘红霞参编,其中滕泓虬编写了模块二、模块八、模块十、模块十一,何受倩编写了模块三、模块五、模块六、模块七,苏瑞娟编写了模块九,黄静编写了模块四,邹月编写了模块一,全书由滕泓虬和邹月统稿。

本书可作为高职高专、培训机构、成人高校及民办高校的计算机类、电子信息类各专业的教材,也可作为有关技术人员的自学参考用书。

由于编者水平有限,书中的疏漏和错误之处在所难免,恳请阅读本书的读者提出宝贵意见。

编者

2010 年 5 月

目 录

前言

模块一 初识 C 语言	1	3.3 字符输入函数与字符输出函数	43
1.1 简单的 C 程序	1	3.3.1 putchar 函数	44
1.2 C 程序的上机过程	4	3.3.2 getchar 函数	44
课后练习	8	3.3.3 字符串输出 puts 函数	45
上机实训	9	3.3.4 字符串输入 gets 函数	46
模块二 数据类型、运算符和表达式	11	课后练习	46
2.1 数据类型	11	上机实训	47
2.2 常量和变量	12	模块四 选择结构程序设计	49
2.2.1 常量	12	4.1 关系运算符与关系表达式	49
2.2.2 变量	13	4.1.1 关系运算符	49
2.3 整型数据	15	4.1.2 关系表达式	50
2.3.1 整型常量	15	4.2 逻辑运算符与逻辑表达式	51
2.3.2 整型变量	16	4.3 if 条件语句	53
2.4 实型数据	19	4.3.1 if 语句的简单形式	53
2.4.1 实型常量	19	4.3.2 if 语句的基本形式	55
2.4.2 实型变量	20	4.3.3 if-else if 阶梯形式	56
2.5 字符型数据	21	4.3.4 if-else 的嵌套形式	58
2.5.1 字符常量	21	4.4 switch 语句	59
2.5.2 字符变量	23	课后练习	63
2.6 运算符和表达式	24	上机实训	65
2.6.1 算术运算符和算术表达式	25	模块五 循环结构程序设计	67
2.6.2 自增和自减运算符	27	5.1 while 循环语句	67
2.6.3 赋值运算符和赋值表达式	28	5.2 do-while 循环语句	68
2.6.4 条件运算符和条件表达式	30	5.3 for 语句	70
2.6.5 表达式中数据类型的转换	30	5.4 循环语句的嵌套	72
课后练习	32	5.5 break 语句和 continue 语句	75
上机实训	33	5.5.1 break 语句	76
模块三 顺序结构程序设计	35	5.5.2 continue 语句	76
3.1 程序的三种基本结构	35	课后练习	77
3.2 格式输入函数与格式输出函数	38	上机实训	79
3.2.1 格式输入函数	38	模块六 数组	81
3.2.2 格式输出函数	40	6.1 一维数组	81

模块十一 应用程序设计综合实例	190	附录 B C 语言中的运算符及其优先级	201
11.1 程序设计方法简介	190	附录 C C 语言常用语法摘要	203
11.2 通讯录管理系统	192	附录 D Turbo C 常用库函数	207
11.3 程序设计题目	199	附录 E Turbo C 2.0 编译错误信息	214
附录 A 常用字符与 ASCII 代码 (十进制) 对照表	200		

模块一 初识 C 语言

通过学习 C 语言，既可以用来编写应用软件和系统软件，又可以培养计算机编程基本思想、编程基本技能及逻辑思维能力。

1.1 简单的 C 程序



学习目标

- 了解 C 语言程序的结构特点。
- 掌握库函数的调用方法。

案例 1-1

在计算机屏幕上输出 “Hello,my C world!”

◇ 案例程序

```
main()                                /*主函数*/
{
    printf("Hello, my C world!\n");    /*输出要显示的内容*/
}
```

运行这个程序，在屏幕上显示一行英文：

Hello,my C.world!

◇ 案例分析

(1) 这是一个最简单的 C 程序，学习 C 语言就由此开始吧！

(2) C 程序是由函数组成的，函数就是相对独立的可以完成特定功能的程序段。本程序就是由一个称为 main 的函数构成的。其中 main 是函数的名字，函数名后都要有一对圆括号，用来写函数参数的，本程序的 main 函数没有参数所以不写，但圆括号不能省略。

(3) 一个完整的程序必须有一个 main 函数，称为主函数，程序总是从 main 函数开始执行，也就是说，main 函数是程序的入口。

(4) main()后面花括号内的部分称为函数体。本程序中的函数体只有一个语句。一般情况下函数体由“说明部分”和“执行部分”组成。说明部分的作用是定义数据类型；执行部分是操作命令。本实例只有执行部分而没有说明部分。

(5) C 程序的每一条基本语句都以“;”结束。

(6) 可以用 /* ... */ 对 C 程序的任何部分作注释，用来解释该条语句或该段程序的含义或作用，这样可以帮助程序阅读者更好地理解程序中有关部分内容。

(7) printf 是 C 的库函数中的一个函数，它的作用是在显示屏上输出指定的内容，此例输出 “Hello,my C world!” 字符串。Printf("Hello,my C world! \n")括号中的内容就是调用 printf

库函数时的参数，通俗地讲就是“通知”printf 要显示的内容。参数中的“\n”是 C 语言中的格式控制符，作用是：在输出“Hello,my C world!”之后执行一个回车换行操作，如果以后还有输出的话，将从下一行的左端开始输出。

◇ 随堂练习

编写一个 C 程序，输出以下信息：

TurboC 2.0

案例 1-2

求从键盘输入的任意两个整数相加的和并将结果显示出来。

◇ 案例程序

```
#include<stdio.h>
main()
{
    int a, b, sum;          /*定义三个整型变量*/
    printf("Input one number: "); /*调用 printf 库函数输出信息在屏幕上*/
    scanf("%d",&a);        /*调用 scanf 库函数执行由键盘输入数据并赋值给加数 a*/
    printf("Input the other number:"); /*提示信息*/
    scanf("%d",&b);        /*由键盘输入数据赋值给被加数 b*/
    sum=a+b;               /*求 a 和 b 两数之和*/
    printf("sum=%d \n", sum); /*输出 sum（和）的值在屏幕上*/
}
```

程序运行结果如下：

```
Input one number:5 ✓
Input the other number:10✓
sum=15
```

◇ 案例分析

(1) 首先应该考虑程序中要用到的数据的个数，本例中要用到三个数，即加数、被加数、和，所以先定义三个变量。语句 `int a, b, sum;` 的作用就是定义三个变量，名字分别为 `a`、`b`、`sum`，类型都是整型。`int` 表示整型，是 C 语言的关键字（见知识链接）。当程序运行的时候，系统将根据定义在内存中为这三个变量分配存储单元。输入的数就存放在这三个单元中，如图 1-1 所示。可以这样说，变量的作用就是用来保存和标识数据，访问数据就可以通过变量名来访问。

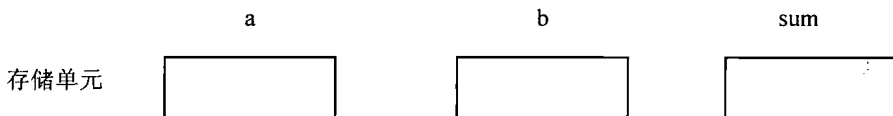


图 1-1

(2) 库函数调用 `scanf("%d",&a);` 用来实现从键盘接收用户输入的数据，并送到变量 `a` 所对应的地址单元中。这也意味着，变量 `a` 的值就是输入的值。例如从键盘输入 5，那么变量 `a` 的值就是 5。`&` 表示取地址（也称为取地址运算符），“`%d`”用来限制输入数据的格式是整型。

(3) `sum=a+b`;表示将 `a` 和 `b` 两个变量单元中的内容相加,并将结果存放在 `sum` 所对应的存储单元中。

(4) 库函数调用 `printf("sum=%d\n", sum)`;用于输出结果到屏幕上。注意第一个“`sum`”,包括“`=`”,都表示字符串常量的一部分,应该按原样输出,即在屏幕上显示 `sum=`。“`%d`”用来限制输出的格式是整型,括号右端的 `sum` 是要输出的变量名字,执行该函数时,就是把 `sum` 变量的值按照整型格式输出在 `%d` 的位置,如图 1-2 所示。所以最后输出: `sum=15`。`%d` 称为格式控制符,也可以称为占位符。

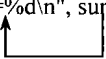
`printf("sum=%d\n", sum);`

用 `sum` 的值取代 `%d`

图 1-2

(5) `#include` 语句是编译预处理语句,其作用是将由双引号或尖括号括起来的文件内容代替这行 `#include` 命令,也就是使该文件的内容被整个调到 `main` 函数的前面。“`.h`”是“头文件”的后缀,输入输出函数一般需要使用 `#include` 语句将包含输入输出函数说明信息的头文件“`stdio.h`”包含到源文件中。

◇ 随堂练习

编程:由键盘输入长方形的长和宽,求长方形的面积。

案例 1-3

求一个数的绝对值并输出。

◇ 案例程序

```
#include<math.h>
main()
{
    float num, n;           /*定义两个实数类型的变量*/
    scanf("%f", &num);      /*从键盘输入数值保存在变量 num 对应的存储单元中*/
    n=abs(num);             /*调用库函数 abs 求 num 变量的绝对值并保存在变量 n 中*/
    printf("%f\n", n);      /*输出所求的绝对值*/
}
```

运行程序:

-1.5 ✓

1.5

◇ 案例分析

(1) `float num,n`;表示定义变量 `num` 和 `n` 为实数类型(浮点型)，“`float`”表示实型，是 C 语言数据类型关键字。

(2) “`%f`”是格式控制符,控制输入输出的数据是实型。

(3) `abs` 是 C 系统的库函数,在调用数学库函数时,要用到一些系统提供的说明信息,这些信息包含在 `math.h` 头文件中,所以在程序的开始要用预处理命令 `#include` 包含进来。

◇ 随堂练习

编程调用库函数，计算一个数的正弦值。

◇ 知识链接

1. C 语言的关键字

关键字是 C 语言已经定义的、具有特殊功能和含义的单词、单词缩写或者单词组合。表 1-1 列举了 32 个关键字，它们与标准 C 句法结合，形成了程序设计语言 C。

表 1-1 C 语言的关键字

auto	break	case	char	const	continue
default	do	double	else	enum	extern
float	for	goto	if	int	long
register	short	signed	sizeof	static	return
struct	switch	typedef	union	unsigned	void
volatile	while				

2. C 语言的库函数

C 语言提供了丰富的库函数，每个函数实现一定的功能。C 语言本身语句很少，许多功能都是通过库函数完成的。例如，前面用到的输入输出的功能和求绝对值的功能都是通过调用库函数实现的。

3. ASCII 字符集

在计算机中，所有的信息都用二进制表示。其中应用最为广泛的编码方式就是 ASCII 码。使用的字符在计算机中就是以 ASCII 码方式存储的。ASCII 码是美国标准信息交换码（American Standard Code for Information Interchange），已被国际标准化组织认定为国际标准，详见附录 A。

1.2 C 程序的上机过程



学习目标

- 掌握 C 程序的编译运行过程。
- 掌握 C 程序的错误查找方法。

案例 1-4

一个简单的 C 程序的运行过程。

◇ 案例分析

(1) 把 Turbo C 2.0 文件夹放在电脑 C 盘下面。

(2) 打开文件夹内的 TC 可运行文件，进入 Turbo C 2.0，以下简称 Turbo C。

(3) 按【Alt+F】快捷键打开【File】（文件）菜单，用键盘上的【↑】和【↓】选择【New】菜单项，进入编辑状态，如图 1-3 所示。在 Turbo C 中，按【Alt】+【对应菜单的首字母】即

可打开相应的菜单。另外，也可按【F10】功能键选中相应的菜单。

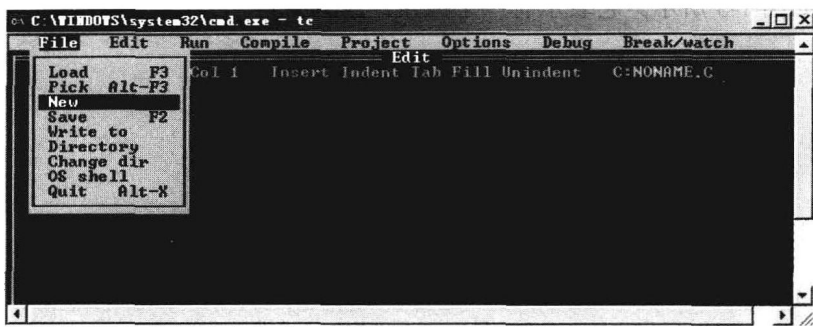


图 1-3

(4) 编辑。在编辑区域输入以下代码：

```
#include "stdio.h"
main()
{
    printf("No pains, no gains! ");
}
```

还可以事先将源代码写在记事本中，将其保存为.c 文件，然后打开 Turbo C 的【File】菜单中的【Load】菜单项或按【F3】键，在 Turbo C 中打开该源文件，这时在 Turbo C 中就出现了在记事本中输入的 C 代码，接着编译、连接和运行它即可。

(5) 保存。输入程序后对程序进行认真检查，并改正已发现的错误。这时应及时将程序保存。选择【File】菜单下的菜单项【Save】或者直接按快捷键【F2】，在弹出的对话框中出现文件的默认路径和默认文件名，这时可以修改路径和文件名，如图 1-4 所示。这里把路径改为 D 盘 ZHANGSAN(张三)文件夹下，文件名为 CH1_1.C，表示模块一第一个 C 程序。

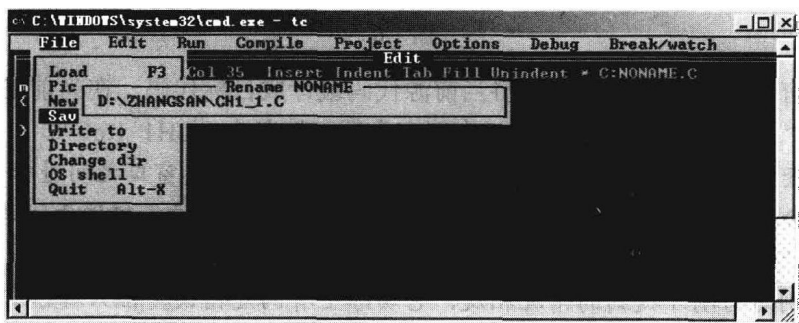


图 1-4

(6) 编译。编辑好的 C 代码，计算机是不能识别的。这是因为计算机只能识别称为机器语言的二进制命令，所以要想执行这段代码，就必须将源代码转换成机器语言。也就是要做第二步编译源代码，将源代码作为输入，并生成一个磁盘文件，该文件包含了与源代码语句对应的机器语言指令，该指令即为目标代码，而包含它们的磁盘文件称为目标文件。

在 Turbo C 中，编译源代码，按【Alt+C】快捷键打开【Compile】菜单，选择【Compile】

菜单下的【Compile to OBJ】菜单项，或直接按【Alt+F9】快捷键，编译程序后，出现编译信息提示窗口，如图 1-5 所示。它表示有一个错误，没有警告。

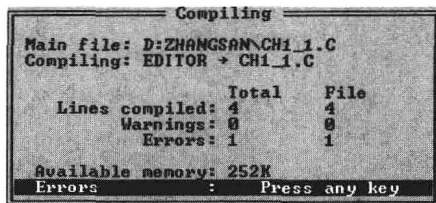


图 1-5

按任意键信息窗口 (Message) 被激活，在出现第一个错误信息提示处以白色高亮度显示，同时，编辑窗口中相应的错误语句所在行也以白色高亮度显示，如图 1-6 所示。按回车键【Enter】可切换到编辑窗口出现错误的语句行，同时对应的错误提示信息以红色亮条在编辑区上部显示。有时候程序可能只有一处错误，但会显示多条错误信息。所以一般修改一处错误后，应该再次进行编译。

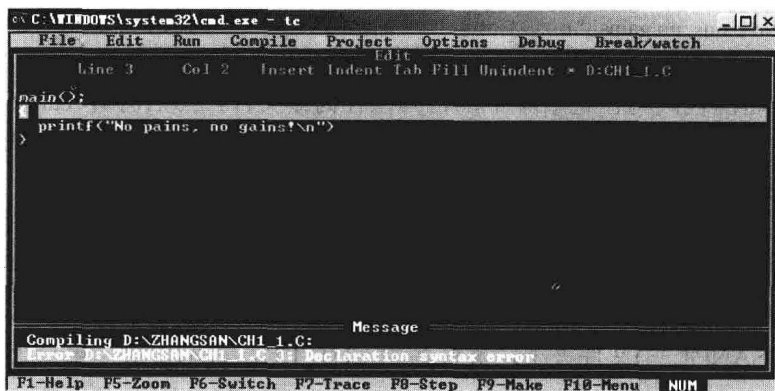


图 1-6

在程序没有错误的情况下，便可将当前源代码编译成目标代码。在当前工作目录下产生一个扩展名为“.obj”的目标程序文件。本例产生的文件名为“CH1_1.obj”。

提示：当查找错误时，可以使用【F6】键在编辑窗口和消息窗口之间进行切换。

错误有时会出现现在错误语句行前面的某条语句处，在图 1-5 中，错误原因是第一行 main() 后不应有分号“;”，而出错信息提示却在第二行上。

这个程序刚好与刚才提到的情况相反，它实际上有两处错误，但只提示有一处错误，在这种情况下，改正一处错误并再次编译后，还会再提示第二处错误。本例的第二处错误是第三行语句应该加分号作为语句结束。

(7) 连接。有了目标文件后，程序还不能直接运行，还必须将目标文件与系统提供的库函数和包含文件 (“#include”命令所包含的文件) 等连接成一个可执行文件，即扩展名为“.exe”的文件，这时才形成最后的可执行文件。

选择【Compile】菜单下的【Link EXE file】菜单项，在当前工作目录中生成一个可执行文件。本例生成可执行文件“CH1_1.exe”。

提示：可以一次完成编译和连接，选择【Compile】菜单下的【Make EXE file】菜单项，或者按【F9】快捷键，在当前工作目录中生成一个目标文件 (*.obj) 和一个可执行文件 (*.exe)。另外，可以指定生成的目标文件和可执行文件到指定的输出目录。本例就可以重新设定输出目录为“D:\ZHANGSAN”。按【Alt+O】快捷键打开【Options】菜单，选择【Options】菜单下的【Directories】菜单项，在弹出对话框中选中【Output directory】选项，然后设定输出目录（Output directory）为“D:\ZHANGSAN”，如图 1-7 所示。

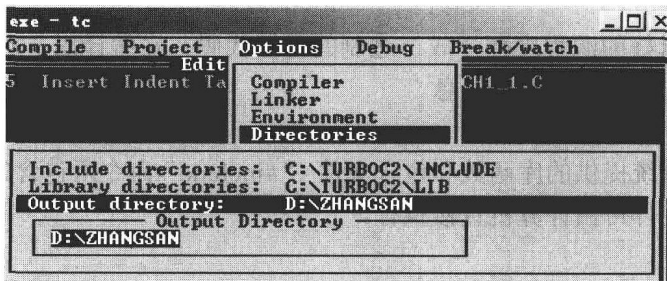


图 1-7

最后在相应目录下，可以查看本例的源文件、目标文件和可执行文件，如图 1-8 所示。



图 1-8

（8）运行。在 Turbo C 集成环境下，按【Alt+R】快捷键打开【Run】菜单，选择【Run】菜单下【Run】菜单项，或者直接按【Ctrl+F9】快捷键，则可执行文件开始运行。

提示：【Ctrl+F9】快捷键的作用不仅是运行程序，而且包含编译、连接和运行，也就是说，如果编辑好源程序后，经检查认为没有问题，通过按【Ctrl+F9】快捷键，则可以一次完成从编译、连接到运行的全过程。

（9）观察运行结果。选择【Run】菜单下的【User screen】菜单项，即可打开用户屏幕查看结果。也可以使用快捷键【Alt+F5】查看运行结果。本例运行结果如图 1-9 所示。



图 1-9

提示：运行过程的常用快捷键是：

F9：进行编译和连接。

Ctrl+F9：运行。

Alt+F5：查看运行结果。

◆ 随堂练习

按照上机过程运行一个简单的 C 程序。

◇ 知识链接

C 语言采用编译方式将源文件转换为二进制的目标代码。从编写到运行一个 C 程序一般经过以下几个步骤：

1. 编辑

内容包括：①将源程序逐个字符输入到计算机内存；②修改源程序；③保存源程序到磁盘文件中。源程序是以 ASCII 代码的形式输入和存储的，不能被计算机执行。

2. 编译

编译就是将编辑好的源程序翻译成二进制的目标代码。在编译时，对源程序进行语法检查，如有错，则在屏幕上显示出错信息。

3. 连接

将目标文件与系统提供的库函数和包含文件（“#include”命令所包含的文件）等连接成一个可执行文件，该文件可由计算机直接执行。

4. 执行

可执行文件连接好后，就可以运行。需要指出的是，经过成功编译和连接的可执行文件不一定能正确运行，或者运行所得的结果与期望不符，出现这种情况，则需返回从头开始检查，找出导致错误的原因。

从编写到运行一个 C 程序的全过程如图 1-10 所示。

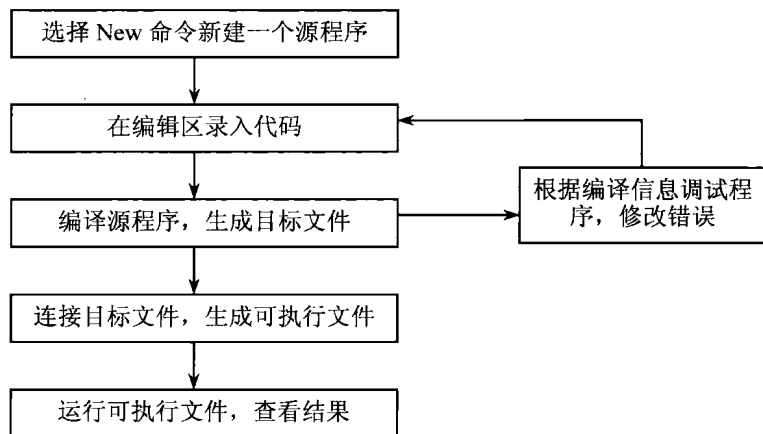


图 1-10

课后练习

◇ 选择题

1. 一个 C 程序的执行是从（ ）。
 - A) 本程序的 main 函数开始，到 main 函数结束
 - B) 本程序文件的第一个函数开始，到本程序文件的最后一个函数结束
 - C) 本程序的 main 函数开始，到本程序文件的最后一个函数结束
 - D) 本程序文件的第一个函数开始，到本程序 main 函数结束

2. 以下叙述不正确的是 ()。
- A) 一个 C 源程序可由一个或多个函数组成
 - B) 一个 C 源程序必须包含一个 main 函数
 - C) 在 C 程序中, 注释说明只能位于一条语句的后面
 - D) C 程序的基本组成单位是函数
3. C 语言规定: 在一个源程序中, main 函数的位置 ()。
- A) 必须在程序的开头
 - B) 必须在系统调用的库函数的后面
 - C) 可以在程序的任意位置
 - D) 必须在程序的最后
4. 一个 C 语言程序是由 ()。
- A) 一个主程序和若干子程序组成
 - B) 函数组成
 - C) 若干过程组成
 - D) 若干子程序组成

◇ 程序分析

分析下面程序的运行结果。

```
main()
{
    int a,b,df;
    a=200; b=50; df=a-b;
    printf("df=%d\n",df);
}
```

◇ 动手编程

1. 为一个学生管理信息系统设计一个界面, 显示下列菜单:

Menu

- ```
=====
1. Input names and scores
2. Search scores of students
3. Modify scores of some students
4. List all information
5. Quit
=====
```

Please input your choice(1-5):

- 2. 编程求三个数的和。
- 3. 编程求两个数中较大的数。
- 4. 由键盘输入一个长方体的长宽高, 求长方体的体积。

## 上机实训

#### ◇ 实训目的

- 掌握 Turbo C 的基本操作方法。
- 通过编程理解 C 语言程序的结构。

- 掌握 C 程序的编辑、编译、连接和运行的过程。

#### ◇ 实训内容

1. 上机按照案例 1-4 的操作步骤运行该实例。

2. 运行程序步骤练习

(1) 打开 Turbo C 集成环境。

(2) 熟悉 Turbo C 集成环境。

1) 熟悉各菜单：按【Ctrl+F】打开【File】菜单，按【↑】和【↓】键选择各菜单项。按【→】和【←】键，分别选中【Edit】、【Run】、【Compile】、【Project】、【Options】、【Debug】、【Break/Watch】菜单，观察各菜单下的菜单项，了解它们的作用。

2) 选择【File】菜单下的【New】菜单项，在编辑窗口【Edit】中输入如下程序，同时观察窗口顶部左端 Line（行）和 Col（列）后面数字的变化。

```
main()
{
 printf("To learn C language well, you have to work hard! \n");
}
```

3) 进行编译和连接，分析编译信息窗口和消息【Message】窗口中显示的信息，如有出错信息，则须逐条改正，直至编译和连接成功。

3. 上机运行“动手编程”中的程序。

## 模块二 数据类型、运算符和表达式

计算机解决实际问题，就要把实际问题描述成数据，然后对数据进行处理。数据是操作的对象，操作的目的是对数据进行加工、处理，以得到期望的结果。

一个程序应包括以下两方面内容：

(1) 对数据的描述。指定数据类型和数据的组织形式，即数据结构 (data structure)。

(2) 对操作的描述。即操作步骤，也就是算法 (algorithm)。即解决怎么做和做什么的问题。

提示：程序=数据结构+算法。

### 2.1 数据类型



#### 学习目标

- 了解数据类型的划分标准。
- 掌握数据类型的分类。

#### 案例 2-1

C 语言的数据类型分析。

##### ◇ 案例分析

C 语言提供的数据结构是以数据类型形式出现的。所谓数据类型就是按被说明量的性质、表示形式、占据存储空间的多少、构造特点来划分的一种数据的组织形式。C 语言的数据类型包含以下四大类，如图 2-1 所示。

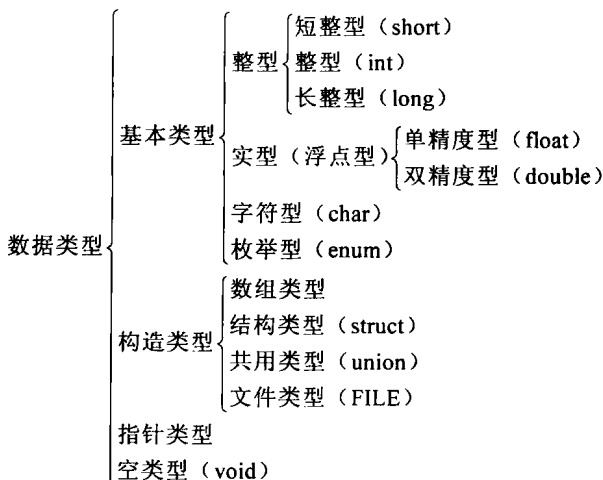


图 2-1 数据类型

### ◇ 随堂练习

分析日常生活中的一些数据适合用什么数据类型表示, 比如: 学生姓名、学生性别、学生人数、学生成绩等。

### ◇ 知识链接

C 语言的数据类型:

#### 1. 基本类型

基本类型中的每一个数据类型都是 C 语言中可以操作的最小数据单元, 不可以再分解为其他类型。本模块主要讲述的就是该种类型, 其他的类型将在后面的模块中介绍。

#### 2. 构造类型

构造类型是在基本类型的基础上构造产生的复合数据类型, 简单地说, 构造类型由基本类型组合而成。构造类型包括数组类型、结构类型和共用类型三种。

#### 3. 指针类型

指针类型是 C 语言的一个重要特色, 是 C 语言区别于其他高级语言的一种重要的数据类型。指针类型的数据用来表示变量等的存储地址, 指针的灵活运用可以大大提高编程的效率, 是学习 C 语言应当掌握的知识。

#### 4. 空类型

为了说明一个函数没有返回值或没有参数, 可以在定义函数时声明其返回值或参数类型为空类型, 用关键字 “void” 表示。

提示: C 语言数据有常量和变量之分, 它们分别属于以上这些类型。即常量有整型、实型、字符型等类型, 变量也是一样。

## 2.2 常量和变量



### 学习目标

- 掌握常量的类型和表示方法。
- 掌握变量的三个要素和命名规则。
- 掌握变量的定义和初始化方法。

C 语言中的数据, 根据在程序运行过程中是否可以改变, 分为常量和变量。

#### 2.2.1 常量

在程序运行过程中, 其值不能改变的量称为常量。常量一般用于给变量赋初始值, 或作为运算式中的操作数等。

##### 案例 2-2

区分下列常量的类型:

127, 3.14159, 'Y', "Sound"

### ◇ 案例分析

根据 C 语言的常量类型划分, 127 是整型常量, 3.14159 是实型常量, 'Y' 是字符型常量,