



“十二五”职业教育国家规划教材
经全国职业教育教材审定委员会审定

高等职业教育
教学用书

C语言程序设计

主 编 赵 睿

高等教育出版社



“十二五”职业教育国家规划教材
经全国职业教育教材审定委员会审定

C Yuyan Chengxu Sheji

C 语言程序设计

主 编 赵 睿

高等教育出版社·北京

内容提要

本书是“十二五”职业教育国家规划教材,经全国职业教育教材审定委员会审定。

全书包括编程起步(程序结构、数据类型及其运算)、顺序程序设计训练、分支程序设计训练、循环程序设计训练、数组应用训练、模块化程序设计训练、构造数据类型应用训练、指针应用训练、文件操作训练、位操作训练和综合实训,共十一个模块。每个模块基本上由“知识准备”、“边学边练”、“拓展提升”、“总结归纳”和“强化练习”五个部分构成,其中“强化练习”涵盖了全国计算机等级考试二级 C 语言的基本内容。为了便教利学,本书另配有丰富的教学资源,包括课程标准、教学课件、源代码等。

本书可作为高等职业院校电子信息类各专业 C 语言程序设计课程的教材,也可作为全国计算机等级考试二级 C 语言考试的复习辅导用书。

图书在版编目 (CIP) 数据

C 语言程序设计 / 赵睿主编. —北京: 高等教育出版社, 2014. 8 (2015. 7 重印)
ISBN 978-7-04-038768-1

I. ①C… II. ①赵… III. ①C 语言-程序设计-高等职业教育-教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2014) 第 164920 号

策划编辑 张玗琳 责任编辑 张玗琳 王 威 封面设计 吴 昊 责任印制 蔡敏燕

出版发行	高等教育出版社	咨询电话	400-810-0598
社 址	北京市西城区德外大街 4 号	网 址	http://www.hep.edu.cn
邮政编码	100120		http://www.hep.com.cn
印 刷	江苏德埔印务有限公司		http://www.hepsh.com
开 本	787mm×1092mm 1/16	网上订购	http://www.landraco.com
印 张	20.75		http://www.landraco.com.cn
字 数	451 千字	版 次	2014 年 8 月第 1 版
购书热线	021-56717287	印 次	2015 年 7 月第 2 次印刷
	010-58581118	定 价	38.50 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换
版权所有 侵权必究
物 料 号 38768-00

出版说明

2012年12月,教育部职业教育与成人教育司启动了“十二五”职业教育国家规划教材的选题立项工作。作为全国最大的职业教育教材出版基地,高等教育出版社认真贯彻执行《教育部关于“十二五”职业教育教材建设的若干意见》(教职成〔2012〕9号)文件精神,按照“统筹规划、优化结构、锤炼精品、鼓励创新”的原则,整合全国的优质出版资源,积极参与了该项工作。目前,已获立项的教材相继完成了编写工作,经教育部组织专家审定并公示后,陆续出版。

高等教育出版社国家规划教材的作者中有参与制定高等职业教育新专业教学标准的专家,有高等职业教育国家专业教学资源库建设项目的主持人,有学科领域的领军人物,有企业的技术人员,他们是保证教材编写质量的基础。

高等教育出版社国家规划教材主要突出以下五个特点:

1. 执行新标准

以《高等职业学校专业教学标准(试行)》为依据,服务经济社会发展和人的全面发展。教材内容与职业标准对接,突出就业能力培养。

2. 构建新体系

教材整体规划、统筹安排,注重系统培养,兼顾多样成才,突出产教融合。遵循技术技能人才培养规律,构建服务于中职高职衔接、职业教育与普通教育相互沟通的现代职业教育教材体系。

3. 找准新起点

教材编写遵循易用、易学、易教的原则,强调以学生为中心,符合职业教育的培养目标与学生认知规律。

4. 推进新模式

在高等职业教育工学结合、知行合一的人才培养模式下,改革教材编写体例,创新内容呈现形式,推进“任务驱动”、“项目化”、“工作过程导向”、“理实一体化”、“做中学、做中教”等教学模式的实施。

5. 配套新资源

秉承高等教育出版社打造数字化教学资源与传统优势,教材内容与高等职业教育国家专业教学资源库紧密结合,同时配套建设课程教学资源,为教学提供全面支持和增值服务。

为了更好地为教学服务,高等教育出版社将以国家规划教材为基础,组织教师培训和教学研讨活动,通过与教师互动以及滚动建设立体化教学资源,把教材建设提高到一个新的水平。

高等教育出版社

2014年7月

前言

本书是“十二五”职业教育国家规划教材,经全国职业教育教材审定委员会审定。本书是依据教育部最新印发的《高等职业学校专业教学标准(试行)》中关于本课程的教学要求编写而成的。

“C 语言程序设计”是高等职业教育电子信息类专业的专业基础课,也是全国计算机等级考试的重要内容。编者长期从事高等职业教育计算机语言课的教学及研究工作,在总结多年有效教学经验的基础上,依据教育部最新发布的专业教学标准,立足于技能的培养,引入企业的专业人员,共同编写了本书。

本书的主要特色有:

1. 结合实例编排教学内容,由浅入深。书中知识点大多辅以对应的程序实例,并提供有算法设计、运行结果和程序解析,便于学生理解,突出了“做中学、做中教”的职业教育教学特点。本书的编写注重知识的层次和技能培养的渐进性,设置了三个学习阶段:基础准备阶段、应用练习阶段、综合训练阶段。基础准备阶段包含编程起步、模块一至模块三,主要学习程序设计的语言表达以及程序结构,以验证性的实验为主;应用练习阶段包含模块四至模块七,以完成应用练习为主;综合训练阶段包含模块八、模块九和综合实训,通过应用程序的完整设计,培养学生软件设计的思想、方法,提高 C 语言的综合应用能力。同时,每个模块的内容安排层层递进,有利于学习的开展和深入。

2. 强化技能训练。按照程序设计过程,对课程教学内容进行优化整合,提炼出了十一个教学模块,每个模块包含“知识准备”、“边学边练”、“拓展提升”、“总结归纳”和“强化练习”五个部分。每个模块都以一个程序引例导入,知识点的讲解与相应的程序代码紧密结合。在边学边练中,引导学生进行实践,学生在解决具体问题的过程中能够完成相应工作任务,掌握相关理论知识,发展职业能力。在拓展提升部分,对相关知识的应用进行了扩展,能让学有余力的学生深入自学,包括 Turbo C 和 Microsoft Visual C++ 的应用。在总结归纳中,简单明晰地以框图的形式概括了本模块的知识结构。

3. 双证融通。理论知识的选取紧紧围绕完成编程任务的需要和够用的原则来进行,突出对学生职业能力的训练,并融入全国计算机等级考试二级 C 语言中的考核点,“强化练习”部分以等级考试二级 C 语言的考核点为主,旨在培养高技能型的“两证”人才。

4. 配套教学资源丰富。本书另配有丰富的教学资源,包括课程标准、教学课件、源代

码等,索取方式参见书后的“教学资源索取单”。

本书建议教学课时数为 90,建议具体课时分配如下:

模 块	课 时		
	理 论	实 训	合 计
编程起步	6	6	12
模块一:顺序程序设计训练	4	4	8
模块二:分支程序设计训练	4	4	8
模块三:循环程序设计训练	4	4	8
模块四:数组应用训练	4	4	8
模块五:模块化程序设计训练	8	8	16
模块六:构造数据类型应用训练	2	2	4
模块七:指针应用训练	6	6	12
模块八:文件操作训练	4	4	8
模块九:位操作训练	2	2	4
机动	2		2
合计	46	44	90
综合实训(一周)	10	20	30

书中带*的章节内容难度较大,可作为选学内容。各院校可根据实际情况选择教学内容。

本书由赵睿主编,参加编写的有张太芳、蒲晓妮、李模刚、孔令赞、赵振学、张平华、蒋永丛。全体编写人员在共同讨论提纲的基础上分别收集材料和编写,最后由赵睿统稿。

本书适合作为高等职业教育电子信息类各专业相关课程的教材,也适合用作全国计算机等级考试二级 C 语言的培训用书。

本书在编写的过程中参考了不少的专著和教材,得到了有关专家学者及院校领导的大力支持,在此一并表示感谢!

由于编者理论水平和专业知识有限,书中疏漏之处在所难免,恳请专家学者、使用本书的老师、同学和读者批评指正,反馈意见请发至电子邮箱:zhaorui68@163.com。

编 者

2014 年 7 月

目录

编程起步	001
知识准备	001
0.1 C 语言程序概述	002
0.2 C 语言程序开发过程	005
0.3 基本数据类型	006
0.4 运算符和运算表达式	013
边学边练	020
拓展提升	024
0.5 C 语言的发展历程及特点	024
0.6 Turbo C 应用简介	025
0.7 Microsoft Visual C++ 6.0 应用简介	026
总结归纳	029
强化练习	029
模块一 顺序程序设计训练	032
知识准备	032
1.1 程序和算法	033
1.2 基本语句	036
1.3 字符输入输出	038
1.4 格式输入输出	040
边学边练	047
拓展提升	051
1.5 Turbo C 环境下的程序调试	051

1.6 Visual C++ 6.0 环境下的程序调试	052
1.7 Visual C++ 6.0 的标准输入输出	053
总结归纳	054
强化练习	056
模块二 分支程序设计训练	060
知识准备	060
2.1 if-else 语句	061
2.2 switch-case 语句	064
边学边练	067
拓展提升	073
* 2.3 多分支结构的 if-else 语句	073
* 2.4 switch-case 语句的巧用	076
总结归纳	077
强化练习	079
模块三 循环程序设计训练	083
知识准备	083
3.1 while 语句	085
3.2 do-while 语句	086
3.3 for 语句	088
3.4 循环嵌套	089
3.5 循环退出语句	091
边学边练	094
拓展提升	100
* 3.6 goto 语句综述	100
总结归纳	100
强化练习	101
模块四 数组应用训练	106
知识准备	106
4.1 一维数组	107

4.2 二维数组	111
4.3 字符数组	117
4.4 字符串	121
边学边练	125
拓展提升	128
* 4.5 Josephus 问题	128
总结归纳	130
强化练习	131
模块五 模块化程序设计训练	135
知识准备	135
5.1 函数的定义及调用	136
5.2 函数的嵌套调用	143
5.3 函数的递归调用	145
5.4 局部变量和全局变量	146
5.5 动态存储变量和静态存储变量	150
5.6 外部函数和内部函数	153
边学边练	154
拓展提升	159
5.7 编译预处理	159
总结归纳	166
强化练习	167
模块六 构造数据类型应用训练	174
知识准备	174
6.1 结构体类型及其变量	175
6.2 结构体数组	178
* 6.3 共用体类型变量	180
边学边练	183
拓展提升	186
* 6.4 枚举类型	186
* 6.5 自定义数据类型	189
总结归纳	191
强化练习	191

模块七 指针应用训练	195
知识准备	195
7.1 指针	196
7.2 指针变量	197
7.3 数组与指针	203
7.4 结构体与指针	209
* 7.5 字符串与指针	211
* 7.6 函数与指针	213
边学边练	215
拓展提升	219
* 7.7 命令行参数	219
总结归纳	221
强化练习	222
模块八 文件操作训练	229
知识准备	229
8.1 C 语言的文件系统	230
8.2 文件的打开与关闭	231
8.3 文件的读写	233
边学边练	239
拓展提升	245
* 8.4 文件定位操作	245
总结归纳	248
强化练习	248
模块九 位操作训练	251
知识准备	251
9.1 位运算符	253
边学边练	260
拓展提升	263
* 9.2 位段	263
总结归纳	265
强化练习	266
综合实训	268
实训准备	268

10.1 程序设计步骤	268
10.2 程序设计应用实例	269
实训练习	286
拓展提升	287
10.3 链表应用	287
10.4 单向动态链表	289
总结归纳	300
单项训练	300
附录	303
附录 A 部分习题参考答案	303
附录 B ASCII 字符编码一览表	305
附录 C 常用库函数选摘	306
附录 D 运算符及其结合方向	311
附录 E Turbo C 编译时的出错信息和警告信息	312
参考文献	318

编程起步

能力目标

- (1) 掌握 C 语言程序的开发过程;
- (2) 掌握程序的结构;
- (3) 熟悉上机操作的环境;
- (4) 掌握各种数据类型;
- (5) 熟练运用运算符与表达式。

知识准备

【引例任务】 用 C 语言编写程序输出“老师,您好!”。

【程序代码】

```
#include <stdio.h>
void main(){
    printf("老师,您好!");    /* 输出“老师,您好!” */
}
```

【程序运行】

老师,您好!

【引例解析】 这是一个完整的 C 语言程序。“/*”与“*/”及其之间的内容是对程序的注释,用于对语句进行说明,对程序的运行没有任何影响。

main()是主函数。“{”与“}”之间的语句是 main()函数的内容,是程序的主体,也称**函数体**。所有的 C 语言程序都必须包含有一个 main()函数。程序从 main()函数的第一行语句开始执行,到最后一条语句结束。

“printf("老师,您好!")”的功能是在屏幕上显示“老师,您好!”。**printf()函数**是 C 语言

提供的按指定格式进行**标准输出**的函数,其功能是输出由双引号括起来的字符序列。在其中可以包含控制字符,“\n”是**换行控制符**,表示该符号之前的内容输出完毕后换行显示后续内容。

本模块的主要内容是学习 C 语言源程序的构成及开发过程,数据类型、常量及变量的定义、运算符及其表达式。

0.1 C 语言程序概述

【做中学 0-1】 键盘输入两个数,比较大小,并输出两个数中的较大值。

【程序代码】

```
#include <stdio.h>
int max(int x,int y);          /* max()函数声明 */
void main(){
    int a,b,m;
    scanf(" %d, %d",&a,&b);
    m = max(a,b);              /* 用户自定义函数的调用 */
    printf("a = %d,b = %d,m = %d",a,b,m);
}

int max(int x,int y) {         /* 用户自定义函数,求两个数中的较大值 */
    int m;
    if (x>y)
        m = x;
    else
        m = y;
    return m;
}
```

【程序运行】*

8,4↵
a = 8,b = 4, m = 8

【程序解析】 程序中包括两个函数 main()和 max()。在函数 main()中,第一行是变量说明部分,后面的多行语句是执行部分。

scanf()用来读取用户从键盘输入的值。scanf()和 printf()的使用方法将在后面的

* 粗体部分表示输入,↵表示按回车键。

学习中详细介绍,这里就不再赘述了。

【知识点】

(1) C 语言程序特点

- ① 每条语句后面都以“;”作为终止符,它是 C 语句结束符。
- ② 每个程序必须有一个且只能有一个主函数,程序从主函数开始执行。一个应用程序可以包含多个源程序文件,每个源程序文件又可以包含多个用户自定义函数。函数之间是相互独立、相互平行的。**源程序的最基本组成单位是函数。**

③ 在 C 语言中,大、小写字母是有区别的。

④ 程序中可以加注释部分,注释有块注释和行注释两种方式。用“/*……*/”提供的注释是块注释,可以注释多行。用“//”提供的注释是行注释,只能注释一行,但 turbo C 中没有这种注释方式。

C 语言的函数分为两类:系统本身提供的库函数和用户的自定义函数,其中库函数又称标准函数。库函数的定义在相应的头文件(头文件的扩展名是.h)中,如果要调用这些库函数,要在源程序最前面使用 include 语句将相应的头文件包含进来,然后在程序中就可调用这些库函数。printf()函数的头文件是 stdio.h,在程序中使用“#include <stdio.h>”语句后,就可以在程序中使用 printf()函数实现输出功能了。用户自定义的函数同标准函数一样,可以在主函数中调用。

(2) C 语言程序书写格式

- ① 程序中每行可写一条语句,也可写多条语句,一般一行写一条语句。
- ② 程序的书写要注意适当缩进,使程序清晰易读。
- ③ 程序中的花括号必须成对出现。
- ④ 在写程序时,要习惯使用注释。

0.1.1 程序构成

通过前面的学习,可以看到 C 语言程序的一般构成形式如下所示。其中 f1()~fn()代表用户定义的函数。

编译预处理行

全局变量说明

```
void main() {
```

 局部变量说明

 程序段

```
}
```

```
返回数据类型 f1(形式参数说明){
```

 局部变量说明

 程序段

```
}
```

```
返回数据类型 f2(形式参数说明){
```

```
    局部变量说明
```

```
    程序段
```

```
}
```

```
...
```

```
返回数据类型 fn(形式参数说明){
```

```
    局部变量说明
```

```
    程序段
```

```
}
```

说明

(1) C 语言的变量在使用之前必须先定义其数据类型, 未经定义的变量不能使用, 且定义变量的语句必须放在可执行语句前面;

(2) 程序段通常由多条语句组成;

(3) 形式参数说明用于传值, 参见后面的函数调用部分内容;

(4) 局部变量、全局变量的用法含义, 参见模块五的内容;

(5) 花括号内的内容统称为函数体;

(6) 用户定义的函数名由用户命名(命名应符合标识符的命名规则);

(7) 用户自定义函数可置于主函数之前, 也可置于主函数之后。若用户自定义函数在主函数之后, 且要在主函数中调用时, 要在主函数之前对被调用函数进行函数声明。

0.1.2 特殊字符

1. 关键字

关键字是 C 语言编译程序本身所使用的专用词, 具有特定的含义, 例如 `int` 用来定义整数类型。关键字一般都是用小写字母来表示的。C 语言的关键字有: `auto`、`break`、`case`、`char`、`const`、`continue`、`default`、`do`、`double`、`else`、`enum`、`extern`、`float`、`for`、`goto`、`if`、`int`、`long`、`register`、`return`、`short`、`signed`、`sizeof`、`static`、`struct`、`switch`、`typedef`、`union`、`unsigned`、`void`、`volatile`、`while`。

2. 标识符

C 语言中, 变量、函数都需要有一个名称, 即标识符。『做中学 0-1』中的 `max` 就是标识符。用户自定义的标识符要符合 C 语言标识符的命名规则。C 语言标识符命名规则如下:

(1) 标识符由字母(A~Z, a~z)、下划线(_)或数字(0~9)组成。

(2) 标识符必须以字母(A~Z, a~z)或下划线(_)开头。

(3) C 语言中标识符严格区分字母大小写。例如 `aB12`、`Ab12`、`AB12`、`ab12` 是不同的标识符。

(4) 标识符不能使用 C 语言的关键字。例如 `char`、`do`、`for`、`if`、`int` 等。

例如,这些是不合法的标识符: 5abc、aa - bb、a&b、M. H. Thatcher、a#、abY。

这些是合法的标识符: A123、a_23、_123、a_b。

定义标识符时尽量做到“见名知意”。如: name 表示姓名,age 表示年龄,或汉语拼音的首字母 xm 表示姓名,nl 表示年龄。

标识符的长度不要过长,建议在 8 个字符以内。不同编译系统支持的标识符长度不同。如 Turbo C 2.0 编译器支持的标识符最大长度为 32 个字符,而 Visual C++ 6.0 中支持的标识符最大长度为 2 048 个字符。

3. 分隔符

分隔符用来使编译器确认代码在何处分隔,C 语言中分隔符包括注释符、空白符及普通分隔符。

(1) 注释符。

(2) 空白符。空白符包括空格、回车、换行和制表符(Tab 键),用来分隔程序的各基本成分。一个或多个空白符的作用完全一样。

(3) 普通分隔符。普通分隔符的作用也是用来分隔程序的各成分,在程序中有特定的含义,不能省略。普通分隔符见表 0-1。

表 0-1 C 语言的普通分隔符

普通分隔符	名 称	用 途
{ }	大括号/花括号	定义复合语句、函数体及数组的初始化
[]	方括号	定义数组类型及引用数组元素
()	小括号	区分函数名及其参数,表达式中限定运算顺序
;	分号	语句结束标志
:	冒号	标号
.	圆点	区分复合类型变量与其成员变量
,	逗号	构成逗号表达式
?	问号	构成问号表达式

0.2 C 语言程序开发过程

C 语言程序的开发过程可以分成四个主要步骤: 编辑→编译→连接→执行,如图 0-1 所示。



图 0-1 C 语言程序开发过程

1. 编辑程序

用户将 C 语言的源程序输入计算机,以文本文件的形式存放在磁盘上。其文件标识为:“文件名.c”。其中,“文件名”是用户指定的、符合操作系统规定的任意字符组合,扩展名是“.c”,表示该文件是 C 语言程序。

2. 编译程序

编译是把 C 语言源程序翻译成用二进制指令表示的目标程序。编译过程由 C 编译系统提供的编译程序完成。目标程序的文件标识是:“文件名.obj”。

3. 程序连接

用系统提供的连接程序将目标程序、库函数或其他目标程序连接装配成可执行的程序。可执行程序的文件标识为:“文件名.exe”。

4. 运行程序

将可执行程序投入运行,以获取编程处理的结果。与编译、连接不同的是,运行程序可以脱离语言编译环境。

对于不同的 C 语言上机环境,编译系统支持性能各异,上述步骤有些可以再分解,有些也可以集成进行批处理,但逻辑上是基本相同的。具体内容可参考拓展提升部分。

0.3 基本数据类型

C 语言规定,程序中使用的数据都属于某种数据类型。C 语言提供了丰富的数据类型,这些数据类型归纳如图 0-2 所示。

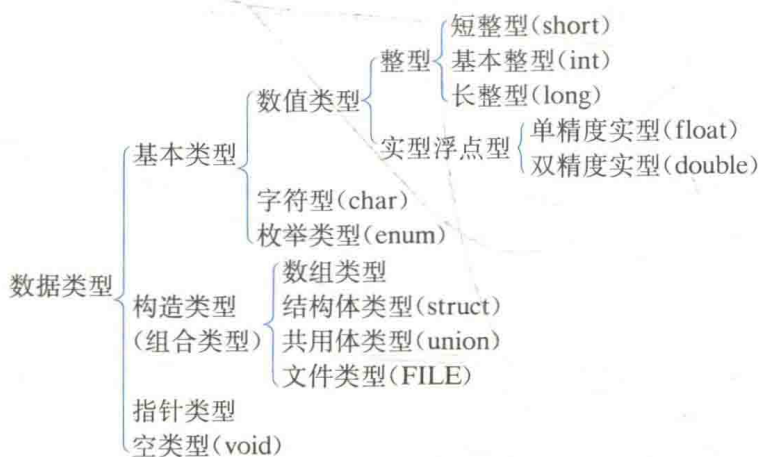


图 0-2 数据类型

C 语言对不同的数据分配不同长度的存储空间,C 语言并没有规定各种数据类型占有多少字节。基本数据类型为常用类型,表 0-2 列出了 Turbo C 中的基本数据类型长度和数值范围,供读者参考。在 Visual C++ 环境中,基本整型数据类型的长度为 4B。

C 语言中的数据还有常量和变量之分。由以上这些基本数据类型还可以构造更加复杂的数据结构,如栈和队列。