



高等学校应用型“十一五”规划教材

C语言程序设计

王 昱 陈延文
李德录 朱启家

编著



西安电子科技大学出版社
<http://www.xduph.com>

高等学校应用型“十一五”规划教材

C 语言程序设计

主编 王 昱 陈延文 李德录 朱启家

参编 张少刚 王 瑛 刘国兴 马广彦

周 珑 鲍丽红 崔兆顺 马海云

韩双旺 魏凯斌 周文勤 刘云芳

霍 静 李海芸 杨晓萍 杨 军

李芙蓉

西安电子科技大学出版社

2009

【内 容 简 介】

本书全面讲述了 C 语言的基本语法知识, 对使用 C 语言编写程序时的重点环节作了讲解, 并参照教育部考试中心制定的《全国计算机等级考试(二级)C 程序设计考试大纲》中有关 C 语言程序设计的要求编排书中内容。

本书主要内容包括: C 语言的基本数据类型和运算符、表达式, 结构化程序的几种基本结构, 构造数据类型(数组、指针、结构体、共用体、枚举类型等), 位运算, 文件操作, 常用的标准库函数, 预处理等内容。

本书由王昱、陈延文、李德录、朱启家主编, 王昱、陈延文完成了最后的统稿和定稿工作, 张少刚、王瑛、刘国兴、马广彦、周珑、鲍丽红、崔兆顺、马海云、韩双旺、魏凯斌、周文勤、刘云芳、霍静、李海芸、杨晓萍、杨军、李芙蓉等也参加了教材的编写工作。

本书结构合理, 篇幅适度, 既能作为高等院校 C 语言课程的教材, 也可以作为 C 语言程序设计爱好者的自学教材。

图书在版编目(CIP)数据

C 语言程序设计 / 王昱等编著. —西安: 西安电子科技大学出版社, 2009.2

高等学校应用型“十一五”规划教材

ISBN 978-7-5606-2181-4

I. C… II. 王… III. C 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2008)第 209510 号

策 划 杨丕勇

责任编辑 杨丕勇

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfxb001@163.com

经 销 新华书店

印刷单位 陕西光大印务有限责任公司

版 次 2009 年 2 月第 1 版 2009 年 2 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 14.5

字 数 338 千字

印 数 1~4000 册

定 价 20.00 元

ISBN 978-7-5606-2181-4/TP · 1113

XDUP 2473001-1

*** 如有印装问题可调换 ***

本社图书封面为激光防伪覆膜, 谨防盗版。

前 言

计算机语言是人类与计算机进行交互的工具，而高级程序设计语言则是为了既能使程序设计员易于设计程序又能使程序高效地被计算机执行而设计的。随着计算机技术和程序设计方法学的发展，在计算机世界里出现了各种各样的高级程序设计语言，从 BASIC 解释语言、面向过程的 C 语言及 PASCAL 语言，到目前被大家普遍关注的 C++ 语言、JAVA 语言等面向对象的程序设计语言，这些高级语言在不同阶段、不同应用领域都发挥着巨大的作用。C 语言作为计算机程序设计语言发展史上具有重要影响的高级程序设计语言，在计算机世界的各个领域已得到广泛的应用。近年来，C 语言课程已成为高等院校的首选计算机高级语言课程，并且已成为“全国计算机等级考试”的首选计算机高级语言。

本书根据编者多年的教学经验以及对教学当中容易混淆的概念和容易出现的错误进行分析，并参照教育部考试中心制定的《全国计算机等级考试(二级)C 程序设计考试大纲》的要求，精心编写而成。本书语言精炼，内容适当，既可以作为高等院校的 C 语言程序设计教材，又可以作为全国计算机等级考试(二级)C 程序设计考试的应试教材。

本书共 12 章，包括的内容有：C 语言初步介绍、程序的基本结构、C 语言基本数据类型、C 语言构造类型(指针、数组、结构体、共用体、枚举类型等)、函数、字符串、存储类型、作用域、生存期、编译预处理、动态内存分配、Bool 运算(位运算)、用户定义类型、输入输出、文件以及上机操作等。

由于编者水平有限，疏漏或不当之处在所难免，敬请读者批评指正。

编 者
2008 年 12 月

目 录

第 1 章 C 语言概述	1
1.1 C 语言简介	1
1.1.1 程序设计语言的概念	1
1.1.2 C 语言的发展历史	1
1.1.3 C 语言的特点	1
1.2 认识 C 语言程序	2
1.2.1 从 Hello World 程序开始	2
1.2.2 C 语言程序的组成	2
1.3 C 语言程序的执行过程	3
1.3.1 从编写 C 语言程序到执行该程序的完整过程	3
1.3.2 计算机是如何执行程序的	4
1.4 C 语言程序的上机步骤	4
1.4.1 C 语言程序上机环境介绍	4
1.4.2 C 语言程序上机步骤	4
习题	4
第 2 章 数据类型、运算符与表达式	5
2.1 字符集和标识符	5
2.1.1 字符集	5
2.1.2 标识符	5
2.1.3 系统定义标识符	6
2.1.4 用户定义标识符	6
2.2 常量和变量	7
2.2.1 常量	7
2.2.2 变量	8
2.3 数据类型	9
2.3.1 整型数据	10
2.3.2 实型数据	11
2.3.3 字符型数据	12
2.4 运算符与表达式	13
2.4.1 运算符与表达式概述	14
2.4.2 算术运算符与算术表达式	15
2.4.3 赋值运算符与赋值表达式	17
2.4.4 关系运算符与关系表达式	18

2.4.5	逻辑运算符与逻辑表达式	20
2.4.6	条件运算符与条件表达式	21
2.4.7	逗号运算符与逗号表达式	22
2.5	数据类型转换	23
2.5.1	数据类型的自动转换	23
2.5.2	数据类型的强制转换	23
	习题	24
第 3 章	顺序结构程序设计	26
3.1	C 语言语句概述	26
3.1.1	简单语句	26
3.1.2	复合语句	28
3.1.3	流程控制语句	28
3.1.4	空语句	28
3.2	数据输入与输出	28
3.2.1	格式输入/输出函数	28
3.2.2	字符输入/输出函数	32
3.3	顺序结构程序设计举例	33
	习题	35
第 4 章	选择结构程序设计	38
4.1	if 语句	38
4.1.1	if 语句的基本形式	38
4.1.2	if 语句的嵌套	39
4.2	switch 语句与 break 语句	41
4.2.1	switch 语句	41
4.2.2	break 语句	42
4.3	选择结构程序设计举例	43
	习题	48
第 5 章	循环结构程序设计	52
5.1	while 语句	52
5.1.1	while 语句的基本形式	52
5.1.2	while 语句的应用	53
5.2	do-while 语句	54
5.2.1	do-while 语句的基本形式	54
5.2.2	do-while 语句的应用	55
5.3	for 语句	56
5.3.1	for 语句的基本形式	56
5.3.2	for 语句的应用	57
5.4	break 语句和 continue 语句	58
5.4.1	break 语句	58

5.4.2	continue 语句	59
5.5	goto 语句	60
5.5.1	goto 语句的基本形式	60
5.5.2	goto 语句的应用	60
5.6	循环的嵌套	63
5.7	循环结构程序设计举例	66
	习题	72
第 6 章	数组	77
6.1	一维数组	77
6.1.1	一维数组的定义	77
6.1.2	一维数组的初始化	78
6.1.3	一维数组的引用	78
6.1.4	一维数组应用举例	79
6.2	二维数组	80
6.2.1	二维数组的定义	80
6.2.2	二维数组的初始化	81
6.2.3	二维数组的引用	83
6.2.4	二维数组应用举例	83
6.3	字符数组和字符串	87
6.3.1	字符数组的定义	87
6.3.2	字符数组的初始化	88
6.3.3	字符数组的引用	88
6.3.4	字符数组的输出	88
6.3.5	字符数组的输入	89
6.3.6	字符数组的应用举例	91
	习题	92
第 7 章	函数	94
7.1	函数概述	94
7.2	函数的定义	96
7.2.1	函数的定义形式	96
7.2.2	函数参数的说明	97
7.2.3	函数定义举例	97
7.3	函数的参数和函数的返回值	98
7.3.1	函数的形式参数和实际参数	98
7.3.2	函数的返回值	99
7.4	函数的调用与声明	99
7.4.1	函数调用的形式	100
7.4.2	函数调用的方式	100
7.4.3	函数的声明	100

7.5 函数的参数传送	101
7.5.1 实参与形参之间的数据单向传送	102
7.5.2 数组作参数	103
7.6 函数的嵌套调用和递归调用	106
7.6.1 函数的嵌套调用	106
7.6.2 函数的递归调用	107
7.7 内部变量和外部变量	111
7.7.1 内部变量	111
7.7.2 外部变量	112
7.8 变量的存储类别	113
7.8.1 变量的存储类别概述	113
7.8.2 内部变量的存储类别	114
7.8.3 外部变量的存储类别	116
7.9 内部函数和外部函数	117
7.9.1 内部函数	117
7.9.2 外部函数	118
7.10 程序举例	118
习题	120
第 8 章 编译预处理	126
8.1 文件包含	126
8.2 宏定义	127
8.2.1 不带参数的宏定义	127
8.2.2 带参数的宏定义	128
8.3 条件编译	129
8.3.1 #if	129
8.3.2 #ifdef 和 #ifndef	130
习题	131
第 9 章 指针	134
9.1 指针和地址	134
9.2 指针变量和指针变量的赋值	135
9.2.1 指针变量	135
9.2.2 指针变量的赋值	136
9.3 指针和函数参数	137
9.4 指针和数组	139
9.4.1 一维数组和数组元素的地址	139
9.4.2 通过一维数组名和指针引用数组元素	140
9.4.3 引用一维数组元素的方法总结	142
9.5 地址运算	142
9.5.1 指针与整数相加、减运算	142

9.5.2	指针赋值	144
9.5.3	同一数组中各元素地址间的关系运算与相减运算	144
9.6	字符指针和有关函数	145
9.7	指针数组和指向指针的指针	147
9.7.1	指针数组的定义	148
9.7.2	指针数组的初始化	148
9.7.3	字符指针数组	150
9.7.4	指向数组的指针	150
9.7.5	命令行参数	151
9.8	指针与函数	152
9.8.1	指针作为函数参数	152
9.8.2	指针作为函数的返回值	153
9.8.3	指向函数的指针	154
	习题	155
第 10 章	结构体与共用体	162
10.1	结构体	162
10.1.1	结构体类型的定义	162
10.1.2	结构体变量与指向结构体变量的指针变量的定义	163
10.1.3	结构体变量的引用	164
10.1.4	结构体数组	165
10.1.5	结构体变量作为函数的参数	168
10.2	链表	172
10.2.1	内存分配函数和回收函数	172
10.2.2	用指针和结构体构成链表	174
10.2.3	单向链表的建立	175
10.2.4	对链表的删除操作	178
10.2.5	对链表的插入操作	179
10.3	共用体	180
10.3.1	共用体类型的定义	180
10.3.2	共用体变量的定义	181
10.3.3	共用体变量的引用	181
10.3.4	共用体类型数据在内存中的存储	182
10.4	枚举类型	184
10.4.1	枚举类型的定义	184
10.4.2	枚举类型变量的定义和引用	185
10.5	typedef 的应用	186
10.5.1	用 typedef 对基本类型定义别名	186
10.5.2	用 typedef 对构造类型定义别名	187
10.5.3	typedef 的应用举例	188

习题	189
第 11 章 位运算	192
11.1 位运算符和位运算	192
11.1.1 &(按位与运算符).....	192
11.1.2 (按位或运算符).....	193
11.1.3 ^(按位异或运算符).....	193
11.1.4 ~(按位取反运算符).....	193
11.1.5 <<(左移运算符).....	194
11.1.6 >>(右移运算符).....	194
11.1.7 位运算符的优先级和结合性	194
11.1.8 复合位运算赋值运算符	195
11.1.9 位运算的应用	195
11.2 位运算举例	196
习题	199
第 12 章 文件	201
12.1 文件概述	201
12.1.1 ASCII 码文件与二进制文件.....	201
12.1.2 缓冲文件系统	202
12.1.3 文件型指针	202
12.2 文件的打开与关闭	203
12.2.1 文件的打开	203
12.2.2 文件的关闭	205
12.3 文件的读/写操作	205
12.3.1 fputc 函数与 fgetc 函数的使用	206
12.3.2 fgets 函数与 fputs 函数的使用	209
12.3.3 fscanf 函数与 fprintf 函数的使用	210
12.3.4 按数据块方式读/写——fwrite 函数与 fread 函数	212
12.4 文件的定位	214
习题	216
附录 1 ASCII 码表	219
附录 2 C 语言关键字	220
附录 3 运算符的优先级和结合性	221
参考文献	222

第1章 C语言概述

本章对C语言的发展历史、特点作一个简单介绍,并通过简单的例子说明C语言的执行过程。

1.1 C语言简介

1.1.1 程序设计语言的概念

程序设计语言(Programming Language)是用于编写计算机程序的语言。那么什么是程序呢?程序是计算机指令的有序集合;而指令就是命令,用于指引计算机完成某种操作。无论是自然语言(人们用于讲话的语言,如汉语、英语等)还是形式语言(人们为具体目的而设计的语言,如化学家用一种形式语言表示分子的化学结构),其构成基础都是语法、规则和语义。计算机语言是用于表示运算(computation)的形式语言,这里的运算不仅是指简单的算术运算,还指计算机处理各种逻辑的过程。计算机语言的种类很多,如高级语言是类似于人类自然语言的一种计算机语言,C语言就是一种高级语言;汇编语言是用助记符(mnemonic)形式表示的一种语言,如Intel 8086汇编语言;机器语言则是用二进制位串表示的一种语言,是目标机器(执行用户程序的中央处理器)能够直接识别的一种语言。

1.1.2 C语言的发展历史

C语言是由美国电话电报公司(AT&T)贝尔(Bell)实验室的Dennis Ritchie开发设计,并首次在一台使用UNIX操作系统的DEC PDP-11小型计算机上实现的。其很多原理和思想来自于一种称为B语言的计算机语言,而B语言的主要思想又来自于BCPL(Basic Combined Programming Language)语言(Dennis Ritchie提出)和CPL(Combined Programming Language)语言。1970年,Bell实验室的Ken Thompson(UNIX操作系统的设计者)在UNIX操作系统上实现了B语言。1972年,Ken Thompson的合作者Dennis Ritchie把BCPL的某些一般特性放到B语言的开发设计过程中,就形成了现在的C语言。

1.1.3 C语言的特点

(1) C语言是一种结构化程序设计语言。结构化语言具有层次清晰、易于调试和维护等特点,许多高级语言是结构化语言,如PASCAL语言、ADA语言等。

(2) C语言的语句简练,书写灵活,具有关键字较少、类型说明符书写简单等特点。

(3) C语言处理能力强。C语言可以通过指针对地址进行运算,通过指针增强了构造型变量,特别是数组和结构的存取能力;具有二进制位运算和寄存器存取功能;运算符丰富。

(4) C语言具有很好的移植性。所谓移植性很好,是指对编写的代码作少量的修改就可

以转移到不同类型的机器上运行。这个特性是 C 语言取得成功的一个很重要的原因。比如在嵌入式系统开发中，C 语言也得到广泛的应用。

(5) C 语言是一种高效率的语言。所谓效率，可以从程序的执行速度和占用的存储空间两方面来衡量，对同一个问题(或者叫做运算)而言，用汇编语言实现和用 C 语言实现在速度和存储空间上相差大约 20%。

1.2 认识 C 语言程序

1.2.1 从 Hello World 程序开始

下面的程序是完成在控制台上输出“Hello, World!”字符串的完整的 C 语言程序：

```
#include<stdio.h>
main()
{
    printf("Hello,World!\n");
}
```

要让这个程序在目标机器上运行，当然还要做点其它工作，比如说“编译”这样的工作，所用的操作系统不同，编译工具(编译程序)也有所不同。具体让该程序如何运行是我们下节要介绍的内容。先让我们来看看这个程序是如何组成的。

1.2.2 C 语言程序的组成

C 语言程序主要由函数和变量组成，一个函数是由用于说明某个运算的语句和变量组成的，语句以分号结束，变量用来存储运算过程中所需的值以及运算过程中产生的中间值。C 语言函数类似于 PASCAL 语言中的过程和函数。“Hello World”程序中我们的函数命名为 main，一般来说，可以用合法形式任意命名一个函数，但 main 函数是一个特殊的函数，是程序运行的开始点，这意味着一个程序有且仅有一个 main 函数。

main 函数常常要调用其它函数来协助完成它的工作，这些函数既可以自己来编写，也可以从函数库中得到。“Hello World”程序中的第一行：

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

它是一条编译预处理指令，目的是告诉编译器在 main 函数中包含了标准输入/输出库函数(printf 函数)。标准输入/输出库函数是以库的形式提供的，而这些库函数的声明放在一个叫 stdio.h 的头文件中，在生成目标程序(目标机器上能直接运行的程序)时会自动将该函数的代码链接到你的程序中。调用函数和被调用函数之间通过参数来交换数据，把参数放在函数名之后的一对括号()之中，如果有多个参数，用逗号分隔构成一个参数列表。在我们的例子中，main 函数中调用了 printf 函数，main 是调用者，printf 是被调用者，main 向 printf 传递了一个参数，这是一个字符串参数“Hello,World!\n”。

用花括号括起来的部分是函数的语句，称为函数体。我们例子中的 main 函数只有一条语句：

```
printf("Hello,World!\n");
```

这条语句是调用标准输出函数 `printf`，输出字符串 “Hello,World!\n”。

在这个简单的程序中没有用到变量。

1.3 C语言程序的执行过程

1.3.1 从编写C语言程序到执行该程序的完整过程

从编写一个C语言程序到在目标机器上执行该程序一般需要经过以下几个过程。

1. 用文本编辑器编辑C语言源程序

可以用任何文本编辑器编写C语言源程序，只要可以生成纯文本(无格式信息的ASCII码文本)文档就可以。本书中的例子使用 Visual C++6.0 IDE 集成的编辑器来生成C语言源程序，一般其扩展名为 .C。

例如：在C盘上建立一个文件夹并命名为 `example`，在该文件夹下建立一个头文件 `example.h` 和一个C语言源文件 `example.c`。

`example.h` 文件的内容如下：

```
float maxium(float a,float b)           //用户自定义函数，放在一个头文件中
{
    return a>b?a:b;
}
```

`example.c` 文件的内容如下：

```
#include<stdio.h>
#include "example.h"                   //包含 example.h 头文件
int main()                             //主函数
{
    float x,y;
    printf("please input x,y:");
    scanf("%f,%f",&x,&y);
    printf("\nmaxium(%f,%f)=%f\n",x,y,maxium(x,y));
    return 0;
}
```

上面程序的功能是实现在两个实数中求最大的数，为说明问题，我们把求最大数的功能用一个用户自定义的函数(maxium)来实现，并把该函数放在一个头文件(example.h)中，而结果的输出在主函数中实现。

2. 编译源程序

在 Visual C++6.0 中建立一个 C++ 空工程，命名为 `example`，然后把 `example.c` 文件加入到该工程中，之后用 Visual C++6.0 的“建立(Build)”菜单中的编译命令对 `example.c` 文件进行编译，无错误后再用“建立(Build)”命令建立可执行文件。

3. 运行程序

上一步中如果无错误，再选择“建立(Build)”菜单中的“执行(execute)”命令，执行该

程序，观察运行结果是否正确。

通过上面的完整例子，现在应该知道怎么让“Hello World!”程序执行了吧？请自己试一试。

1.3.2 计算机是如何执行程序的

计算机硬件系统主要由五大部件组成，计算机程序主要是由 CPU 执行的，CPU 对指令进行解释执行之前要从内存中存取二进制比特串。显然，我们用高级语言编写的程序(源程序)不可能直接传送到内存中以供 CPU 读取执行，要做相应的转换后才能装入内存，这个转换过程就称为编译。编译是指通过编译程序(另外一个程序)把用高级语言编写的程序翻译成目标机器能够执行的机器指令(不同系列的计算机其机器指令不相同)，这些机器指令被存放在一个叫做目标程序的文件中，以二进制文件的形式(在 Visual C++ 中其扩展名为 .obj, Visual C++6.0 中的编译程序为 CL.EXE)存储在磁盘上。生成的目标程序还不能直接装入目标机器中，还需要用连接程序(LINK.EXE)将目标程序和其它有关的模块进行连接，并进行一些必要的地址变换，生成一个可装载到目标机器上(由操作系统装载到目标机器上)的可执行文件(扩展名为.EXE)，然后才能执行。

1.4 C 语言程序的上机步骤

1.4.1 C 语言程序上机环境介绍

操作系统不同，C 语言程序的上机环境也有差别。以 Windows 操作系统为例，可以用 Turbo C、Visual C++ 等编程工具，这些编程工具是把高级语言程序设计的一些必要工具做成一个集成开发环境(IDE)，所用到的工具以菜单命令的形式提供给程序员，这样就大大地简化了程序的设计和调试，提高了编程效率。本书选用比较流行的 Turbo C 和 Visual C++6.0 作为 C 语言的编程工具来讲解例子程序，有关这两个编程工具的使用请参考相关手册。

1.4.2 C 语言程序上机步骤

以 Visual C++6.0 为例，C 语言程序的上机步骤如下：

- (1) 在磁盘上建立一个自己的文件夹用来存放工程文件。
- (2) 建立一个空工程，并用 Visual C++6.0 的编辑程序编写有关的文件(头文件、C 程序源文件等)。把建立的文件存放在该工程对应的文件夹下。
- (3) 把 C 程序源文件添加到工程中。
- (4) 编译、连接、调试、执行。

习 题

- (1) 什么是程序？简述计算机高级语言和人类自然语言的异同。
- (2) 简述计算机执行程序的过程。
- (3) 用 Visual C++6.0 或 Turbo C 调试运行 1.3 节中的例子程序。把与程序有关的文件放在自己的文件夹下。

第2章 数据类型、运算符与表达式

数据是程序处理的对象，程序的主要任务就是对数据进行加工、处理，而计算机中的数据包括数值型数据和非数值型数据。数据类型是按照数据的性质、表示形式和占据存储空间的大小以及构造特点来划分的。数据类型系统是一种抽象机制，它将具有相同性质的数据抽象成同一种类型的数据。

运算是对于数据进行加工的过程；用来表示各种不同运算的符号称为运算符；参加运算的数据称为运算量（也称运算对象或操作数）。

表达式就是用运算符将运算量连接起来的、符合C语言语法规则的式子。

本章主要介绍C语言的基本数据类型和算术运算、关系运算、逻辑运算等基本运算及其表达式。

2.1 字符集和标识符

2.1.1 字符集

字符集是高级语言的编译系统所能识别的字母、数字和特殊符号。每种高级语言都有自己特定的字符集。

C语言的字符集包括：

- 大、小写英文字母：A, B, ..., Z, a, b, ..., z
- 数字：0, 1, 2, ..., 9
- 运算符：+ - * / % > < = & | ? ! ^ ~ ,
- 括号：() {} []
- 标点符号：' " : ;
- 特殊符号：\ _ \$ #
- 空白符：空格符、换行符、制表符

C语言字符集中的字符按照C语言语法组合起来，就可以通过C语言编译系统的语法和词法来分析了。不在C语言字符集中的字符可以在C语言的字符串中出现。

2.1.2 标识符

标识符是用来标识C程序中的变量、常量(指符号常量)、函数、标号及用户定义对象的名称的。标识符的首字符必须是字母或下划线，后跟字母、数字或下划线，其有效长度为1~32个字符。以下列举了几个正确和错误的标识符：

正确	错误
count	1count
Test23	Hi!there
High_balance	High..blance

在 C 语言中，标识符中字母大小写是有区别的，所以 count、Count 和 COUNT 是三个不同的标识符。

C 语言中的标识符分为系统定义标识符和用户定义标识符。

2.1.3 系统定义标识符

系统定义标识符是指具有固定名字和特定含义的标识符，分为关键字和预定义标识符两种类型。

1. 关键字

关键字也称保留字，是 C 语言规定的具有特定含义的标识符。ANSI C 共有 32 个关键字，分为以下几类：

1) 数据类型说明符

int, long, short, char, float, double, signed, unsigned, struct, union, enum, void, volatile, const, typedef。

2) 存储类别说明符

auto, static, register, extern。

3) 语句命令字

goto, return, break, continue, if, else, while, do, for, switch, case, default。

4) 运算符

sizeof。

2. 预定义标识符

预定义标识符是具有特殊含义的标识符，包括系统标准库函数名和编译预处理命令。C 语言中的预定义标识符有很多，以下列出常见的几种：

1) 标准库函数

输入输出函数：scanf, printf, getchar, putchar, gets, puts 等。

数学函数：sqrt, fabs, sin, cos, pow 等。

2) 编译预处理命令

define, endif, include, ifdef, ifndef, line 等。

2.1.4 用户定义标识符

用户定义标识符是用户按照 C 语言语法规则定义的标识符，用于标识用户使用的变量、符号常量、数据类型和函数等操作对象，也就是用户在程序设计时为变量、常量和函数起的名字。

使用用户定义标识符时需要注意：

(1) 用户定义标识符不能使用关键字和预定义标识符。

- (2) 用户定义标识符为了标识不同的对象，标识符的前8个字符要有区别。
- (3) 用户定义标识符最好按义取名，即根据它所代表的操作对象的含义取名。
- (4) 避免使用容易混淆的字符，如1与l，o与0，z与2等。
- (5) 大、小写代表不同的意义。
- (6) 尽量不要与某个库函数同名。

2.2 常量和变量

2.2.1 常量

在 C 语言中，常量是指在程序执行的过程中其值不能改变的量。常量分为直接常量和符号常量。直接常量不需要类型说明就可以直接使用，其类型是由常量本身隐含决定的。符号常量是为了增加程序的可读性和易修改性而经常使用的另外一种形式的常量。

1. 直接常量

直接常量有数值常量、字符常量和字符串常量三类。

1) 数值常量

C 语言中的数值常量分为整型常量和实型常量。

● 整型常量

十进制整数：12、-345、0 等。十进制整数的每个数字位可以是 0~9。

八进制整数：012、-023、0345 等。八进制整数以 0 开头，其后的每个数字位可以是 0~7。

十六进制整数：0x12、-0x23、0x345 等。十六进制整数以 0x 开头，其后每个数字位可以是 0~9 和 A~F。

● 实型常量

一般形式：12.45、-0.567、3.1415926 等。

指数形式：23.45e4、0.38E-4 等。

2) 字符常量

字符常量是由一对单引号括起来的一个字符，例如'A'、'a'、'3'、'+','!'等。

3) 字符串常量

字符串常量是由一对双引号括起来的一个字符序列，例如"Hello"、"How are you?"、"a+b="等。字符串中可以包含字母、数字和各种专门的字符。

2. 符号常量

符号常量定义的一般形式为

```
#define 符号常量标识符 直接常量
```

例如：

```
#define MAX 100
```

```
#define MIN 10
```

```
#define PI 3.1415926
```

其中，#define 是一条预处理命令，称为宏定义命令，其功能是将符号常量标识符定义为其后的直接常量。一经定义，以后做预处理时，凡是碰到该符号常量标识符，都将替换为相