

C语言程序设计(第2章数据类型、运算符和表达式) PDF转换
可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/134/2021_2022_C_E8_AF_AD_E8_A8_80_E7_A8_8B_c97_134888.htm 2.1 C语言的数据类型
2.2 常量与变量 2.2.1 标识符命名 2.2.2 常量 2.2.3 变量 2.3 整型数据 2.3.1 整型常量 2.3.2 整型变量 2.4 实型数据 2.4.1 实型常量 2.4.2 实型变量 2.5 字符型数据 2.5.1 字符常量 2.5.2 字符串常量 2.5.3 转义字符 2.5.4 符号常量 2.5.5 字符变量 2.6 运算符 2.6.1 算术运算符 2.6.2 自增和自减 2.6.3 关系和逻辑运算符

2.1 C语言的数据类型 C语言有五种基本数据类型：字符、整型、单精度实型、双精度实型和空类型。尽管这几种类型数据的长度和范围随处理器的类型和C语言编译程序的实现而异，但以bit为例，整数与CPU字长相等，一个字符通常为一个字节，浮点值的确切格式则根据实现而定。对于多数微机，表2-1给出了五种数据的长度和范围。表2-1 基本类型的字长和范围

类型	长度(bit)	范围
char(字符型)	8	0~255
int(整型)	16-32	-32768~32767
float(单精度型)	32	约精确到6位数
double(双精度型)	64	约精确到12位数
void(空值)	0	无值

表中的长度和范围的取值是假定CPU的字长为16 bit。C语言还提供了几种聚合类型(aggregate types)，包括数组、指针、结构、共用体(联合)、位域和枚举。这些复杂类型在以后的章节中讨论。除void类型外，基本类型的前面可以有各种修饰符。修饰符用来改变基本类型的意义，以便更准确地适应各种情况的需求。修饰符如下：#8226. unsigned(无符号)。#8226. short(短型符)。修饰符signed、short、long和unsigned适用于字符和整数两种基本类型，而long还可用于double(注意，由

于long float与double意思相同，所以ANSI标准删除了多余的long float)。表2-2给出所有根据ANSI标准而组合的类型、字宽和范围。切记，在计算机字长大于16位的系统中，short int与signed char可能不等。表2-2 ANSI标准中的数据类型

类型	长度(bit)	范围
char(字符型)	8	ASCII字符
unsigned char(无符号字符)	8	0~255
signed char(有符号字符型)	8	-128~127
int(整型)	16-32	768~32767
unsigned int(无符号整型)	16-32	0~65535
signed int(有符号整型)	16	同 int
short(短整型)	8-16	-128~127
unsigned short(无符号短整型)	8-16	0~255
signed short(有符号短整型)	8	同short int
long(长整型)	32-64	-2147483648~2147483647
signed long(有符号长整型)	32-64	-2147483648~2147483647
unsigned long(无符号长整型)	32-64	0~4294967296
float(单精度型)	32	约精确到6位数
double(双精度型)	64	约精确到12位数

*表中的长度和范围的取值是假定CPU的字长为16 bit。因为整数的缺省定义是有符号数，所以signed这一用法是多余的，但仍允许使用。某些实现允许将unsigned用于浮点型，如unsigned double。但这一用法降低了程序的可移植性，故建议一般不要采用。为了方便，C编译程序允许使用整型的简写形式：
 #include <limits.h>
 long int 简写为long。
 unsigned int 简写为unsigned。
 unsigned long int 简写为unsigned long。即，int可缺省。

2.2 常量与变量

2.2.1 标识符命名

在C语言中，标识符是对变量、函数标号和其它各种用户定义对象的命名。标识符的长度可以是一个或多个字符。绝大多数情况下，标识符的第一个字符必须是字母或下划线，随后的字符必须是字母、数字或下划线（某些C语言编译器可能不允许下划线作为标识符的起始字符）。下面是一些正确或错误标识符命名的实例。正确形

式 错误形式 count 2count 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com