

计算机等级考试二级 C 各章内容摘要2 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/133/2021_2022__E8_AE_A1_E7_AE_97_E6_9C_BA_E7_c97_133738.htm

第2章数据类型、运算符和表达式【考点一】C语言的数据类型C中的数据类型大体可以分成三类，它们是：基本类型、导出类型和用户定义类型。基本类型是由C语言预先定义好的数据类型，它包括：逻辑型、字符型、整型、浮点型和空值型。导出类型是由某种已知类型构造出来的数据类型，它包括：数组、指针和引用。用户定义类型是指必须由用户给出具体定义后才能使用的数据类型，它包括：枚举类型和类类型。

1.基本类型

(1) 逻辑型逻辑型用关键字bool表示，因此又称为bool型(布尔型)。逻辑型的取值范围只包含true和false，它们都是C关键字。

(2) 字符型] 字符型用关键字char表示，因此又称为char型，即把字符作为一种数据进行操作。字符型的取值范围是全部基本字符以及ASCII码集或扩充ASCII码集对应的全部符号。

(3) 整型整型用关键字int表示，因此又称为int型，它是C中最常用的数据类型，整型数据的取值范围是计算机所能表示的所有整数。

(4) 浮点型浮点型包括单精度型和双精度型两种，单精度型用关键字float表示，又称float型；双精度型用关键字double表示，又称double型。它们对应着数学中实数的概念，即带有小数点的数。浮点型数据的取值范围原则上是任意大小和精度的小数，但实际上不可能是任意大小的，而是有一定范围的。C中的浮点数可以采用尾数加阶码的表示形式，表示的浮点数范围是：单精度型-3.4E38~3.4E38，但不能超过7位有效数字；双精度

型-1.7977E308~1.7977E308，但不能超过15位有效数字。(5)空值型空值型用关键字void表示，因此又称为void型。空值型的取值为空。C中不存在void型的常量或变量。C语言规定，所有函数说明都必须指明返回值类型，没有返回值的函数应说明为void类型的函数；另外，void类型还可以用来声明一个指针变量。

2.基本类型的派生类型

C语言中，基本类型的字长及其取值范围可以放大和缩小，改变后的类型就叫做基本类型的派生类型。派生类型声明符是由基本类型关键字char、int、float、double前面加上类型修饰符组成的。类型修饰符包括：short短类型，缩短字长；long长类型，加长字长；signed有符号类型，取值范围包括正负值；unsigned无符号类型，取值范围只包括正值；

【考点二】常量常量是指在程序执行过程中值不改变的量。常量有两种表示形式，即字面常量和符号常量。字面常量的类型是根据书写形式来区分的，它们的类型分别为：整型、浮点型、字符型、字符串型，每个字面常量的字面本身就是它的值。符号常量是一个标识符，在程序中必须遵循“先声明，后使用”的原则。

1.逻辑常量

逻辑常量主要用在逻辑运算中，此外，由于逻辑值对应着整数1或0，所以它也能够像其他整数一样出现在表达式中，参与各种整数运算。

2.字符常量

字符常量简称字符，它以单引号作为起止符号，中间有一个或若干个字符。这是一般意义上的字符常量；除此之外还有一种特殊形式的字符常量，以“\”开头的包括多个字符的字符序列也是符合规定的字符常量。但是，无论字符常量包含一个还是多个字符，每个字符常量只能表示一个字符，当字符常量的一对单引号内多于一个字符时，则将按照一定的规则解释为一个字符。计算机中

常用的ASCII字符也是字符型的数据，因为ASCII码值在0~127之间，正好落在字符型数据的取值范围之内。ASCII字符集中的每一个显示字符(个别字符除外)都可以作为一个字符常量。但是，对于一些特殊的字符，如回车、换行等具有控制功能的字符，或者像单引号、双引号等作为特殊标记使用的字符，就无法直接采用单引号作为起止符号来表示。C中引入了"转义"字符的概念，其含义是：用反斜线"\\"引导的下一个字符失去了原来的含义，而转义为具有某种控制功能的字符。另外，还允许用反斜线引导一个具有1至3位的八进制整数或一个以字母x(大、小写均可)作为开始标记的具有1至2位的十六进制整数，对应的字符就是以这个整数作为ASCII码的字符。

3. 整型常量

整型常量就是整型常数，简称整数，它有十进制、八进制和十六进制3种表示。

(1) 十进制整数

十进制整数以正号(+)或负号(-)开头，由首位非0的一串十进制数字组成。若以正号开头则为正数，若以负号开头则为负数，若省略正负号，则默认为正数。当一个十进制整数大于等于-2 147 483 648(即-2³¹)，同时小于等于2 147 483 647(即2³¹-1)时，则被系统看作是整型常量；当在2 147 483 648~4 294 967 295即2³²-1范围之内时，则被看作是无符号整型常量；当超过上述两个范围时，则无法用C整数类型表示，只有把它用实数(即带小数点的数)表示才能够被有效地存储和处理。

(2) 八进制整数

八进制整数以数字0开头，后面接若干个八进制数字(借用十进制数字中的0~7)。八进制整数前面不带正负号，全部默认为正数。当一个八进制整数大于等于0同时小于等于017777777777时，则称为整型常量，当大于等于020000000000同时小于等于037777777777时，则称为无符号

整型数量。不要使用超过上述两个范围的八进制整数，因为没有与此相对应的C 整数类型。(3)十六进制整数十六进制整数以数字0和字母x(大、小写均可)开头，后面接若干个十六进制数字(借用十进制数字0~9，字母A~F或a~f)。当一个十六进制整数大于等于0同时小于等于0x7FFFFFFF时，则称为整型常量，当大于等于0x80000000同时小于等于0xFFFFFFFF时，则称为无符号整型常量，与八进制整数类似，超过上述两个范围的十六进制整数也没有与之相对应的C 整数类型，所以也不能使用它们。C 中的整型数据除了一般表示方法之外，还允许给它们添加后缀u或l。对于任一种进制的整数，若后缀为字母u(大、小写均可)，则规定它为一个无符号整型(unsigned int)数，若后缀为字母l(大、小写均可)，则规定它为一个长整型(long int)数。在一个整数的末尾，可以同时使用u和l，并且对排列无要求。

4.实型常量

实型常量有十进制的定点和浮点两种表示方法，但并不存在其他进制的表示，下面分别介绍这两种表示方法。

(1)定点表示 定点表示的实数简称定点数，实际上是以小数形式表示实数。定点表示的实数是由一个正号或负号(正号可以省略)后接若干个十进制数字和一个小数点所组成，这个小数点可以处在任何一个数字位的前面或后面。

(2)浮点表示 浮点表示的实数简称浮点数，实际上是以指数形式表示实数。浮点表示的实数是由一个十进制整数或定点数后接一个字母E(大、小写均可)和一个1至3位的十进制整数所组成，字母E之前的部分称为该浮点数的尾数，之后的部分成为该浮点数的指数，该浮点数的值就是它的尾数乘以10的指数幂。对于一个浮点数，若将它尾数中的小数点调整到最左边第一个非零数字的后面，则称它为规格化(

或标准化)浮点数。实型常量分为单精度(float)、双精度(double)和长双精度(long double)三类。一般float型数据在内存中占4个字节，提供7位有效数字；double型数据占8个字节，提供15位有效数字；long double型数据占10个字节，提供19位有效数字。对于一个定点数或浮点数，C自动按一个双精度数(double型)来存储。若在一个定点数或浮点数之后加上字母F(大、小写均可)，则自动按一个单精度数来存储。

5. 枚举常量

枚举常量是枚举类型中定义的值，即枚举植。枚举类型属于用户定义类型，用关键字enum表示，因此又称为enum类型。用户通常利用枚举类型定义程序中需要使用的一组相关符号常量。声明枚举类型的语法格式为：`enum <类型名> { <枚举值表> }`；其中，enum是关键字，指出要定义的是枚举类型。<类型名>是标识符，即由用户给出的具体枚举类型名。<枚举值表>包含多个枚举值，它们用逗号分隔开，每个枚举值就是一个枚举常量。枚举值有两种定义形式：一是<值名>；二是<值名> = <整型常量>。关于enum类型有四点说明：(1)一个enum类型实际上是int类型的一个子集，其每一个枚举值代表一个整数。(2)n个枚举值全部未赋常量值时，它们自左至右分别与整数0, 1...n-1对应。(3)若第i个枚举值赋常量值为m，则其未赋常量值的后续枚举值分别与整数m 1, m 2...对应，直到下一个赋了值的枚举值或结束。因此，为枚举值所赋的整型常量值应从左至右递增。(4)枚举类型的声明也可作为成组声明若干整型符号常量的方法。也就是说，把每个要声明的符号常量作为一个枚举值，将各个枚举值合在一起定义成一个枚举类型。由于枚举类型属于用户定义数据类型，一旦枚举类型被定义后就可以使用它来定

义变量。【考点三】变量1.变量的定义变量的定义是通过变量声明语句来实现的，变量声明语句的一般格式为：[<存储类>] <类型名> <变量名> [= <初值表达式>] , ... ; <存储类>有四种，它们分别是auto、register、static、extern。<类型名>为已存在的一种数据类型名称，如char，short，int，long，float，double等基本数据类型名，或者用户定义的数据类型名。<变量名>是用户定义的一个标识符，用来表示一个变量，该变量可以通过后面的可选项赋予一个值，称为给变量赋初值，也叫做对变量进行初始化。C中标识符是区分大小写的，也就是说，大写字母和小写字母被认为是不同的字母。变量名的命名遵循如下规则：(1)不能是C关键字；(2)第一个字符必须是字母或下划线；(3)中间不能有空格；(4)变量名中不能包括；， -之类的特殊符号。实际上变量名中除了能使用26个英文大小写字母和数字外，只能使用下划线"_"。

2.变量的使用方式(1)全局变量和局部变量全局变量是在所有函数定义、类定义和程序块之外声明的变量。声明全局变量时如果在程序中不对它进行专门的初始化，该变量会被系统自动初始化为0。在程序的任何一个函数、类或程序块之内均可以访问全局变量。局部变量是在某个函数定义、类定义或程序块之内声明的变量。局部变量只能在声明它的函数、类或程序块中被访问。(2)生存期与作用域生存期是指从一个变量被声明且分配了内存开始，直到该变量声明语句失效，它占用的内存空间被释放为止。一个全局变量的生存期从它被声明开始，直到程序结束；一个局部变量的生存期从它被声明开始，直到包含它的最近的一个程序块结束。作用域是指变量名可以代表该变量存储空间的使用

范围。一般情况下，变量的作用域与其生存期一致，但由于C语言允许在程序的不同部分为不同变量取同一名字，因此一个变量名的作用域可能小于其生存期。（3）变量的存储类属性在C中变量还可以按存储分配方式的不同被划分为4种不同的存储类别，它们分别是：

auto变量：用关键字auto声明的局部变量称为自动变量。auto为变量声明时的默认存储类别，即在变量定义时，如果不显式标明存储类别，则系统自动按auto变量处理。auto变量所占用存储空间的分配和释放工作将由系统自动完成。

register变量：用关键字register声明的局部变量称为寄存器变量。register变量可能以寄存器作为其存储空间。声明寄存器变量时，关键字register的作用只能是建议（而不是强制）系统使用寄存器，原因是寄存器虽然存取速度快，但空间有限，当寄存器不够用时，该变量仍然按自动变量处理。

static变量：用关键字static声明的变量称为静态变量。任何静态变量的生存期将延续到整个程序的终止。与全局变量一样，为静态变量分配的存储空间在整个程序运行过程中不再被释放；如果静态变量未被赋初值，系统将自动为其赋初值为0。

extern变量：用关键字extern声明的变量称为外部变量。变量一旦被声明为外部变量，系统就不必像一般变量那样为其分配内存，因为该变量已在这一局部的外面被定义。外部变量一般用于多个文件组成的程序中，有些变量在多个文件中被声明，但却是指同一变量。标明某一变量为外部变量可以避免为其重复分配内存。

（4）typedef类型说明使用关键字typedef可以为已有类型名定义一个新类型名。其语法格式为：`typedef <已有类型名> <新类型名>`

> typedef类型说明并没有真正地定义新的数据类型，它只是

相当于给某个已有的数据类型起了一个别名。在规模较大的程序中为了提高代码可读性常采用这种形式。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