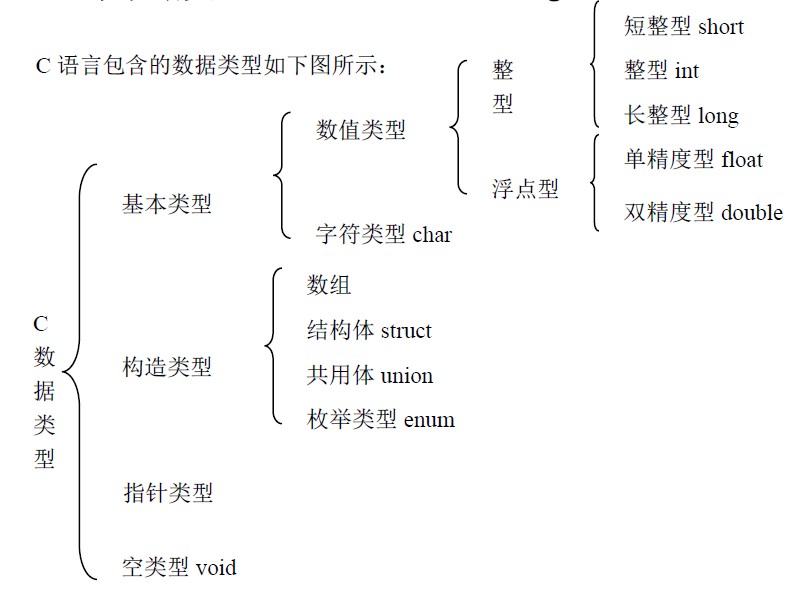
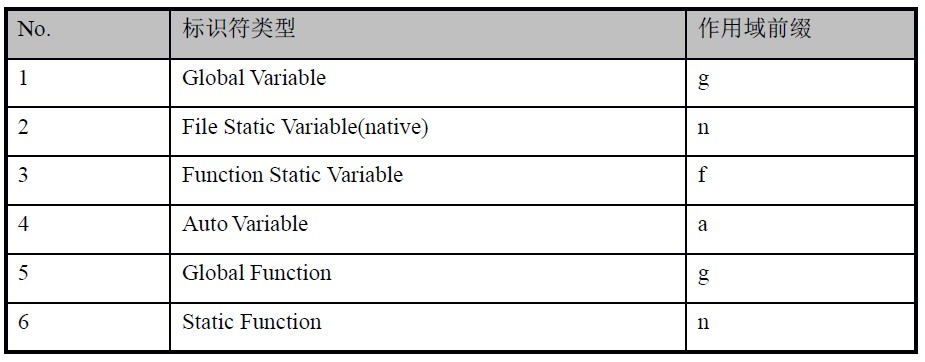
C 语言包含的数据类型如下图所示：



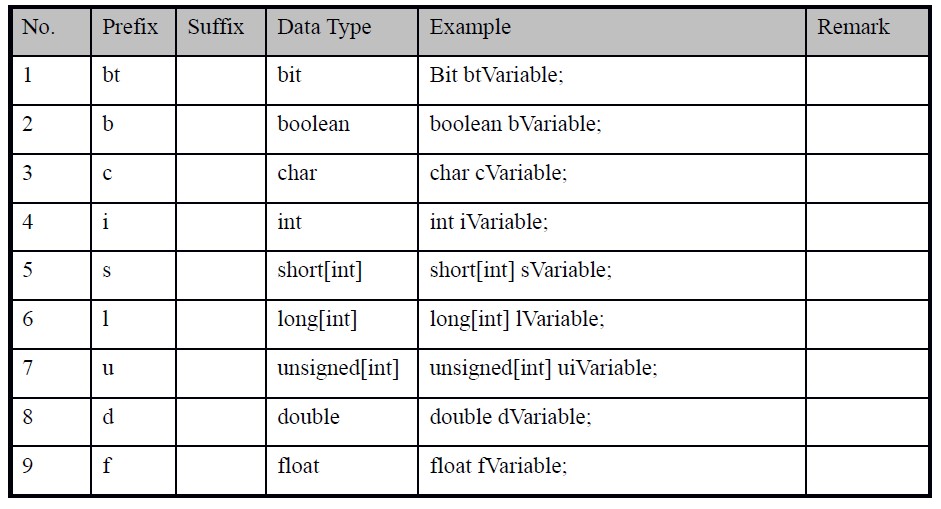
一、数据类型与“模子”  
short、int、long、char、float、double 这六个关键字代表C 语言里的六种基本数据类型。  
  
怎么去理解它们呢? 举个例子：见过藕煤球的那个东西吧？(没见过？煤球总见过吧)。那个东西叫藕煤器，拿着它在和好的煤堆里这么一咔，一个煤球出来了。半径12cm，12 个孔。不同型号的藕煤器咔出来的煤球大小不一样，孔数也不一样。这个藕煤器其实就是个模子。  
  
现在我们联想一下，short、int、long、char、float、double 这六个东东是不是很像不同类型的藕煤器啊？拿着它们在内存上咔咔咔，不同大小的内存就分配好了，当然别忘了给它们取个好听的名字。  
  
在32 位的系统上short 咔出来的内存大小是2 个byte；  
int 咔出来的内存大小是4 个byte；  
long 咔出来的内存大小是4 个byte；  
float 咔出来的内存大小是4 个byte；  
double 咔出来的内存大小是8 个byte；  
char 咔出来的内存大小是1 个byte。  
（注意这里指一般情况，可能不同的平台还会有所不同，具体平台可以用sizeof 关键字测试一下）  
  
很简单吧？咔咔咔很爽吧？是很简单，也确实很爽，但问题就是你咔出来这么多内存块，你总不能给他取名字叫做x1,x2,x3,x4,x5…或者长江1 号，长江2 号…吧。它们长得这么像(不是你家的老大，老二，老三…)，过一阵子你就会忘了到底哪个名字和哪个内存块匹配了(到底谁嫁给谁了啊？^\_^)。所以呢，给他们取一个好的名字绝对重要。下面我们就来研究研究取什么样的名字好。  
  
二、变量的命名规则  
**1、命名应当直观且可以拼读，可望文知意，便于记忆和阅读。**  
标识符最好采用英文单词或其组合，不允许使用拼音。程序中的英文单词一般不要太复杂，用词应当准确。  
  
**2、命名的长度应当符合“min-length && max-information”原则。**  
C 是一种简洁的语言, 命名也应该是简洁的。例如变量名MaxVal 就比MaxValueUntilOverflow 好用。标识符的长度一般不要过长，较长的单词可通过去掉“元音”形成缩写。  
  
另外，英文词尽量不缩写，特别是非常用专业名词，如果有缩写，在同一系统中对同一单词必须使用相同的表示法，并且注明其意思。  
  
**3、当标识符由多个词组成时，每个词的第一个字母大写，其余全部小写。**比如：  
   int CurrentVal；  
这样的名字看起来比较清晰，远比一长串字符好得多。  
  
**4、尽量避免名字中出现数字编号，如Value1,Value2 等，除非逻辑上的确需要编号。**比如驱动开发时为管脚命名，非编号名字反而不好。  
初学者总是喜欢用带编号的变量名或函数名，这样子看上去很简单方便，但其实是一颗颗定时炸弹。这个习惯初学者一定要改过来。  
  
**5、对在多个文件之间共同使用的全局变量或函数要加范围限定符(建议使用模块名(缩写)作为范围限定符)。**  
(GUI\_ ，etc)标识符的命名规则：  
  
6、**标识符名分为两部分：规范标识符前缀(后缀) + 含义标识。非全局变量可以不用使用范围限定符前缀。**

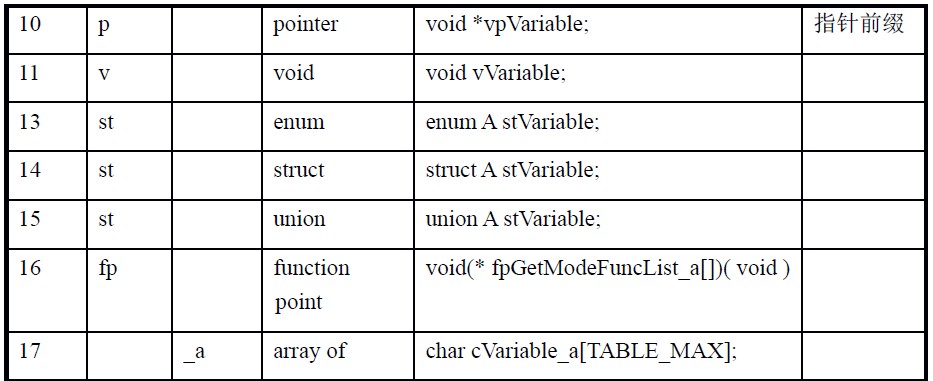


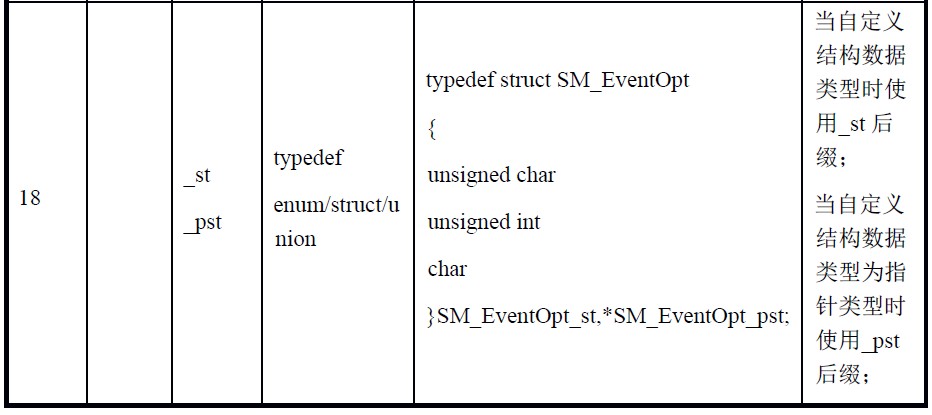
**7、作用域前缀命名规则。**



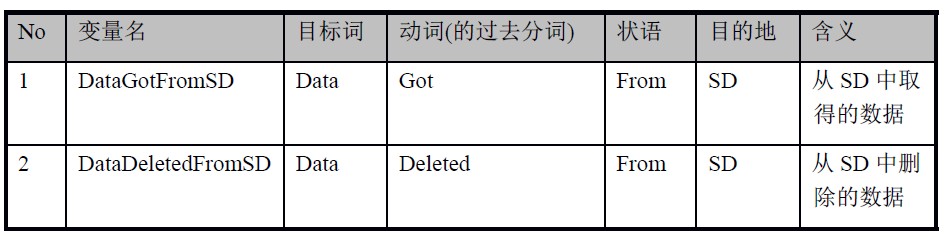
**8、数据类型前缀命名规则。**



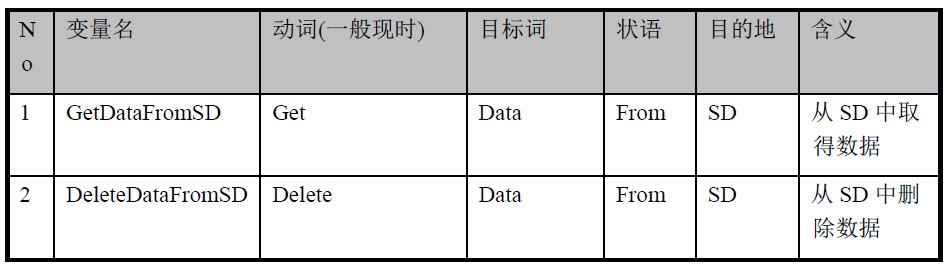




**9、含义标识命名规则，变量命名使用名词性词组，函数命名使用动词性词组。**例如：



变量含义标识符构成：目标词+ 动词(的过去分词)+ [状语] + [目的地]；



函数含义标识符构成：动词(一般现时)+目标词+[状语]+[目的地]；  
  
**10、程序中不得出现仅靠大小写区分的相似的标识符。**例如：  
   int x, X; 变量x 与X 容易混淆  
   void foo(int x); 函数foo 与FOO 容易混淆  
   void FOO(float x);  
这里还有一个要特别注意的就是1（数字1）和l（小写字母l）之间，0（数字0）和o（小写字母o）之间的区别。这两对真是很难区分的，我曾经的一个同事就被这个问题折腾了一次。  
  
**11、一个函数名禁止被用于其它之处。**例如：  
#include "c\_standards.h"  
void foo(int p\_1)  
{  
   int x = p\_1;  
}  
void static\_p(void)  
{  
   int foo = 1u;  
}  
  
**12、所有宏定义、枚举常数、只读变量全用大写字母命名，用下划线分割单词。**例如：  
const int MAX\_LENGTH = 100; //这不是常量，而是一个只读变量，具体请往后看  
   #define FILE\_PATH “/usr/tmp”  
  
**13、考虑到习惯性问题，局部变量中可采用通用的命名方式，仅限于n、i、j 等作为循环变量使用。**  
一定不要写出如下这样的代码：  
   int p;  
   char i;  
   int c;  
   char \* a;  
一般来说习惯上用n,m,i,j,k 等表示int 类型的变量；c，ch 等表示字符类型变量；a 等表示数组；p 等表示指针。当然这仅仅是一般习惯，除了i,j,k 等可以用来表示循环变量外，别的字符变量名尽量不要使用。  
  
**14、定义变量的同时千万千万别忘了初始化。定义变量时编译器并不一定清空了这块内存，它的值可能是无效的数据。**这个问题在内存管理那章有非常详细的讨论，请参看。  
  
**15、不同类型数据之间的运算要注意精度扩展问题，一般低精度数据将向高精度数据扩展。**