**一、c89 c99相比：**

1、增加restrict指针

2、inline（内联）关键字

3、新增数据类型

\_Bool　值是0或1。C99中增加了用来定义bool、true以及false宏的头文件夹<stdbool.h>，以便程序员能够编写同时兼容于C与C++的应用程序。在编写新的应用程序时，应该使用

<stdbool.h>头文件中的bool宏。

long long int C99标准中引进了long long int（-(2e63 - 1)至2e63 - 1）和unsigned long long int（0 - 2e64 - 1）。long long int能够支持的整数长度为64位。

4、对数组的增强

5、单行注释

引入了单行注释标记 "//" , 可以象C++一样使用这种注释了。

6、分散代码与声明

7、预处理程序的修改

8、for语句内的变量声明

C99中，程序员可以在for语句的初始化部分定义一个或多个变量，这些变量的作用域仅于本for语句所控制的循环体内。

9、复合赋值

C99中，复合赋值中，可以指定对象类型的数组、结构或联合表达式。当使用复合赋值时，应在括弧内指定类型，后跟由花括号围起来的初始化列表；若类型为数组，则不能指定数组的大小。建成的对象是未命名的。

例： double \*fp = (double[]) {1.1, 2.2, 3.3};

该语句用于建立一个指向double的指针fp，且该指针指向这个3元素数组的第一个元素。 在文件域内建立的复合赋值只在程序的整个生存期内有效。在模块内建立的复合赋值是局部对象，在退出模块后不再存在。

10、柔性数组结构成员

11、指定的初始化符

12、printf()和scanf()函数系列的增强

C99中printf()和scanf()函数系列引进了处理long long int和unsigned long long int数据类型的特性。另外，C99还引进了hh修饰符。

13新增了一些库

14、\_\_func\_\_预定义标识符

**二、c99 c11相比**

1. 对齐处理操作符 alignof，函数 aligned\_alloc()，以及 头文件 <stdalign.h>。

2. \_Noreturn 函数标记，类似于 gcc 的 \_\_attribute\_\_((noreturn))。例子：

\_Noreturn void thrd\_exit(int res);

3. \_Generic 关键词，有点儿类似于 gcc 的 typeof。例子：

#define cbrt(X) \_Generic((X), long double: cbrtl, \

default: cbrt, \

float: cbrtf)(X)

4. 静态断言（ static assertions），\_Static\_assert()，在解释 #if 和 #error 之后被处理。例子：

\_Static\_assert(FOO > 0, "FOO has a wrong value");

5. 删除了 gets() 函数，C99中已经将此函数被标记为过时，推荐新的替代函数 gets\_s()。

6. 新的 fopen() 模式，(“…x”)。类似 POSIX 中的 O\_CREAT|O\_EXCL，在文件锁中比较常用。

7. 匿名结构体/联合体。

8. 多线程支持，包括：\_Thread\_local，头文件 <threads.h>，里面包含线程的创建和管理函数（比如 thrd\_create()，thrd\_exit()），mutex （比如 mtx\_lock()，mtx\_unlock()）等等。

9. \_Atomic类型修饰符和 头文件 <stdatomic.h>。

10. 带边界检查（Bounds-checking）的函数接口，定义了新的安全的函数，例如 fopen\_s()，strcat\_s() 等等。更多参考 Annex K。

11. 改进的 Unicode 支持，新的头文件 <uchar.h> 等。

12. 新增 quick\_exit() 函数，作为第三种终止程序的方式，当 exit() 失败时可以做最少的清理工作（deinitializition。

13. 创建复数的宏， CMPLX()。

14. 更多浮点数处理的宏。

15. struct timespec 成为 time.h 的一部分，以及宏 TIME\_UTC，函数 timespec\_get()。

**三．布尔类型**

布尔类型对象可以被赋予文字值true或者false，所对应的关系就是真与假的概念。我们通常使用的方法是利用他来判断条件的真与假，但是一些概念不清的人却不知道布尔类型的对象也可以被看做是一种整数类型的对象，但是他不能被声明成signed，unsigned，short long，如果你生成(short bool found=false;)，那么将会导致编译错误。

其为整数类型的概念是这样的：当表达式需要一个算术值的时候，布尔类型对象将被隐式的转换成int类型也就是整形对象，false就是0，true就是1。

**四、枚举类型**

枚举可以根据Integer、Long、Short或Byte中的任意一种数据类型来创建一种新型变量。这种变量能设置为已经定义的一组之中的一个，有效地防止用户提供无效值。该变量可使代码更加清晰，因为它可以描述特定的值。

枚举的定义枚举类型定义的一般形式为：

enum 枚举名{ 枚举值表 };

在枚举值表中应罗列出所有可用值。这些值也称为枚举元素。

例如：enum weekday{ sun,mou,tue,wed,thu,fri,sat };

其中，枚举值是常量，不是变量。不能在程序中用赋值语句再对它赋值；枚举元素本身由系统定义了一个表示序号的数值，从0开始顺序定义为0，1，2…。如在weekday中，sun值为0，mon值为1…sat值为6。