#### volatile 关键字

volatile 关键字是一种类型修饰符，用它声明的类型变量表示可以被某些编译器未知的因素更改，比如：操作系统、硬件或者其它线程等。遇到这个关键字声明的变量，编译器对访问该变量的代码就不再进行优化，从而可以提供对特殊地址的稳定访问。

一般说来，volatile用在如下的几个地方：

* 1) 中断服务程序中修改的供其它程序检测的变量需要加 volatile；
* 2) 多任务环境下各任务间共享的标志应该加 volatile；
* 3) 存储器映射的硬件寄存器通常也要加 volatile 说明，因为每次对它的读写都可能由不同意义；

#### C++模板特化与偏特化

#### Void\*

当void\*指针作为函数的输入和输出时，表示可以接受任意类型的输入指针和输出任意类型的指针

对主版本模板类、全特化类、偏特化类的调用优先级从高到低进行排序是：全特化类>偏特化类>主版本模板类。这样的优先级顺序对性能也是最好的。

#### =default

自定义默认构造函数存在两个问题：1. 程序员工作量变大 2. 没有编译器自动生成的默认特殊构造函数效率高。

所以使用=default构造默认函数

#### string\_view

string\_view模板专用化提供了一种有效的方法，可将只读、异常安全、非拥有的句柄传递给任何类似于字符串的对象，并将其序列的第一个元素置于位置零。

#### [[noreturn]]

指定函数不会返回，编译器可以做特定优化