1. C和C++的区别：
2. 宏观上：C:面向过程编程，通过main函数调用各个子函数，程序运行的顺序是程序员事先决定好的。C++面向对象编程，将数据和数据的操作放在类中，以事件来驱动对象处理。主要区别是解决问题的思想方法不同。
3. 语法上：类型检查更为严格（C中的const可以用指针修改，C++中的不能），增加了异常处理，运算符重载，标准模板库，命名空间等。
4. 变量声明的位置：c中变量声明和代码是分开的，必须在函数开始处声明。C++中变量可以在任意处声明，只要保证先声明后使用的原则就可以。
5. struct:C++中的struct结构体支持成员函数的定义，C中的不支持。
6. bool类型：C++中才有。
7. 函数：C语言函数没有默认参数值，C++中有
8. 引用：C++中有引用，函数调用可以作为左值。
9. C++中struct和Class的区别：
10. 默认的继承访问权限。struct是public的，class是private的。
11. C++的缺点：
    1. 内容多，学起来难
    2. 没有垃圾回收机制，可能引起内存泄漏。
12. 线程和进程：
    1. 进程有独立的地址空间，线程只是一个进程中不同的执行路径，线程有自己的堆栈和局部变量，但线程之间没有单独的地址空间，一个线程死掉就等于整个进程死掉，所以多进程的程序要比多线程的程序健壮。进程切换时，耗费资源较大，效率差，但对于一些要求同时进行并且又要共享某些变量的并发操作，只能用线程，不能用进程。
    2. 一个程序至少有一个进程，一个进程至少有一个线程。多个线程共享内存。
    3. 线程在执行过程中与进程还是有区别的。每个独立的线程有一个程序运行的入口、顺序执行序列和程序的出口。但是线程不能够独立执行，必须依存在应用程序中，由应用程序提供多个线程执行控制。
    4. 从逻辑角度来看，多线程的意义在于一个应用程序中，有多个执行部分可以同时执行。但操作系统并没有将多个线程看做多个独立的应用，来实现进程的调度和管理以及资源分配。这就是进程和线程的重要区别。