空类的sizeof大小。

类A空，类B继承A，B中有一个虚函数，sizeof大小

类A空，类B中有一个虚函数和一个类A对象，sizeof大小

函数重载。如果返回值不同或者参数位置不同是否是重载

构造函数能否是虚函数，为什么

C++变成可执行文件的过程

拷贝构造函数和赋值函数。拷贝构造函数有什么缺点

1、list和vector实现有什么区别？讲讲vector的resize操作？

虚函数是否可以是内联函数

进程间通信方式有哪些？

虚函数实现原理

9. 构造函数与析构函数可以是虚函数吗

线程间同步、互斥

线程模型？

进程和线程的区别，进程间通信方式

struct和union的区别，字节对齐

 线程安全是什么

判断一个函数是否线程安全不是一件很容易的事情。但是读者可以通过下面这几条确定一个函数是线程不安全的。

* a, 函数中访问全局变量和堆。
* b, 函数中分配，重新分配释放全局资源。
* c, 函数中通过句柄和指针的不直接访问。
* d, 函数中使用了其他线程不安全的函数或者变量。

因此在编写线程安全函数时，要注意两点：

* 1, 减少对临界资源的依赖，尽量避免访问全局变量，静态变量或其它共享资源，如果必须要使用共享资源，所有使用到的地方必须要进行互斥锁 (Mutex) 保护；
* 2, 线程安全的函数所调用到的函数也应该是线程安全的，如果所调用的函数不是线程安全的，那么这些函数也必须被互斥锁 (Mutex) 保护；

I/O多路复用

子类构造析构函数的调用顺序

linux io模式？

如何保证线程安全？

*简述TCP三次握手和四次挥手*

杀死一个进程怎么操作

有哪些进程杀不死

僵尸进程怎么杀死

简述TCP拥塞控制

分清楚C++中重载、重写和重定义（隐藏）

我觉得只有边沿触发才必须设置为非阻塞。

边沿触发的问题：

1. sockfd 的边缘触发，高并发时，如果没有一次处理全部请求，则会出现客户端连接不上的问题。不需要讨论 sockfd 是否阻塞，因为 epoll\_wait() 返回的必定是已经就绪的连接，所以不管是阻塞还是非阻塞，accept() 都会立即返回。

2. 阻塞 connfd 的边缘触发，如果不一次性读取一个事件上的数据，会干扰下一个事件，所以必须在读取数据的外部套一层循环，这样才能完整的处理数据。但是外层套循环之后会导致另外一个问题：处理完数据之后，程序会一直卡在 recv() 函数上，因为是阻塞 IO，如果没数据可读，它会一直等在那里，直到有数据可读。但是这个时候，如果用另一个客户端去连接服务器，服务器就不能受理这个新的客户端了。

3. 非阻塞 connfd 的边缘触发，和阻塞版本一样，必须在读取数据的外部套一层循环，这样才能完整的处理数据。因为非阻塞 IO 如果没有数据可读时，会立即返回，并设置 errno。这里我们根据 EAGAIN 和 EWOULDBLOCK 来判断数据是否全部读取完毕了，如果读取完毕，就会正常退出循环了。

总结一下：

1. 对于监听的 sockfd，最好使用水平触发模式，边缘触发模式会导致高并发情况下，有的客户端会连接不上。如果非要使用边缘触发，可以用 while 来循环 accept()。

2. 对于读写的 connfd，水平触发模式下，阻塞和非阻塞效果都一样，建议设置非阻塞。

3. 对于读写的 connfd，边缘触发模式下，必须使用非阻塞 IO，并要求一次性地完整读写全部数据。