

1、引用与指针有什么区别？

答：引用必须被初始化，指针不必；引用初始化以后不能被改变，指针可以改变所指的对象；不存在指向空值的引用，但是存在指向空值的指针。

2、头文件的作用是什么？

答：（1）通过头文件来调用库功能。在很多场合，源代码不便（或不准）向用户公布，只要向用户提供头文件和二进制的库即可。用户只需要按照头文件中的接口声明来调用库功能，而不必关心接口怎么实现的。编译器会从库中提取相应的代码。

（2）头文件能加强类型安全检查。如果某个接口被实现或被使用时，其方式与头文件中的声明不一致，编译器就会指出错误，这一简单的规则能大大减轻程序员调试、改错的负担。`static` 有什么用途？（请至少说明两种）

限制变量的作用域；设置变量的存储域

3、头文件中的 `ifndef/define/endif` 干什么用？

答：防止头文件被重复引用

4、描述实时系统的基本特性

答：在特定时间内完成特定的任务，实时性与可靠性

5、全局变量和局部变量在内存中是否有区别？如果有，是什么区别？

答：全局变量储存在静态数据库，局部变量在堆栈。

6、什么是平衡二叉树？

答：左右子树都是平衡二叉树，且左右子树的深度差值的绝对值不大于 1。

7、简述数组与指针的区别

答：数组要么在静态存储区被创建（如全局数组），要么在栈上被创建。指针可以随时指向任意类型的内存块。堆栈溢出一般是由什么原因导致的？

没有回收垃圾资源

8、Internet 采用哪种网络协议？该协议的主要层次结构？

答:tcp/ip 应用层 /传输层 /网络层 /数据链路层 /物理层

9、C++ 中引用与指针的区别；

答： 引用实际上是所引用的对象或变量的别名，而指针是包含所指向对象或变量的地址的变量；引用在定义时必须初始化，而指针在定义时不初始化；不可以有 NULL 的引用，而可以有指向 NULL 的指针；引用在初始化后不可以改变引用关系，而指针可以随时指向其他对象（非 const 指针）。

10、Internet 物理地址和 IP 地址转换采用什么协议？

答： ARP (Address Resolution Protocol) （地址解析协议）

11、IP 地址的编码分为哪俩部分？

答： IP 地址由两部分组成，网络号和主机号。不过是要和 “子网掩码 ” 按位与上之后才能区分哪些是网络位哪些是主机位。

12、为什么数组名作为参数，会改变数组的内容，而其它类型如 int 却不会改变变量的值？

答： 当数组名作为参数时，传递的实际上是地址。而其他类型如 int 作为参数时，由于函数参数值实质上是实参的一份拷贝，被调函数内部对形参的改变并不影响实参的值。

13、局部变量能否和全局变量重名？

答： 能，局部会屏蔽全局。要用全局变量，需要使用 "::" 局部变量可以与全局变量同名，在函数内引用这个变量时，会用到同名的局部变量，而不会用到全局变量。对于有些编译器而言，在同一个函数内可以定义多个同名的局部变量，比如在两个循环体内都定义一个同名的局部变量，而那个局部变量的作用域就在那个循环体内。

14、如何引用一个已经定义过的全局变量？

答： extern

可以用引用头文件的方式，也可以用 extern 关键字，如果用引用头文件方式来引用某个在头文件中声明的全局变理，假定你将那个编写错了，那么在编译期间会报错，如果你用 ex

tern 方式引用时，假定你犯了同样的错误，那么在编译期间不会报错，而在连接期间报错。

15、全局变量可不可以定义在可被多个 .C 文件包含的头文件中？为什么？

答：可以，在不同的 C 文件中以 static 形式来声明同名全局变量。可以在不同的 C 文件中声明同名的全局变量，前提是其中只能有一个 C 文件中对此变量赋初值，此时连接不会出错。

16、内联函数在编译时是否做参数类型检查

答：内联函数要做参数类型检查，这是内联函数跟宏相比的优势。

17、请问 C++ 的类和 C 里面的 struct 有什么区别？

答：class 中默认的成员访问权限是 private 的，而 struct 中则是 public 的。

18、全局变量和局部变量有什么区别？怎么实现的？操作系统和编译器是怎么知道的？

答：生命周期不同：全局变量随主程序创建和创建，随主程序销毁而销毁局部变量在局部函数内部，甚至局部循环体等内部存在，退出就不存在；内存中分配在全局数据区
使用方式不同：通过声明后全局变量程序的各个部分都可以用到局部变量只能在局部使用；分配在栈区操作系统和编译器通过内存分配的位置来知道的，全局变量分配在全局数据段并且在程序开始运行的时候被加载。局部变量则分配在堆栈里面。

19、do…… while 和 while …… do 有什么区别？

答：前一个循环一遍再判断，后一个判断以后再循环

20、队列和栈有什么区别？

答：队列先进先出，栈后进先出

21、软件测试都有那些种类？

黑盒：针对系统功能的测试 白盒：测试函数功能，各函数接口。

22、堆和栈中的存储内容

答：栈：在函数调用时，第一个进栈的是主函数中后的下一条指令（函数调用语句的下一条可执行语句）的地址，然后是函数的各个参数，在大多数的 C 编译器中，参数是由右往左入栈的，然后是函数中的局部变量。注意静态变量是不入栈的。当本次函数调用结束后，局部变量先出栈，然后是参数，最后栈顶指针指向最开始存的地址，也就是主函数中的下一条指令，程序由该点继续运行。堆：一般是在堆的头部用一个字节存放堆的大小。堆中的具体内容程序员安排。

23、类的静态成员和非静态成员有何区别？

答：类的静态成员每个类只有一个，非静态成员每个对象一个。

24、数组和链表的区别

数组：数据顺序存储，固定大小

链表：数据可以随机存储，大小可动态改变

25、进程和线程的差别。

答：线程是指进程内的一个执行单元，也是进程内的可调度实体。

与进程的区别：

- (1)调度：线程作为调度和分配的基本单位，进程作为拥有资源的基本单位
- (2)并发性：不仅进程之间可以并发执行，同一个进程的多个线程之间也可并发执行
- (3)拥有资源：进程是拥有资源的独立单位，线程不拥有系统资源，但可以访问隶属于进程的资源。
- (4)系统开销：在创建或撤消进程时，由于系统都要为之分配和回收资源，导致系统的开销明显大于创建或撤消线程时的开销。

26、关键字 static 的作用是什么？

答：这个问题很少有人能回答完全。在 C 语言中，关键字 static 有三个明显的作用：1). 在函数体，一个被声明为静态的变量在这一函数被调用过程中维持其值不变。2). 在模块内（但在函数体外），一个被声明为静态的变量可以被模块内所用函数访问，但不能被模块外其它函数访问。它是一个本地的全局变量。3). 在模块内，一个被声明为静态的函数只可被这一模块内的其它函数调用。那就是，这个函数被限制在声明它的模块的本地范围内使用。大多数应试者能正确回答第一部分，一部分能正确回答第二部分，同是很少的人能懂得第三部分。这是一个应试者的严重的缺点，因为他显然不懂得本地化数据和代码范围的好处和重要性。

27、确定模块的功能和模块的接口是在软件设计的那个阶段完成的？

答：概要设计阶段

1. 进程和线程的差别。

线程是指进程内的一个执行单元，也是进程内的可调度实体。与进程的区别：

(1) 调度：线程作为调度和分配的基本单位，进程作为拥有资源的基本单位

(2) 并发性：不仅进程之间可以并发执行，同一个进程的多个线程之间也可并发执行

(3) 拥有资源：进程是拥有资源的一个独立单位，线程不拥有系统资源，但可以访问隶属于进程的资源。

(4) 系统开销：在创建或撤消进程时，由于系统都要为之分配和回收资源，导致系统的开销明显大于创建或撤消线程时的开销。

28、测试方法

答：人工测试：个人复查、抽查和会审

机器测试：黑盒测试和白盒测试

29、谈谈 IA32 下的分页机制

答：小页 (4K) 两级分页模式，大页 (4M) 一级

30、给两个变量，如何找出一个带环单链表中是什么地方出现环的？

答：一个递增一，一个递增二，他们指向同一个接点时就是环出现的地方

31、在 IA32 中一共有多少种办法从用户态跳到内核态？

答：通过调用门，从 ring3 到 ring0，中断从 ring3 到 ring0，进入 vm86 等等

32、一个单向链表，不知道头节点，一个指针指向其中的一个节点，问如何删除这个指针指向的节点？

答：将这个指针指向的 next 节点值 copy 到本节点，将 next 指向 next->next,并随后删除原 next 指向的节点。

33、进程死锁的原因

答：资源竞争及进程推进顺序非法

34、死锁的 4 个必要条件

答：互斥、请求保持、不可剥夺、环路

35、死锁的处理

答：鸵鸟策略、预防策略、避免策略、检测与解除死锁

36、操作系统中进程调度策略有哪几种？

答：FCFS(先来先服务)，优先级，时间片轮转，多级反馈

37、C 语言参数的入栈顺序？为什么这么实现？

答：参数从左往右依次压栈对于被调用的函数来说，对实参的处理将变得非常容易。
调用者（ Caller）将参数从右往左压栈，那么对于最左边的函数，即第一个参数是在栈顶。
那么被调用的函数（ Caller）就可以直接用 `bp` 去取该参数， `bp+4` 则指向第二个实参，依此类推。这是 C 调用协定。这样做的好处还有就是可以处理不定参数个数的函数调用。