1. http 的post与get区别与联系，实践中如何选择它们？

（1）get是从服务器上获取数据，post是向服务器传送数据。

（2）在客户端，Get方式在通过URL提交数据，数据在URL中可以看到；POST方式，数据放置在HTML HEADER内提交。

（3）对于get方式，服务器端用Request.QueryString获取变量的值，对于post方式，服务器端用Request.Form获取提交的数据。

（4）GET方式提交的数据最多只能有1024字节，而POST则没有此限制。

（5）安全性问题。正如在（1）中提到，使用 Get 的时候，参数会显示在地址栏上，而 Post 不会。所以，如果这些数据是中文数据而且是非敏感数据，那么使用 get；如果用户输入的数据不是中文字符而且包含敏感数据，那么还是使用 post为好。

Http定义了与服务器交互的不同方法，最基本的方法有4种，分别是GET，POST，PUT，DELETE。URL全称是资源描述符，我们可以这样认为：一个URL地址，它用于描述一个网络上的资源，而HTTP中的GET，POST，PUT，DELETE就对应着对这个资源的查，改，增，删4个操作。GET一般用于获取/查询资源信息，而POST一般用于更新资源信息。

1. 检查内存管理问题的方式有哪些？

1）点击Xcode顶部菜单中的ProductàAnalyze。这种方法主要可以查看内存泄露，变量未初始化，变量定义后没有被使用到

2）使用Instrument工具检查。点击Xcode顶部菜单中的ProductàProfile，弹出一个界面，选择左侧的Memory后，再选右侧的Leaks。

3）人工检查

3、NSString 对象为何用copy？NSMutableString呢？

NSString在Objective-C中是一种非常特殊的对象，其引用系数不受引用计数规则的控制。NSString对象不管是alloc、retain还是release，其引用计数都是-1。对于NSString来说，用alloc和用retain都是指向同一块内存，区别不大。对于NSMutableString来说，alloc实际上就是开辟了一块新内存，再把内容复制进来，而retain内存不变引用计数+1。如果NSMutableString中的内容被改变了的话，用retain之后内容也被改变。这是写程序过程中所不想要的结果。所以这种情况下，用 copy比较安全。苹果的官方SDK，都把NSString属性声明为copy，比如UILabel中的两个属性。

1. 谈谈obj-c的优缺点？

优点：    
1) Cateogies     
2) Posing     
3) 动态识别  
4) 指标计算     
5）弹性讯息传递    
6) 不是一个过度复杂的 C 衍生语言    
7) Objective-C 与 C++ 可混合编程   
缺点:     
1) 不支援命名空间机制 ：须在其类别名称加上前缀，时常引致冲突。  
2) 不支持运算符重载  
3） 不支持多重继承：只容许对象继承一个类别  
4）使用动态运行时类型，所有的方法都是函数调用，很多常见的编译时性能优化方法都不能应用于Obj-C（如内联函数、常数传播、交互式优化、纯量取代与聚集等），性能劣于类似的对象抽象语言（如C++）。

4、堆和栈的区别？

1.申请大小：  
栈：在Windows下,栈是向低地址扩展的数据结构，是一块连续的内存的区域。这句话的意思是栈顶的地址和栈的最大容量是 系统预先规定好的，在WINDOWS下，栈的大小是2M（也有的说是1M，总之是一个编译时就确定的常数），如果申请的空间超过栈的剩余空间时，将提示 overflow。因此，能从栈获得的空间较小。  
堆：堆是向高地址扩展的数据结构，是不连续的内存区域。这是由于系统是用链表来存储的空闲内存地址的，自然是不连续的，而链表的遍历方向是由低地址向高地址。堆的大小受限于计算机系统中有效的虚拟内存。由此可见，堆获得的空间比较灵活，也比较大。  
2.碎片问题：对于堆来讲，频繁的new/delete势必会造成内存空间的不连续，从而造成大量的碎片，使程序效率降低。对于栈来讲，则不会存在这个问题，因为栈是先进后出的队列，他们是如此的一一对应，以至于永远都不可能有一个内存块从栈中间弹出  
3.分配方式：堆都是动态分配的，没有静态分配的堆。栈有2种分配方式：静态分配和动态分配。静态分配是编译器完成的，比如局部变量的分配。动态分配由 alloca函数进行分配，但是栈的动态分配和堆是不同的，他的动态分配是由编译器进行释放，无需我们手工实现。  
4.分配效率：栈是机器系统提供的数据结构，计算机会在底层对栈提供支持：分配专门的寄存器存放栈的地址，压栈出栈都有专门的指令执行，这就决定了栈的效率比较高。堆则是C/C++函数库提供的，它的机制是很复杂的。

1. obj-c有多重继承么?不是的话有什么替代方法?

cocoa 中所有的类都是NSObject 的子类，多继承在这里是用protocol 委托代理来实现的。你不用去考虑繁琐的多继承，虚基类的概念。多态特性在 obj-c 中通过委托来实现。

5、问你觉得iOS7为什么要扁平化，扁平化和之前的比有什么优势（北京龙宇北腾）

开放型问题，参考答案：

苹果首席设计师谈为何会在iOS上选择扁平风格<http://ndnews.oeeee.com/html/201306/11/71078.html>

在iOS 6之前，拟真设计（拟物化设计）作为图标设计和启动应用时的设计备受重视。说到拟真设计的背景，计算机起初就是通过接近现实来让用户明白其功能的。比如，电子书如果没有设计成木纹风格的书架就无法传达其概念，看到翻页日历才会联 想到日历。在不习惯用计算机的人看来，电脑上的垃圾箱要跟实际丢弃文件这件事接近才容易理解。因此计算机开始采用拟真设计。而如今，数字化已经成熟，原本采用隐喻设计的东西不需要再使用比喻了。在计算机中，电子邮件不断独立发展，已经不需要用以前的隐喻设计来做说明了。如果“硬要”采用以前的隐喻设计，变得过于写实，其本身的视认性就会降低。比如“报刊杂志”的木纹和质感现在已成了多余的信息。对于使用该应用的人而言，想看的是其中的内容，只需要画一条能够识别内容的线即可。现在，整个行业都准备放弃不必要的写实概念。就拿电子书来说，以前曾就翻页时在多大程度上表现出纸张背面投过来的字迹等进行过激烈的讨论。而现在设计电子书的前提不再是隐喻纸质书的背面字迹，如何让用户顺畅地阅读内容才更为重要。现实已经超越了隐喻设计，UI世界不再需要像以前那样模仿现实，而是要建立全新的原始状态（original position）。因此，仅用现有的隐喻设计已经无法表现。比如：**iOS 6的“报刊杂志”** 就做成了整个书架的样子，而与此相关的设计还有Passbook、备忘录与各种纹理界面**。拟真设计实例木纹和质感对于接受内容的人而言，已成为多余的信息。**系统设计领导者认为用户已经习惯了触摸屏，所以他们不再需要具有物理外形的按钮。因此我们自由设计的灵活度很大，不必过于拘泥现实世界。我们尝试创建的系统环境不再那么具体化。iOS系统同所有模拟现实世界的外观设计，用更数字化的方式加以呈现，使其对现实世界的依赖越来越少。**iOS 7的“电话”和“邮件”图标。虽然变得扁平，却沿用了iOS 6的图标。**“电话”图标在iOS 7中为电话机模样，“邮件”图标也沿用了信封的设计。苹果并未放弃以前的iOS用户，iOS 7也是对iOS 6的正常升级，因此还有很多直接沿用和继承的部分。如果变得完全不一样了，就会变成无人了解的世界，因此也有自我模仿和简化的一面。拟物化应用在形象的优势背后，同时也会为用户带来许多不必要的噪音。 Windows 的 Metro 则更直观。在当前人与信息的交互下，我们仅需要一个符号就能理解功能。”如果对比一下 Metro 风格的界面与当前iOS界面我们就能感觉到，虽然 Retina 屏幕让显示效果非常优秀，但整个界面五颜六色，信息噪音非常高。而Metro风格的界面则更“符号化”，界面整洁，并且当我们需要寻找某个功能的时候，能迅速通过这些符号化的图形定位到该功能。因此，从拟物化到扁平化的设计过度，更像是一个人机交互发展的过程，在早期我们需要用相似图形去表现计算机功能，而当互联网渗透进入生活，每个人几乎都已经拥有十年以上的网龄之后，扁平化设计则更佳。这在当前的苹果与微软之间能看到巨大的差异，而谷歌则有点中庸，选择了一种介乎于拟物化与扁平化的设计风格，不提供过多的细节，也不过于简化。从Chrome Logo的变化就能看得出来，原本立体、多细节的Logo 变为扁平化的Logo。

6、假如ios8由你设计你将怎样设计（北京龙宇北腾）

开放型问题，现在已经出来了。

7、谈谈安卓与苹果的优缺点（北京龙宇北腾）

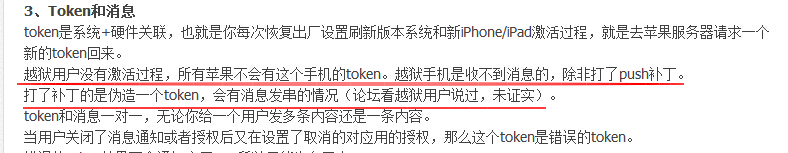
参考答案：苹果系统优点是左右流畅，软件多，界面华丽，图标统一，很美观；缺点是系统封闭，不允许用户过多的个性化设置，而且只能在苹果手机上用。安卓系统优点是开放，可以自己扩展的东西很多，支持的硬件也多，各个价位的手机都有；缺点就是软件太杂乱，兼容性有问题，图标混乱不美观。iOS的确比android流畅，这仅仅体现在较大软件切换时，其他差不多流畅，iOS并不能做到完全后台，如果它完全后台估计也不会比安卓流畅多少。反之，如果安卓只是注重单个运行，流畅度也会大大提升，iOS系统更新没有android那么频繁，爱体验的人会选安卓，那些怕烦的会选iOS。iOS的硬件需求选不及android，以至于android机会相对iOS较热，较费电额。

|  |  |
| --- | --- |
| |  | | --- | | android机优缺点 优点 1.可以自定义桌面 以及各类插件 2.部分手机支持FM收音机 3.各种ROM刷机 正所谓萝卜白菜各有所爱 众多ROM里 总有适合你的 4.实体按键比iphone多 有些操作会很方便  例子“我看电子书的时候 有一个按键可以设置为翻页 按一下翻一页” 5.随意换电池，也可以用大容量电池【换特殊背壳才行】 6.机型多，1000左右的也有，四五千的也有 选择多 缺点 1.机型众多导致的应用不通用，每个机器都要找自己专用的 2.机型升级过快导致【拼硬件】的现象，用户体验较差 3.各种ROM良莠不齐，有的狂吃电量之类的缺陷 4.机型多导致的 某种机器零件稀缺，维修困难  例【我的G3坏了，人家告诉我修好的钱能换个新的..........汗啊 才出了几年】 5.一些老机器被Google抛弃 没有官方升级 关于后台的问题每人都有自己的看法，我就说下【自己的看法】  我比较新欢ios的伪后台因为伪后台不影响机子的使用，而我用过的android机 总需要关后台要不然就会很卡，经常有因为后台程序过多导致 来电话无法接听 因为内存不够 下面说iphone的优缺点 优点 1.简单易懂，容易操作 2.应用程序多，质量好，种类齐全 3.硬件和软件高度匹配，硬件能很好地运转软件，软件能很好地发挥硬件的机能 4.配件多 各种壳子 镜头  5.用户体验好 6.手机质感好【不知道算不算】 缺点 1.贵  【不过可以买3Gs】 2.【中国区】售后服务不完善 导致维修一下就好好几百 3.易遭小偷关顾 4.软件收费【不过一般的10块左右 贵的大概60以内吧 少吃顿必胜客 大M 就好了】 5.易造成攀比现象 6.实体按键少【不过这是苹果的特色 无伤大雅】 | |

12、对安卓新版本的理解（北京龙宇北腾）

13、什么 点阵密度 视觉差（北京龙宇北腾）

14、越狱后消息推送有什么不同



15、有方法查看当前系统内存使用情况吗？（北京龙宇北腾）

1.静态分析   
通过静态分析我们可以最初步的了解到代码的一些不规范的地方或者是存在的内存泄漏，这是我们第一步对内存泄漏的检测。当然有一些警告并不是我们关心的可以略过。  
2.通过instruments来检查内存泄漏  
这个方法能粗略的定位我们在哪里发生了内存泄漏。方法是完成一个循环操作，如果内存增长为0就证明我们程序在该次循环操作中不存在内存泄漏，如果内存增长不为0那证明有可能存在内存泄漏，当然具体问题需要具体分析。  
3.代码测试内存泄漏  
在做这项工作之前我们要注意一下，在dealloc的方法中我们是否已经释放了该对象所拥有的所有对象。观察对象的生成和销毁是否配对。准确的说就是init（创建对象的方法）和dealloc是否会被成对触发（简单说来就是走一次创建对象就有走一次dealloc该对象）。  
下面是自己遇到的一些比较隐秘的造成内存泄漏的情况：  
1.两个对象互相拥有：也就是说对象a里面retain/addSubview了b对象，b对象同时也retain/addSubView了a对象。注意：delegate不要用retain属性，要用assign属性也会导致互相拥有。  
2.有时候需要用removeFromSuperView来释放：具体说明，也许我的a对象拥有一个b对象，b对象add到了c对象上，而在我们的设计中b对象的生命周期应该和a对象相同；这时候只一句[b release]/self.b = nil是不能把b对象释放掉的（一般情况下release会使其retainCount－1,[super dealloc]会再次将所有subView的retainCount－1,而b并不是a的subView，所有最后的一次－1没有了）；所以我们需要在之前加上[b removeFromSuperView]。

搜房网面试问题大概：（写什么问什么）

1. 代理通知kvo在不使用的时候怎么做？搜房网

.通知中心Notification Center，它是一个单例对象，允许当事件发生时通知一些对象。它允许我们在低程度耦合的情况下，满足控制器与一个任意的对象进行通信的目的。这种模式的基本特征是为 了让其他的对象能够接收到在该controller中发生某种事件而产生的消息，controller用一个key（通知名称）。这样对于 controller来说是匿名的，其他的使用同样的key来注册了该通知的对象（即观察者）能够对通知的事件作出反应。

优势：

1.不需要编写多少代码，实现比较简单；

2.对于一个发出的通知，多个对象能够做出反应，即1对多的方式实现简单

3.controller能够传递context对象（dictionary），context对象携带了关于发送通知的自定义的信息

缺点：

1.在编译期不会检查通知是否能够被观察者正确的处理；

2.在释放注册的对象时，需要在通知中心取消注册；

3.在调试的时候应用的工作以及控制过程难跟踪；

4.需要第三方对喜爱那个来管理controller与观察者对象之间的联系；

5.controller和观察者需要提前知道通知名称、UserInfo dictionary keys。如果这些没有在工作区间定义，那么会出现不同步的情况；

6.通知发出后，controller不能从观察者获得任何的反馈信息。

随便介绍下.委托delegation和键值观察key value observing，KVO

2、数据库访问的时候，多线程中锁住怎么办? 搜房网

3、运行时你是怎么理解的，怎么用的? 搜房网

ObjC Runtime 其实是一个 Runtime 库，基本上用 C 和汇编写的，这个库使得 C 语言有了面向对象的能力（脑中浮现当你乔帮主参观了施乐帕克的 SmallTalk 之后嘴角一抹浅笑）。这个库做的事前就是加载类的信息，进行方法的分发和转发之类的。OC是一种面向runtime(运行时)的语言，也就是说，它会尽可能地把代码执行的决策从编译和链接的时候，推迟到运行时。这给程序员写代码带来很大的灵活性，比如说你可以把消息转发给你想要的对象，或者随意交换一个方法的实现之类的。这就要求runtime能检测一个对象是否能对一个方法进行响应，然后再把这个方法分发到对应的对象去。我们拿 C 来跟 ObjC 对比一下。在 C 语言里面，一切从 main 函数开始，程序员写代码的时候是自上而下地，一个 C 的结构体或者说类吧，是不能把方法调用转发给其他对象的。这个问题其实浅涉及到两个概念，运行时和多态。 简单来说，运行时机制使我们直到运行时才去决定一个对象的类别，以及调用该类别对象指定方法。 多态：不同对象以自己的方式响应相同的消息的能力叫做多态。意思就是假设生物类（life）都用有一个相同的方法-eat; 那人类属于生物，猪也属于生物，都继承了life后，实现各自的eat，但是调用是我们只需调用各自的eat方法。 也就是不同的对象以自己的方式响应了相同的消息（响应了eat这个选择器。因此也可以说，运行时机制是多态的基础。比如KVO中我们就用了。

1. 你单例怎么理解怎么用的? 搜房网

Singleton Pattern单例设计模式，通过单例模式可以保证系统中一个类只有一个实例而且该实例易于外界访问，从而方便对实例个数的控制并节约系统资源。如果希望在系统中某个类的对象只能存在一个，单例模式是最好的解决方案。类只能有一个实例，而且必须从一个为人熟知的访问点对其进行访问，比如工厂方法。这个唯一的实例只能通过子类化进行扩展，而且扩展的对象不会破坏客户端代码。例如，UIApplication的sharedApplication方法，任何时候都会返回一个当前应用程序的UIApplication实例。在程序中，我需要一系列的对象，他们每一个内部都包含有一组变量和功能，是静态的，而且整个程序都只需要拥有一个该类的对象。例如：1.控制程序执行的命令器；2.管理数据库；3.音效控制；4.文件处理

1. sdwebimage缓存图片怎么实现的，在内存中有缓存吗? 搜房网

SDWebImage托管在github上。这个类库提供一个UIImageView类别以支持加载来自网络的远程图片。具有缓存管理、异步下载、同一个URL下载次数控制和优化等特征。SDWebImage提供了如下三个category来进行缓存。[MKAnnotationView(WebCache)](http://hackemist.com/SDWebImage/doc/Categories/MKAnnotationView%28WebCache%29.html)、[UIButton(WebCache)](http://hackemist.com/SDWebImage/doc/Categories/UIButton%28WebCache%29.html)、[UIImageView(WebCache)](http://hackemist.com/SDWebImage/doc/Categories/UIImageView%28WebCache%29.html)。

SDImageCache类提供一个创建空缓存的实例，并用方法imageForKey:来寻找当前缓存。UIImage \*myCachedImage = [[SDImageCache sharedImageCache] imageFromKey:myCacheKey]; 存储一个图像到缓存是使用方法storeImage: forKey: 默认情况下，图像将被存储在内存缓存和磁盘缓存中。如果仅仅是想内存缓存中，要使用storeImage:forKey:toDisk:方法的第三个参数带一负值来替代。

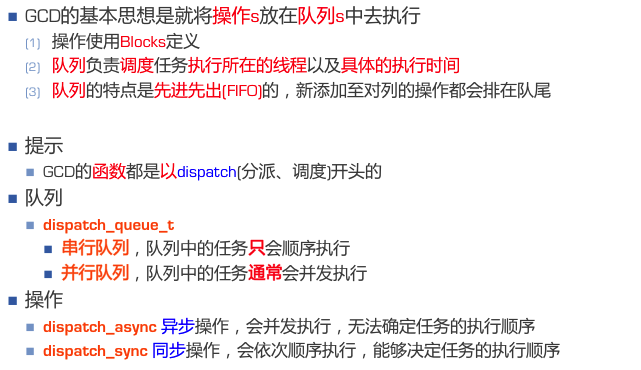
1. 来说说oc这门语言吧，和c c++比，你觉得哪个效率高，高到哪里了? 搜房网

object\_c比C++稍慢，是因为object\_c的一些语言特性，比如反射，影响了执行效率。区别主要有以下一些方面。单一继承：Objective-C不支持多重继承，（同Java和Smalltalk），而C++语言支持多重继承。动态：Objective-C是动态定型（dynamicaly typed)所以它的类库比C++要容易操作。Objective-C 在运行时可以允许根据字符串名字来访问方法和类，还可以动态连接和添加类。C++ 跟从面向对象编程里的Simula67(一种早期OO语言）学派，而Objecive-C属于Smalltalk学派。 在C++里，对象的静态类型决定你是否可以发送消息给它，而对Objecive-C来说，由动态类型来决定。Simula 67学派更安全，因为大部分错误可以在编译时查出。而Smalltalk学派更灵活，比如一些Smalltalk看来无误的程序拿到Simualr 67那里就无法通过。从很多方面来看，C++和Objective-C的差别，与其说时技术上的，不如说是思维方式上的

1. 还有GCD用的那些api, 怎么处理线程间依赖, GCD的group操作有什么? 搜房网

GCD是底层的C语言构成的API，提供了一种新的方法来进行并发程序编写。从基本功能上讲，GCD有点像NSOperationQueue，他们都允许程序将任务切分为多个单一任务然后提交至工作队列来并发地或者串行地执行。GCD比之NSOpertionQueue更底层更高效，并且它不是Cocoa框架的一部分。GCD的API很大程度上基于block，当然，GCD也可以脱离block来使用，比如使用传统c机制提供函数指针和上下文指针。实践证明，当配合block使用时，GCD非常简单易用且能发挥其最大能力。

如果NSOperation对象有执行顺利要求的话，比如operationB需要在operationA执行完毕后才可以 执行，那就可以通过设置NSOperation之间的依赖关系来实现：[operationB addDependency:operationA]。



同步操作不会新建线程、操作顺序执行（没用！）同步任务中包含同步会造成线程阻塞

异步操作会新建线程、操作顺序执行（非常有用！）

8、字符串的内存，对象方法和类方法创建其内存和地址有什么区别？搜房网

9、Meta-class？搜房网

类的实例对象的 isa 指向该类;该类的 isa 指向该类的 metaclass。成员方法记录在class method-list中，类方法记录在meta-class中。即instance-object的信息在class-object中，而class-object的信息在meta-class中。meta-class 是 Class 对象的类。每个 Class 都有个不同的自己的 meta-class（因此每个 Class 都可以有一个自己不同的方法列表）。也就是说每个类的 Class 不完全相同。meta-class 总是会保证 Class 对象会有从基类继承的所有的的实例和类方法，加上之后继承的类方法。如从 NSObject 继承的类，就意味着在所有的 Class（和 meta-class）对象中定义了所有从 NSObject 继承的实例和协议方法。所有的 meta-class 使用基类的 meta-class（NSObject 的 meta-class 用于继承自 NSObject 的类）作为他们自己的类，包括在运行时自己定义的基础的 meta-class。

10、创建一个属性对外只读、对内可读可写？搜房网

属性就是用来封装的，尤其是属性还可能改变的时候，所谓封装就是在外面只能看，只能读，在里面是真正的操作，有时可以改变。细节的改变不影响外边的读，用，就用get()和set()最好。

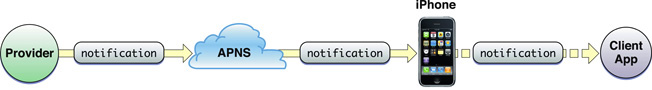
public XX getXX()  
private void setXX(XX xx)

11、在类扩展中重写@property？搜房网

12、协议？搜房网

13、如果在发送异步请求的情况下，当前的界面删除了，会出现什么后果，如何解决？搜房网

14、消息推送是怎么现实的？

iOS消息推送的工作机制可以简单的用下图来概括：  


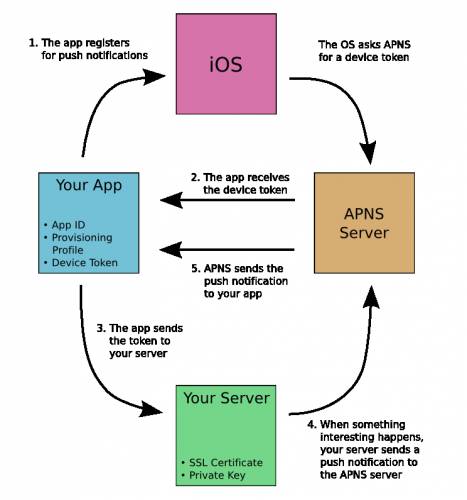
Provider是指某个iPhone软件的Push服务器，APNS是Apple Push Notification Service的缩写，是苹果的服务器。

上图可以分为三个阶段：

第一阶段：应用程序把要发送的消息、目的iPhone的标识打包，发给APNS。

第二阶段：APNS在自身的已注册Push服务的iPhone列表中，查找有相应标识的iPhone，并把消息发送到iPhone。

第三阶段：iPhone把发来的消息传递给相应的应用程序，并且按照设定弹出Push通知。



从上图我们可以看到：

1、应用程序注册消息推送。

2、iOS从APNS Server获取device token，应用程序接收device token。

3、应用程序将device token发送给PUSH服务端程序。

4、服务端程序向APNS服务发送消息。

5、APNS服务将消息发送给iPhone应用程序。

15、一上来，直接就说，我给你出个简单的，你给我写一个贪吃蛇的算法？金山面试

16、然后，还有一个面试是，请写一个类似于NSMutableArray的类，可以添加，删除，以及如何以最快的速度查找某个元素？金山面试

17、群组通讯怎么做？xmpp里面不是提供有接口吗？那里面除了一对一通信聊天之外，还有群聊的接口（金山面试）

18、算法：给一个数字，判断从左读和从右读是否是一致的？例如12321和4444（北京品科艺科技有限公司）

19、给一个数组[1、3、2、4、8]排序后[1、2、3、4、8]？北京品科艺科技有限公司

20、怎么用下面的类？北京品科艺科技有限公司

1）NSuserDefault

2）NSMange Object Context

3）NSPredict

21、什么是plist文件？怎么加载？平时怎么用？北京品科艺科技有限公司

22、怎么从xib文件加载成UIview？北京品科艺科技有限公司

23、通知与代理的区别？北京品科艺科技有限公司

24、viewDidload和viewWillApper区别？北京品科艺科技有限公司

25、UITablewCell。。。。。。didRow。。。。。和UITablewCell。。。。。。willDid。。。。。区别？北京品科艺科技有限公司

26、流媒体面试？

今天问的 视频 直播 和 视频切割 播放

缓存

视频直播，就是服务器端下载好数据或者有视频镜像做好切片后发送给客户端

银客网：

1. OC有没有多继承？没有的话用什么代替？
2. 线程和进程的区别？
3. 多线程有几种方法？各有什么优缺点？
4. 代理、通知、KVO各有什么优缺点？

康源互动，网信金融点财网的面试题：

31,OAuth2.0授权的过程，是否用过1.0？

32,id是编译时还是运行时

33,蓝牙

34,自定义实现控制器的切换

35,runtime

36,storyboard和xib的区别？

37,谈谈对iOS和android的看法

38,fmdb操作表的常用两个方法

39,fmdb修改数据返回的布尔值是表示这条sql执行成功还是数据修改成功

40，如何使用支付宝

41，如何使用分享

42，io8的新特性

43、请你用简单的语言描述 客户端的工作的过程

44、Macintosh HD:Users:zuoze:Desktop:BZT8UJ]XF}05A[~VFYNF11W.jpg

45、怎么用c++ 实现kvo？

46、怎么实现在一个线程中同时执行两个任务？答应不可能，挖坑让你跳

47、collectionView 是ios 6,还是7出现的？

48、机试：实现一个无线滚动？

49、iOS中怎么使用二维码？

50、在公司开发一个项目的具体流程？各部门怎么协作？

51、你的项目中是否用到了UDP /tcp/socket？

52、网上的实现过程用到了runtime机制？<http://bbs.9ria.com/thread-242558-1-1.html>

53、AFN /SDWebImage/ HDD框架底层怎么实现的？

54、NSFileManager/NSFileHandle为什么不用文件加载而用数据库？

55、三维动画的旋转的原理？底层怎么实现？点与点 用距阵变换实现的。

56、写block c++ 写容器 链表, 数据库 增加？

57、逻辑运算跟位运算的区别？

58、用c++实现单例？

59、你用过github么 ？用来干嘛？

60、大文件上传服务器,用什么办法？Socket、最好是tcp\ip一段一段读取，读取一段发一段，利用NSFileHandle。

61、CFNetwork基于啥？

62、还问了afn和asi基于什么？

63、uikit基于什么？

64、coreanimation的上一层和下一层分别是什么？

65、多线程有几种方式？4种NSThread

66、http协议？post.get.put等等

67、默写二分查找算法？

68、默写block函数？

69、ios6/ios7导航栏里面的结构有什么变化？

70、iOS5/6什么时候出来的？

71、你当时学iOS用得开发工具是几你怎么说?开始学的时候是arc 还是非arc 怎么回答?arc从Xcode几开始有的？ios版本是多少？

答：iOS5(2011年6月出)开始出现ARC，2011年11月29日 4.3就支持ARC了。2012年6月，苹果公司在WWDC 2012上宣布了iOS 6，提供了超过 200 项新功能。

2013年6月10日，苹果公司在WWDC 2013上发布了iOS 7，几乎重绘了所有的系统APP，去掉了所有的仿实物化，整体设计风格转为扁平化设计。将于2013年秋正式开放下载更新。[1]

2013年9月10日，苹果公司在2013秋季新品发布会上正式提供iOS 7下载更新。

2014年6月3日（太平洋时间2014年6月2日），苹果公司在WWDC 2014上发布了iOS 8，并提供了开发者预览版更新。

72、字符串通常用copy关键词，但为什么用copy不用strong呢？

73、delegate为什么需要用weak呢？delegate内部有retain,为了防止循环引用, delegate用strong不能释放 循坏引用

74、根据什么确定要开启线程数？一般情况下开启几条？

答：一般不需要管理线程数量，程序开放时是针对队列开发的，向队列添加操作就可以了。具体的线程数量由底层的线程池管理，开启线程的数量一般会根据用户的网络情况决定。WIFI一般5-6条，3G一般2-3条。

75、设计一个忽略大小写比较两个字符串的算法？

76、怎么过滤一段既有字符串又有数字的，让他只剩数字？

77、Sdwebimage的设计思想？多线程下载、内存管理，防止内存溢出。

解析下xml？

- (IBAction)loadData

{

// 1. URL

NSURL \*url = [NSURL URLWithString:@"http://127.0.0.1/videos.xml"];

// 2. Request

NSURLRequest \*request = [NSURLRequest requestWithURL:url];

// 3. 异步方法

[NSURLConnection sendAsynchronousRequest:request queue:[NSOperationQueue mainQueue] completionHandler:^(NSURLResponse \*response, NSData \*data, NSError \*connectionError) {

// XML解析

// 1> 实例化解析器

NSXMLParser \*parser = [[NSXMLParser alloc] initWithData:data];

// 2> 设置代理

parser.delegate = self;

// 3> 解析器开始解析

[parser parse];

}];

}

董大超 13:56:59

// 1. 开始文档

- (void)parserDidStartDocument:(NSXMLParser \*)parser

{

NSLog(@"开始文档");

// 初始化准备，清空数组

[self.videosM removeAllObjects];

}

// 2. 开始节点

- (void)parser:(NSXMLParser \*)parser didStartElement:(NSString \*)elementName namespaceURI:(NSString \*)namespaceURI qualifiedName:(NSString \*)qName attributes:(NSDictionary \*)attributeDict

{

NSLog(@"%@ %@", elementName, attributeDict);

// 如果是video，创建Video对象，准备设置内容，顺便设置videoId

if ([elementName isEqualToString:@"video"]) {

// 实例化对象

self.currentVideo = [[Video alloc] init];

// 设置videoId

[self.currentVideo setValue:attributeDict[@"videoId"] forKeyPath:@"videoId"];

}

// 要拼接的字符串清空

[self.elementM setString:@""];

}

董大超 13:57:09

// 3. 发现节点文本

- (void)parser:(NSXMLParser \*)parser foundCharacters:(NSString \*)string

{

NSLog(@"%@", string);

// 拼接字符串

[self.elementM appendString:string];

}

// 4. 结束节点

- (void)parser:(NSXMLParser \*)parser didEndElement:(NSString \*)elementName namespaceURI:(NSString \*)namespaceURI qualifiedName:(NSString \*)qName

{

NSLog(@"结束节点 %@", elementName);

// 如果是/video，第二步实例化的video添加到数组

if ([elementName isEqualToString:@"video"]) {

[self.videosM addObject:self.currentVideo];

}

else if (![elementName isEqualToString:@"videos"]) {

[self.currentVideo setValue:self.elementM forKeyPath:elementName];

}

董大超 13:57:15

// 5. 结束文档

- (void)parserDidEndDocument:(NSXMLParser \*)parser

{

NSLog(@"结束文档 %@", self.videosM);

self.dataList = self.videosM;

}

// 6. 出错

- (void)parser:(NSXMLParser \*)parser parseErrorOccurred:(NSError \*)parseError

{

NSLog(@"%@", parseError.localizedDescription);

}

解析json数据的demo？

- (void)loadData

{

NSLog(@"JSON 加载……");

// 1. URL

NSURL \*url = [NSURL URLWithString:@"http://192.168.26.3/videos.json"];

// 2. Request

NSURLRequest \*request = [NSURLRequest requestWithURL:url];

// 3. 异步方法

[NSURLConnection sendAsynchronousRequest:request queue:[NSOperationQueue mainQueue] completionHandler:^(NSURLResponse \*response, NSData \*data, NSError \*connectionError) {

// 反序列化

NSArray \*array = [NSJSONSerialization JSONObjectWithData:data options:0 error:NULL];

[array writeToFile:@"/Users/apple/Desktop/123.plist" atomically:YES];

// 转换成video的数组

NSMutableArray \*arrayM = [NSMutableArray array];

for (NSDictionary \*dict in array) {

[arrayM addObject:[Video videoWithDict:dict]];

}

// 设置数据

self.dataList = arrayM;

}];

}

自定义cell的时候用到nstimer会有什么问题？

北京263网络公司：

// 1.过滤“hgdghghfgghh12457t367888dhjhd55fhfh”字符串，仅保留数字

// 2.delegate用什么修饰？weak assign strong ？ 有什么例外？为什么？

// 3 self.name = XXX 与 name = XXX有什么区别

// 4，常用的多线程的API 除了网络异步 还在哪些情况应用

// 5.你在开发中又遇到EXC\_BAD\_ACCEXX吗？你是怎么处理的

// 5，在开发中你有遇到内存不足的警告吗？你是怎么做预处理的？

// 7.画出UIViewController的生命周期

// 8.从viewA跳转到viewB,有几种方式来实现viewB的值传递

//9.有什么可以改进的？

// NSMutableArray \*arr = [[NSMutableArray alloc] init];

// for (int i = 0; i < 1000; i ++) {

// NSString \*str = [NSString stringWithFormat:@"%d",i];

// [arr addObject:str];

// }

// NSLog(@"%@",arr);

// [arr release];

return 0;

}

//10.对于以下代码

/\*\*

\* 代码A

\*/

void exampleA(NSMutableArray \*array){

char b = 'B';

[array addObject:^{

printf("%c\n",b);

}];

}

void testA()

{

NSMutableArray \*array = [NSMutableArray array];

exampleA(array);

void (^block)() = [array objectAtIndex:0];

block();

}

/\*\*

\* 代码B

\*/

void exampleB(NSMutableArray \*array){

[array addObject:^{

printf("C\n");

}];

}

void testB()

{

NSMutableArray \*array = [NSMutableArray array];

exampleA(array);

void (^block)() = [array objectAtIndex:0];

block();

}

//你认为代码A会（B）代码B会（）

// A always works

//B only works with ARC

//C,only works with MRC

//D never works

// 11. 如果我的应用想要获得系统的通讯录，怎么样使系统改变的通讯录同步到我的自己额应用上？如何改进系统的通讯录来实现这种同步

runtime你们在哪里用到了？KVO。程序运行中动态创建一个子类，在子类实现了set方法去监听改变

KVO底层实现原理？

6、支付功能。

7、socket

2、.Objective-C内存管理有哪几种方式,说说你的看法?

1. 编程:一个由整数(不包括0)长度为N(N>0)构成的数组,请只遍历一次(时间复杂度O(1)，不允许开辟额外数组空间)，把所有负数放在原数组左边，正数放在原数组右边(负数和正数不要求是有序)；并找出正数和负数的分界点的位置？
2. 如何判断字符串是否为纯数字？
3. 性能调优，就问我调优工具。
4. 我说协助项目经理架构，就问我怎么架构的，我无语了！