

千锋问题汇总 问题汇总:

1. 你如何理解 iOS 内存管理

1. new alloc copy retain 这些对象我们都要主动的 release 或者 autorelease

2. 如果是类方法创建的对象，那么系统自动释放池自动在适当的时候会帮我们 release

3. ARC xcode 会自动帮我们人工智能的添加 release autorelease 操作

2. C 语言里的数组与 OC 数组的区别

OC 数组是一个对象，有大量的方法，c 没有都需要自己写

C 数组删除是需要后面往前移动，oc 数组自动处理

3. 浅拷贝与深拷贝的区别 或者什么是深拷贝 什么是浅拷贝

copy, mutableCopy

```
@interface A {
```

```
    B *b;
```

```
}
```

浅拷贝只是拷贝对象本身，不会对里面的子对象进一步拷贝

深拷贝会对子对象以及子对象的子对象进一步拷贝

4. 解释 TCP/IP 协议

HTTP

TCP UDP

IP

TCP/IP 是网络开发中常见的传输协议，他传输和 udp 相比是可靠的。http 是基于 tcp/ip 的主要用户互联网的协议

所谓可靠是 tcp 传输对方会给一个 ACK 信号(确认信号)

tcp 传输不如 udp 快，吞吐量不如 udp 大

tcp 是顺序的，udp 是无序的

tcp 会保持连接，udp 不会保持连接

在我们 iOS 开发中用 http 较多。http 协议是一个被动式的

5. 界面之间传值有哪几种方法

单例，代理，直接赋值，通知中心/广播，数据库等多种

6. 什么是 MVC 模式

Model, View, Controller, 是 iOS 开发中的典型的设计模式。比如通过 http 把网络数据下载并解析然后存于自己的数据模型 Model 中，然后通知 controller 去刷新界面，通过 controller 讲 Model 和 View 进行关联，这种模式就叫 mvc。

这样的好处是，可以隔离数据模型 model 和 View 界面。遵循了低耦合的设计思想。

7. 常见的点击手势有哪几种

UIPinchGesture, UITapGestureRecognizer, UISwipeGesture,
UILongPressGesture, UIRotationGesture

8. 瀑布流怎么理解和实现的

1. 如果简单的来说，用 3 个 `tableview` 就可以实现瀑布流，3 个 `tableview` 实现联动滚动。
2. 其实最好的做法在 `scrollview` 上使用 3 个复用队列，如果一种一个 `cell` 超过屏幕，不能 `release`，而是把它回收到复用队列中，如果要创建一个 `cell`，首先从复用队列中取一个，然后使用。

9. 你如何理解复用机制

一般是对 `UIScrollView` 做复用机制，因为 `UIScrollView` 滚动窗口没有复用，所以要做，原理就是超过屏幕的 `view` 不能销毁，而要放在复用队列/池里面存放起来，然后以后要在 `scrollview` 显示 `view` 首先不要 `alloc` 创建，而要首先去复用池里面找有没有可复用的 `view`，如果没有就 `alloc` 如果有就直接用。

10. 类别的作用

类别是给已有的类添加方法，但是不能添加变量，匿名类别除外
0

11. 类别和继承什么区别

类别/类目/`Category` 很方便给现有类添加方法。但是不能添加成员变量，匿名类除外，比如可以给 `NSString` 增加方法，给 `UINavigationController` 增加方法，比如 `SDWebImage` 给 `UIImageView` 增加了 `setImageWithURL:` 方法。类别对于使用者很方便

继承可以给现有的类增加方法和成员变量。继承对于使用者来说不如类别方便，比如对 `SDWebImage` 如果用继承的话，那么就需要写一个类似于 `QFImageView` 然后把所有的 `UIImageView` 改成

UIImageView, 这里不如类别方便

12. C 和 OC 有哪些基本数据类型

C char, short, int, struct, union, enum

NSString, NSArray, NSDictionary.

13. 解释多线程、NSThread、NSOperation、GCD

多线程在 iOS 用的很多, 比如每个 ASI 请求, sdwebimage 请求, 数据请求等待等网络数据处理, 多线程/异步就是主要是为了界面流畅, 防止界面假死。

每一个 ASI 请求就是一个 NSOperation

每一个 NSURLConnection 也是一个线程

NSThread 是创建线程的一个通用的类。比如线程创建, 取消, 开始等。

NSOperation 就是一个简单的以任务为导向的多线程模型。目的是为了不懂操作系统, 不懂线程的人使用的

GCD 类似于 NSOperation, 是一个 blocks 版本的线程模型。

14. 什么是异步

相对于同步来说, 单独起一个或者多线程去处理

异步是一个概念 线程还是一个技术, 异步就是用线程这种技术实现的

比如界面下载数据, 我们启动一个异步任务 ASI 去网络下载数据, 然后异步刷新界面, 我们无需等待网络数据下载完成。

15. 代理和协议什么区别

代理是一种概念，协议是一种技术，代理是用协议来实现的,代理是 2 个对象之间通讯的一种方式。

代理主要做反向传值的。实现系统的一些回调方法，比如 `scrollview` 滑动事件，选择照片，`asi` 网络下载完成等。

iOS 开发和 objective-c 区别

16. 代理和通知中心/广播/NotificationCenter 什么区别

代理主要是反向传值，一般用来 1: 1 的两个对象通讯上。

通知中心是通过注册然后接收事件的一种 n:n(多对多) 的方法

17. UITextField、UITextView 的区别

`UITextField` 不能换行，`UITextView` 可以换行。`TextView` 可以点击字体放大，可以复制。相同的是他们都有代理类似

18. 怎么解析 HTML 源码

一般 HTML 是通过 `webview` 来显示的。一般简单的 html 是通过解析字符串来解析。没有成熟现成 html 解析器。

一般服务器去抓取 html 解析 html 形成 json xml 文件供手机端读取。

19. Objective-C 如何和 javascript 通讯

Objective-c --→javascript ---→html 里面内容

Objective-c 是通过 `stringByEvaluatingJavaScriptFromString` 函数来执行 html 中的 javascript

Javascript -→url --→objective-c 本地方法

需要通过 `uiwebview` 中的代理函数

```
- (BOOL)webView:(UIWebView *)webView  
shouldStartLoadWithRequest:(NSURLRequest *)request  
navigationType:(UIWebViewNavigationType)navigationType;
```

20. 如何调用 iOS 打电话，发短信

```
[[UIApplication sharedApplication] openURL:[NSURL URLWithString:@"tel://1543434"]];  
[[UIApplication sharedApplication] openURL:[NSURL URLWithString:@"sms://1572234"]];  
[[UIApplication sharedApplication] openURL:[NSURL  
urlWithString:@"mailto:hello@hello.com"]];  
[[UIApplication sharedApplication] openURL:[NSURL  
urlWithString:@"http://1000phone.net"]];
```

21. 下拉刷新需要实现哪几个方法,刷新流程

下来一般使用 EGORefresh 进行。原理是利用 scrollView 的反弹效果把刷新 view 加载 scrollView 的负坐标上。通过代理方法去触发。

22. 解释 const, static, inline 关键字

const 修饰指针，或者常量，比如不可变，

static 修饰变量表示作用域，比如全局的私有变量，函数内部的 static 是内部的私有变量。

Static 修饰函数表示函数是文件作用域

Inline 表示内联。一般来说 inline 需要和 static 联合用

```
一般用法是 static inline int max(int a, int b) {  
    return a>b?a:b;  
}
```

static inline 作用是和宏类似，只不过是方便调试（宏不能断掉调试，static inline 可以）。运行时候是一样的。

一般 c/c++短小的函数都用 static inline 内联函数

23. OC 里怎么实现多继承

OC 没有多继承。用协议实现多继承，把协议中的方法在实现的类中重写

C++有多继承，多继承其实用好了很方便。C++多继承很容易出现二一性比如：

类 A { int a;}

类 B: 类 A, 类 C: 类 A

类 D : B, C 那么 int a 到底是继承自谁的

单继承

24. 解释 KVC 和 KVO

Key value coding, Key value observer.

Kvc 是路径访问的规范，kvo 是观察某个变量的变化过程

KVO 可以观察某个对象的变量变化过程，KVC 是满足被观察的编码规范。

KVC/KVO 类似于代理，通知中心。都是一种通讯方法。

25. 什么是单例模式

单例就是在多个对象之间共享数据，类似于全局变量，比如数据库打开一次，多个界面都可以使用

26. NSString *name = @"1000phone.com",[name release]会出现什么情况

这个代码不满足 objective-c 的内存管理黄金法则。没有 alloc 就 release,

27. 列举所知道的 iOS 的 API

28. 列举所知道的 iOS 的类

29. 列举所知道的 iOS UI 控件

30. 解释 C++/STL List、map、set、vector

list = ~ array

map = ~ dictionary

set = ~ nsset

vector = ~ array

31. C/C++ 和 OC 怎么混用

.m -> .mm

32. POST 请求的链接参数怎么拼接

POST 参数有 2 中，一种文件 POST 一种非文件 POST，对于非文件 POST 格式（ form-data/x-www-urlencoded ） 是
name=xxx&id=22&sxx=33

对于文件 POST(multiple/form-data)

33. POST 和 GET 有何区别

GET 和 POST 都是 HTTP 请求方式的 2 中。

POST 是安全的。GET 是不安全的。GET 是放在浏览器中地址暴露出来了。POST 不会。但是在 App 上 GET 和 POST 都看不见。

GET 和 POST 都是和服务器提交参数/通讯的一种方式。

GET 参数不能太长<1024B POST 没有限制<4G

GET 不能上传文件， POST 可以上传文件。

34. 本地存储方式有哪几种

数据库， nsuserdefaults， file ,plist, archive 归档

比较小的就放在 nsuserdefaults(以文件设计的存放在 Library 中)

数据库里面适合存放一条一条的记录

file 一般存普通文件， 图片， 视频， 音频等

plist 存放比 nsuserdefaults 大一些的

归档 nsarchive 可以存对象

35. 解释 OC 里的 id 类型

id 是泛型指针， 可以指向任何 oc 类型，

id = ~ NSObject *

id 是 objc_object 的 typedef， 执行 Class 对象

36. retain、 copy、 readonly、 readwrite 的区别

retain 对计数器+1 copy 是创建一个新对象, readonly 只产生 getter

函数， readwrite 产生 setter, getter 函数

37. copy 是深拷贝还是浅拷贝

copy 缺省是浅拷贝.但是 copy 也可以做成深拷贝

38. copy 和 mutableCopy 区别

copy 是拷贝， mutableCopy 是可变拷贝， 比如把 NSString 通过 mutableCopy 变成 NSMutableString, 把 NSArray 通过 mutableCopy 变成 NSMutableArray.

NSString -> NSMutableString, NSArray -> NSMutableArray,

NSDictionary→NSMutableDictionary, NSData->NSMutableData;

39. 怎么实现对象的本地存储

归档 NSArchieve

40. HTTP 协议、Socket 协议的区别

	数据协议格式		编程
高	HTTP		ASI, NSURLConnection
中	TCP 长连接	UDP	BSD Socket AsyncSocket
低	IP		

HTTP 启动一个 NSURLConnection 在 didFinish

长连接，长期的和服务器保持连接。目的是为了减少每次都要连接。因为和服务器连接的过程非常复杂。也耗时。所以连接一次。

HTTP 短连接只要处理完就断掉，下次重新连接 对服务器的负担小。

长连接对服务器压力很大。

41. 什么是 Push Notification

42. 遍历数组的三种方式有什么区别

for forin iterator

forin 也叫快速循环。为什么他叫快速循环，原因是速度快？为什

么速度快? 可以通过优化提前把 forin 里面的 array 数组数组全部导入到 cpu cache 中。因为 cache 访问速度比内存快至少 100 倍

43. NSArray, NSDictionary 这些是如何做的?

NSArray 使用链表做的 NSDictionary 使用 hash 表做出来的

44. C++怎么调用 C 的方法

```
extern "C" int foo(){ };
```

45. Blocks 语法有什么优点

Blocks 类似于 c 中回调函数, 和代理类似的, 类似于 java 中的 listener 匿名函数

46. Frame 和 bounds 有什么区别

Frame 是子视图相对于父视图的坐标系统, x y width height, bounds 是 uiview 自己的坐标系统 bounds x y=0

47. 什么叫异步下载,怎么实现图片的异步下载

48. 什么是缓存

49. ScrollView 的复用机制

滚动过程中把 scrollview 里面的 超出屏幕的 view 放在重用队列中, 在滚动过程中如果有新的 view 进来, 首先从重用队列中取得 view, 如果没有就新建一个。

50. 什么是委托代理模式

委托代理是 2 个对象之间的一种通讯方式。一方使用协议, 代理

另外一方实现协议，类似于回调，**blocks** 等语法。

目的是为了软件设计的低耦合

51. 什么是通知中心

通知中心是多对多的平等的通讯模式。主要用在多个对象之间松耦合的通讯模式。对象和对象之间通过通知的名字就可以进行关联通讯。

52. 解释关键字 **static**、**const**、**inline**、**static inline**、**volatile**

static 修饰变量表示静态变量，根据作用域 不同

extern 和 **static** 是相反的。

Static 修饰函数是表示函数在本文件有效

Static inline 联合在一起表示内联函数。类似于宏

Const 修饰变量和变量的内容只读

Volatile 表示每次都从内存中真正的读取。主要是用于嵌入式中，读取硬件。

53. 如何实现 2 个线程 一个线程+1 一个线程-1

```
int sum;
NSLock *sumLock;
- (void) demo {
    sumLock = [[NSLock alloc] init];
    [NSThread detachNewThreadSelector:@selector(thread1:) toTarget:self withObject:nil];
    [NSThread detachNewThreadSelector:@selector(thread2:) toTarget:self withObject:nil];
}
- (void) thread1:(id)arg {
    while (1) {
        [sumLock lock];
        sum++;
        [sumLock unlock];
    }
}
```

```

        [NSThread sleepForTimeInterval:1];
    }
}
- (void) thread2:(id)arg {
    while (1) {
        [sumLock lock];
        sum--;
        [sumLock unlock];
        [NSThread sleepForTimeInterval:0.5];
    }
}
}

```

54. OC 中有私有方法吗?怎么实现

在.m 文件中实现 Category，只是在.m 中方法

55. OC 中有私有成员变量吗?怎么实现

在.m 文件中实现匿名类别 Category

56. 成员变量的作用域有哪几种

@public, @protected, @private

57. 数据库能不能存图片,怎么存?

可以存图片， NSData 存，但是我们一般不这样存，我们存路径把图片存在沙盒中。

58. 线程之间怎么通信?

59. 线程生产者，消费者如何实现?

1 个线程是生产者，一个是消费者如何实现这个模型.

```

NSMutableArray *dataList;
NSLock *dataLock;
- (void) produceAndConsume {
    dataList = [[NSMutableArray alloc] init];
    dataLock = [[NSLock alloc] init];
    [NSThread detachNewThreadSelector:@selector(produce:) toTarget:self withObject:nil];
    [NSThread detachNewThreadSelector:@selector(consume:) toTarget:self withObject:nil];
}
- (void) produce:(id)arg {

```

```

int index = 0;
while (1) {
    NSNumber *n = [NSNumber numberWithInt:index++];
    [dataLock lock];
    [dataList addObject:n];
    [NSThread sleepForTimeInterval:0.5];
}
}
- (void) consume:(id)arg {
    while (1) {
        if ([dataList count] > 0) {
            [dataLock lock];
            id obj = [dataList objectAtIndex:0];
            [dataList removeObject:0];
            [dataLock unlock];
            NSLog(@"消费 obj %@", obj);
        }
        [NSThread sleepForTimeInterval:1];
    }
}
}

```

60. 不同屏幕怎么适配

iphone, iphone3G, iphone3GS 320x480

iphone4, iphoen4S 640x960 retina

iphone5, iphone5S, 640x1136

写程序需要有 2 套图片 demo.png demo@2x.png

iphone5 适配.

1136/2-44-49

```
[[UIScreen mainScreen] applicationFrame] = (320x460, 320x548)
```

```
[[UIScreen mainScreen] bounds] = (320x480, 320x568)
```

iPad, iPad2, iPad Mini 1024x768

iPad3, iPad4, 2048x1563

61. 不同版本屏幕旋转怎么适配

62. 内存警告应该怎么做

尽量多的释放内存。把一些内容写到文件中，不要保存在内存中。

63. iOS5, iOS6, iOS7 区别

64. NSNotification 是同步还是异步的

是同步的. 如果需要异步 必须使用 `NSNotificationQueue` 实现

A notification center delivers notifications to observers synchronously. In other words, the `postNotification:` methods do not return until all observers have received and processed the notification. To send notifications asynchronously use **`NSNotificationQueue`**.

In a multithreaded application, notifications are always delivered in the thread in which the notification was posted, which may not be the same thread in which an observer registered itself.

Hope it helps you.