
面试题汇总

百度

1. OC 里用到集合类是什么？

基本类型为：NSArray，NSSet 以及 NSDictionary

可变类型为：NSMutableArray，NSMutableSet 以及 NSMutableDictionary

2. 用 OC 写一个快速排序

```
-(void)quickSortWithArray:(NSMutableArray *)aData
left:(NSInteger)left right:(NSInteger)right{
    if (right > left) {
        NSInteger i = left;
        NSInteger j = right + 1;
        while (true) {
            while (i+1 < [aData count] && [aData objectAtIndex:i] <
[aData objectAtIndex:left]) ;
            while (j-1 > -1 && [aData objectAtIndex:j] > [aData
objectAtIndex:left]) ;
            if (i >= j) {
                break;
            }
            [self swapWithData:aData index1:i index2:j];
        }
        [self swapWithData:aData index1:left index2:j];
    }
}
```

```
[self quickSortWithArray:aData left:left right:j-1];

[self quickSortWithArray:aData left:j+1 right:right];

}

}
```

```
-(void)swapWithData:(NSMutableArray *)aData
index1:(NSInteger)index1 index2:(NSInteger)index2{
    NSNumber *tmp = [aData objectAtIndex:index1];

    [aData replaceObjectAtIndex:index1 withObject:[aData
objectAtIndex:index2]];

    [aData replaceObjectAtIndex:index2 withObject:tmp];
}
```

3. viewController 中 **didReceiveMemoryWarning** 什么时候调用，默认如何处理

在程序收到内存警告的时候会调用该方法

默认操作为检查 controller 是否可以安全地释放它的 **view**，比如 **view** 本身没有 **superview** 并且可以被很容易地重建（通过 **nib** 或者 **loadView** 函数）

4. OC 中 **frame** 与 **bounds** 有什么不同

Frame 指的是：该 **view** 在父 **view** 坐标系统中的位置和大小（参照点是父类的坐标系统）

Bounds 指的是：该 **view** 在本身坐标系统中的位置和大小（参照点是本身的

坐标系统)

5. OC 中有无多重继承，如何实现多重继承

没有多重继承。

cocoa 中所有的类都是 NSObject 的子类，多重继承在这里是用 protocol 协议来实现的，所以我们不用去考虑繁琐的多继承、虚基类的概念，多态特性在 oc 中通过委托来实现。

6. 用 OC 写两个线程，对同一个变量 分别进行+/-操作

```
//首先创建一个线程锁
NSLock *_lock = [[NSLock alloc] init];

//创建两个工作线程 访问同一个资源 类的对象的变量_value

[NSThread detachNewThreadSelector:@selector(thread1:)
toTarget:self withObject:@(10)];
[NSThread detachNewThreadSelector:@selector(thread2:)
toTarget:self withObject:@(10)];
}

//第一个工作线程 中会调用的函数
- (void)thread1:(id)obj{
    //我们需要当前线程 对_value进行 连续+10的操作 每次递增10

    @autoreleasepool {
        int num = [obj intValue];

        [_lock lock]; //加锁 修改之前加锁

        for (int i = 0 ; i < num; i++) {

            _value += 10;
            NSLog(@"thread1_value:%d",_value);
            [NSThread sleepForTimeInterval:0.5];
        }
    }
}
```

```

        [_lock unlock]; //修改完毕要解锁
    }
}

//第二个线程中调用的函数
- (void)thread2:(id)obj{

    @autoreleasepool {
        int num = [obj intValue];

        [_lock lock]; //加锁
        for (int i = 0 ; i < num; i++) {
            _value -= 10;
            NSLog(@"thread2_value:%d", _value);
        }

        [_lock unlock]; //解锁
    }
}

```

7. OC 中如何创建一个线程，要求可以一直工作，不会只执行一次就结束

```

    @synthesize name;

    - (void)viewDidLoad
    {
        [super viewDidLoad];

        // Do any additional setup after loading the view, typically from
        a nib.

        NSThread *thread = [[NSThread alloc] initWithTarget:self
        selector:@selector(threadFunc) object:nil];

        [thread start];
    }

    static bool over = NO;

```

```

- (void)threadFunc
{
    NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
    while (YES) {
        @synchronized(name){
            name = @"Frank";
            [NSThread sleepForTimeInterval:2];
            if ([NSThread isMultiThreaded]) {
                NSLog(@"%@ isMultiThreaded",name);
            }
            if (over) {
                break;
            }
        }
    }
    [pool release];
}

```

8. 在 7 的基础上如何结束一个线程

- a) break;
- b) [thread cancel]

9. 写一个你所经历的项目架构或者 cocoa 里面的 mvc 模式

二. c/c++

1. 得到一个 field 在结构体中的偏移量

```
#define OFFSETOF(type, field) (((size_t)&(((type *)0)->field))
```

2. 引用和指针的区别

1.从现象上看：指针在运行时可以改变其所指向的值，而引用一旦和某个对象绑定后就不再改变。引用访问一个变量是直接访问，而指针是间接访问。

2.从内存分配上看：程序为指针变量分配内存区域，而引用不分配内存区域

3.从编译上看：程序在编译时分别将指针和引用添加到符号表上，符号表上记录的是变量名及变量所对应地址。指针变量在符号表上对应的地址值为指针变量的地址值，而引用在符号表上对应的地址值为引用对象的地址值。符号表生成后就不会再改，因此指针可以改变指向的对象（指针变量中的值可以改），而引用对象不能改

3.写出 float X 与零值比较的 if 语句

```
const double EPSILON = 1.00e-07;  
if (x<abs(EPSILON))
```

4.如何引用一个已经定义过的全局变量

可以用引用头文件的方式，也可以用 extern 关键字的方式来引用定义过的全局变量

5.static 全局变量与普通的区别，局部变量呢？ 函数呢？

static 全局变量与普通的全局变量有什么区别：static 全局变量只初使化一次，防止在其他文件单元中被引用；

static 局部变量和普通局部变量有什么区别：static 局部变量只被初始化一次，下一次依据上一次结果值；

static 函数与普通函数有什么区别：static 函数在内存中只有一份，普通函数在每个被调用中维持一份拷贝

6.队列和栈的区别

栈（Stack）是限定只能在表的一端进行插入和删除操作的线性表。

队列（Queue）是限定只能在表的一端进行插入和在另一端进行删除操作的线性表。

1.队列先进先出，栈先进后出。

2.对插入和删除操作的"限定"。栈是限定只能在表的一端进行插入和删除操作的线性表。队列是限定只能在表的一端进行插入和在另一端进行删除操作的线性表。

3.遍历数据速度不同。队列遍历数据的速度要快得多。

7.写一个单链表 要求可以插入数据和删除单个数据

```
#include<stdio.h>
#include<malloc.h>

typedef struct Node          //定义节点
{
    int data;                //可以为其他类型
    struct Node *next;
}List;

void create(struct Node *&L,int a[],int n) //创建， L 是链表头,a 是要插入数组,n 是要插入元素个数
{
    List *s; int i; L=(List *)malloc(sizeof(List));
    L->next=NULL;
    for(i=0;i<n;i++) {
        s=(List *)malloc(sizeof(List));
        s->data=a[i];
        s->next=L->next; L->next=s;
    }
}

int insert(List *&L,int i,int e) //插入， i 是插入位置， e 是插入的值
{
    int j=0;
    List *p=L,*s;
    while(j<i-1&&p!=NULL) {
        j++;
        p=p->next;
    }
    if(p==NULL)
        return 0;
```

```
else {
    s=(List *)malloc(sizeof(List));
    s->data=e;
    s->next=p->next;
    p->next=s; return 1;
}
}
```

int delete(List *&L,int i,int &e)//i 是要删除元素的位置，e 保存被删除元素的值

```
{
    int j=0;
    List *p=L,*q;
    while(j<i-1&& p!=NULL) {
        j++; p=p->next;
    }
    if(p==NULL)
        return 0;
    else {
        q=p->next;
        if(q==NULL)
            return 0;
        e=q->data;
        p->next=q->next;
        free(p);
        return 1;
    }
}
```


班德尔笔试题。

1 你是怎么理解 delegate，举例如何使用

delegate 就是委托/代理，是 iOS 设计模式中的代理设计模式
比如 UI 中的 UITextField 控件，当键盘弹起时 UITextField 自己是不会收键盘的，这个时候我们可以遵守 UITextFieldDelegate 协议，设置 UITextField 的代理，当点击 return 键时委托代理收键盘

2，让你做个相册你会怎么做

用 UIScrollView 做一个九宫格显示照片缩略图，然后点击进入之后用 UIScrollView 左右滚动显示大图，单击图片还可以放大

3 怎样自定义一个 uiTabBarController

用 UIView 做一个 tabBar 然后在这个 UIView 上粘贴 label 和 button 通过点击 button 来控制切换的视图控制器

4 你用过动画效果吗？请举例说明

实现iOS漂亮的动画效果主要有两种方法，

一种是UIView层面的，

一种是使用CATransition进行更低层次的控制，

第一种是UIView，UIView方式可能在低层也是使用CATransition进行了封装，它只能用于一些简单的、常用的效果展现，这里写一个常用的示例代码，供大家参考。

```
[UIView beginAnimations:@"Curl"context:nil];//动画开始
```

```
[UIView setAnimationDuration:0.75];
```

```
[UIView setAnimationDelegate:self];
```

```
[UIView  
setAnimationTransition:UIViewAnimationTransitionCurlUp  
forView:myview cache:YES];  
[myview removeFromSuperview];  
[UIView commitAnimations];
```

第二种方式相对复杂一些,但如果更好的进行控制,还是使用这种方法吧,基本使用方法可以看一下如下例子:

```
CATransition *animation = [CATransition animation];  
[animation setDuration:1.25f];  
[animation setTimingFunction:[CAMediaTimingFunction  
functionWithName:kCAMediaTimingFunctionEaseIn]];  
[animation setType:kCATransitionReveal];  
[animation setSubtype: kCATransitionFromBottom];  
[self.view.layer addAnimation:animation forKey:@"Reveal"];
```

5, 实现一个 Cell 要有标题和内容

```
- (UITableViewCell *)tableView:(UITableView *)tableView  
cellForRowAtIndexPath:(NSIndexPath *)indexPath{
```

```
    static NSString * cellId = @"cellID";
```

```
    UITableViewCell *cell = [tableView  
dequeueReusableCellWithIdentifier:cellId  
forIndexPath:indexPath];  
    if (cell == nil) {
```

```
        cell = [[UITableViewCell alloc]
initWithStyle:UITableViewCellStyleSubtitle
reuseIdentifier:cellId] autorelease];

    }

    cell.textLabel.text = [NSString stringWithFormat:@"第%d
行", indexPath.row];

    cell.detailTextLabel.text = [_dataArray
objectAtIndex:indexPath.row];

    return cell;
}
```

6 解释一下 http，如何下载一张图片代码实现

超文本传输协议（英文：HyperText Transfer Protocol，缩写：HTTP）是互联网上应用最为广泛的一种网络协议。

7.简述对 view，viewController，calayer 和 window 的理解

UIView:属于 UIKit.framework 框架,负责渲染矩形区域的内容,为矩形区域添加动画,响应区域的触摸事件,布局和管理一个或多个子视图 UIWindow:属于 UIKit.framework 框架,是一种特殊的 UIView,通常在一个程序中只会有一个 UIWindow,但可以手动创建多个 UIWindow,同时加到程序里面。UIWindow 在程序中主要起到三个作用:

1、作为容器,包含 app 所要显示的所有视图 2、传递触摸消息到程序中 view 和其他对象 3、与

UIViewController 协同工作,方便完成设备方向旋转的支持

CALayer:属于 QuartzCore.framework,是用来绘制内容的,对内容进行动画处理依赖与 UIView 来进行显示,不能处理用户事件。UIView 和 CALayer 是相互依赖的,UIView 依赖 CALayer 提供内容,CALayer 依赖 UIView 一共容器显示 绘制内容。延伸: UIViewController:管理视图的几成熟,每个视图控制器都有一个自带的视图,并且负责这个视图相关的一切事务。方便管理视图中的子视图,负责 model 与 view 的通信;检测设备旋转以及内存警告;是所有视图控制类的积累,定义了控制器的基本功能。

一、 Objective – C 及面向对象基础（30 分）

1. Objective-C 中有无私有成员？有无私有方法？如何区分类方法和实例方法？

objective-c 类里面的方法只有两种，类方法和实例方法.但是可以通过把方法的声明和定义都放在.m 文件中来实现一个表面上的私有方法。有私有变量,可以通过@private 来修饰,或者把声明放到.m 文件中。在 Objective-C 中,所有实例变量默认都是私有的,所有实例方法默认都是公有的 +开头是类方法 – 开头是实例方法

2. Block 与函数有何异同？block 有何优点？

Block 是没有名字的匿名函数，类似于 C 语言中的函数指针，可以作为参数进行传递用于回调，block 可以定义在方法中，函数一般不可以这样定义，

Block 语法简介，可以在方法中定义实现，可以方法访问内部的局部变量使代码更加紧凑 结构化

3. Objective-C 中发送消息（如，[object message]），与 c++成员方法或者是 c 的成员方法调用，有何区别和联系？

OC 中的发送消息实际上就是调用方法 消息就是方法
发送消息的过程就是执行方法的过程

OC 的实例方法需要对象指针调用，而 c 的函数或者 c++方法不需要

4. 以下代码有什么问题？（5 分）

```
(将一个 int 类型的参数转换为 NSNumber 类型，并返回)
- (NSNumber *) number: (int) num
{
    return  [[NSNumber alloc] initWithInt:num];
}
```

代码中应该用 autorelease 进行自动释放

5. 有如下两个类 Class A 和 Class B 的声明和实现：（10 分）

```
@interface Class A: NSObject
- (void) print;
@end
@implementation Class A
-(void)print
```

```

{
    NSLog(@"this is Class A");
}
@interface Class B: NSObject
- (void) print;
@end
@implementation Class B
-(void)print
{
    NSLog(@"this is Class B");
}

```

问：以下代码中，变量 obj 调用 print 时，输出的结构各是多少？是否相同，为什么会出现这样的结果？

```

id obj = nil;
obj = [[Class A alloc]init];
[obj print];
[obj release];
obj = [[Class B alloc]init];
[obj print];
[obj release];
Obj = nil;

```

结果: this is Class A this is Class B

因为首先 obj 执行的时 ClassA 类的对象，所以调用的方法是 ClassA 类的实例方法 print 然后 又对 obj 重新赋值指向了 ClassB 类的对象 这时调用的 ClassB 类的 print

二、 计算机数据结构及算法基础（30 分）

1. 说明 UIViewController 类成员函数：

initWithNibName: bundle;

/dealloc/loadView/viewDidLoad/viewDidUnload 调用顺序，每个方法应该做的工作是什么？（5 分）

1.InitWithNibName: bundle:

初始化 UIViewController，执行关键数据初始化操作

2.loadView: 当访问 UIViewController 的 view 属性时，view 如果此时是 nil，那么 VC 会自动调用 loadView 方法来初始化一个 UIView 并赋值给 view 属性

3.viewDidLoad: 当 VC 的 view 对象载入内存后调用，用于对 view 进行额外的初始化操作

4.viewDidUnload:

在 VC 的 view 对象从内存中释放之后调用，可以在 view 被释放后做一些 view 相关的引用清理操作，此时 view 为 nil。在 iOS6.0 开始就废弃了，该方法不再会调用

5.dealloc: UIViewController 对象销毁的时候调用

-
2. 说明 UIView 类属性 frame/center/bounds 的联系与区别, 改变 bounds 的值会不会对 frame 和 center 造成改变;

Frame 指的是: 该 view 在父 view 坐标系统中的位置和大小 (参照点是父类的坐标系统)

Bounds 指的是: 该 view 在本身坐标系统中的位置和大小 (参照点是本身的)

Center: 相对于父视图坐标系的中心点坐标

改变 bounds 的值不会对 frame 和 center 造成改变

3. 请说明计算机中栈和堆得含义及他们的用途 (5 分);
4. 说说快速排序算法的基本思路或过程;
5. 不用存取器的方法, 重写 TestClass 类: (10 分)

```
@interface TestClass : NSObject{
    NSString *str;
}
@property(retain)NSString *str;
@end
```

```
@implementation TestClass
@synthesize str = _str;
@end
```

答:

```
@interface TestClass : NSObject{
    NSString *str;
}
- (void)setStr: (NSString *) str;
- (NSString *)str;
@end
```

```
@implementation TestClass
- (void)setStr: (NSString *) str{
    if(str!=_str){
        [_str release];
        _str = [str retain];
    }
}
- (NSString *)str{
    return[ str autorelease];
}

```

```
@end
```

知识面 HR HumanResource

1. 说说你所了解以及日常使用的版本控制系统，以及它们的优缺点

SVN 优点：除包括了 CVS 的大多数特点外，还有一些如：文件目录可以方便的改名、基于数据库的版本库、操作速度提升、权限管理更完善等功能非常好用。

SVN 缺点：代码库不是分布式，而是集中在一个地方，所以导致跨国访问的时候速度很慢，远程开发非常不利。其次由于代码库只有一份，特别是元数据都存在服务器上，所以当服务器崩溃的时候损失严重。

GIT 优点：是免费、开源、它采用了分布式版本库的方式，不必服务器端软件支持，使源代码的发布和交流极其方便。每一个 GIT 克隆都是一个完整的文件库，含有全部历史记录和修订追踪能力。其最大特色就是“分支”及“合并”操作快速、简便。支持离线工作，GIT 是整个项目范围的原子提交，而且 GIT 中的每个工作树都包含一个具有完整项目历史的仓库。

GIT 缺点：对 windows 以及中文的支持不是很好。

2. 你通常以怎样的数据格式跟服务端做数据交换

JSON、XML,

Swagger/JsonSchema/ 开源库

Gdata,

http

3. 写下你掌握的其他编程语言

C、C++、C#、JAVA、PHP、Ruby、

javascript, Lua, Python 等

4. 你了解的第三方库，常用哪些 Framework

ASIHTTPRequest、TXML、TouchJSON、MBProgressHUD、

EGOTableViewPullRefresh、FMDB、ZBarSDK、MinZIP

5. 你了解的设计模式

MVC设计模式

单例模式

代理模式

通知模式

工厂模式

适配器模式

简单介绍集中设计模式：

1) 模型视图控制器 (MVC)。控制器负责行为，模型提供数据源，视图显示UI。模型和视图之间尽量不要直接打交道，他们之间的交互应该通过控制器来进行，控制器充当着桥梁的作用。这样设计的目的是使不同功能的类之间尽量解耦，以利于程序的扩展。

2) 代理模式

委托代理 (deegate)，顾名思义，把某个对象要做的事情委托给别的对象去做。那么别的对象就是这个对象的代理，代替它来打理要做的事。反映到程序中，首先要明确一个对象的委托方是哪个对象，委托所做的事情是什么。这里所做的事情是靠协议中的方法来实现，方法分两种：必需实现 (@required) 的方法和根据情况选择实现 (@optional) 的方法。

举个例子：你是房屋租赁中介，某个房东和你签订协议，请你替他房子出租出去。这时，你就是房东的代理，你必须实现的方法是把房子出租出去，选择实现的方法是装修、添置家具、打隔断等（依据协议而定）。

3) 通知模式

通知模式是观察者模式的一种。a对象在通知中心注册了观察者之后，b对象

发出通知广播，a对象收到通知后就知道去做具体的事。观察者可以是一个或多个，也可以没有。举个例子：微博切换帐号后会发出一个通知，让多个界面重新刷新数据。

4) KVO模式

Key-Value-Observer模式也是观察者模式的一种。KVO的机制为：当指定的被观察对象的属性被修改的时候，KVO都会自动的去通知相应的观察者。举个例子，在控制器里通过addObserver:forKeyPath:options:context:注册一个数据源观察者，当数据源里的数据发生变化时，通过willChangeValueForKey:和didChangeValueForKey:这一对方法发出广播，控制器收到广播后就可以利用新的数据来刷新界面。

5) 单例模式

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。

6. 平时会代码复用，或经常重构自己的代码吗？

会，为了提高软件质量、性能、软件的扩展性和维护性、使其更加合理、

UIImageView 不能点击

框架 封装，继承 多态

iOS SDK:

1. 如何实现 Push 通知的机制（大致写下）

1 用网站把要发送的消息、目的 iPhone 的标识打包，发给 APNS。

2 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，并把消息发到 iPhone。

3 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

2. Key-Value Coding 是什么?

KVC 是通过字符串标识符间接访问对象属性和关系的机制。

5. 说说 Auto Release Pool，内存管理模式, GC, retain count

auto release pool 内存自动释放池。放到池中的会在失效后自动释放

内存管理模式：可以手动申请释放，使用 auto release pool，使用 ARC

GC: Gabbage Collector 垃圾回收

iOS 有没有 GC Gabage Collector 垃圾回收机制

mac os x 有 ios 没有

retain count: 内存计数器

技术题:

1. 怎样实现一个 Memory Cache

alloc 太耗时间

```
malloc(100);
```

```
malloc(200);
```

2. 如何在有一个属性名和值的情况，给一个对象赋值

对象名.属性名 = 值

```
obj.name = v;
```

3. GBK 如何转换为 UTF-8 (3 字节)

如何转成 Unicode. (2 字节)

```
NSStringEncoding eco=CFStringConvertEncodingToNSStringEncoding(
```

```
    kCFStringEncodingGB_18030_2000);
```

```
urlStr = [urlStr
```

```
    stringByAddingPercentEscapesUsingEncoding:eco];
```

```
NSURL * url=[NSURL URLWithString:urlStr];
```

4. 如何实现一个带删除线的 UILabel

~~用 CG 重绘 drawRect~~

~~RTLabel 开源的富文本~~

5. 如何让 imageNamed 同时兼容 3.x 及 4.x 系统，以及不失去缓存

imageNamed 自动缓存不会消失

按照 3.X 和 4.X 的分辨率做两张图片，图片名相同需要支持 4.X 的图片名后加上 @2X

test.png test@2x.png

```
NSString *path = [[NSBundle mainBundle]
pathForResource:@"xxx" ofType:@"png"];
UIImage *iv = [[UIImage alloc]
initWithContentsOfFile:path];
```

6. 如何实现一个异步的 UIImageView

开启异步线程下载图片，或用三方库进行一步操作

7. 如何实现像 Settings 一样的界面

用 TableView、UISwitch

8. 如何实现滚动到底部自动加载，以及显示一个 loading cell

用 scrollViewDidEndDragging:(UIScrollView *)scrollView

willDecelerate:(BOOL)decelerate 方法计算是否触底，调用异步刷新事件

scrollViewDidScroll

9. 如何实现从外部打开程序

在项目的 Info.plist 文件中 添加一个 URL types 节点就可以了

```
[[UIApplication sharedApplication] openURL:[NSURL
URLWithString:@"http://www.baidu.com"]];
[[UIApplication sharedApplication] openURL:[NSURL
URLWithString:@"tel://18533221222"]];
```

```
[[UIApplication sharedApplication] openURL:[NSURL  
URLWithString:@"sms://18533221222"]];
```

10. 如何将图片添加到照片库

```
UIImage *img = [UIImage imageNamed:@"123.png"];  
UIImageWriteToSavedPhotosAlbum(img, nil, nil, nil);
```

11. copy 与 retain 的区别

copy 复制一个对象

retain 会让计数器+1

12. Undo, Redo 如何实现

UIMenuItem

13. NSCoder (archive, unarchive)

```
- (id) initWithCoder:(NSCoder *)coder;
```

```
-(void) encodeWithCoder:(NSCoder*)coder;
```

加分题

1. 如何实现一个实现渲染自定义格式字符串的 UILabel

CoreText. RTLabel

2 Protocol 与 performSelector: 的相同与区别，哪个更好（基础参数类型传递）

Protocol 是协议，同时也走了 NSObject 的协议在 NSObject 协议中定义了很多基本的协议函数，其中包括 performSelector。

3. 如果使用多线程进行编码

```
[NSThread detachXXXX];  
[self performSelectorInBackground:@selector(xxx)  
withObject:xxx];  
+ (void)detachNewThreadSelector:(SEL)selector  
toTarget:(id)target withObject:(id)argument;
```

奇创微迅

实现多线程方式;

NSThread **NSOperation** **GCD**

代理和通知作用与区别;

Delegate:把某个对象要做的事情委托给别的对象去做。那么别的对象就是这个对象的代理,代替它来打理要做的事。反映到程序中,首先要明确一个对象的委托方是哪个对象,委托所做内容是什么。

Notification 是观察者模式的实现,**Notification** 通过 **NotificationCenter** 和 **remove observer** 工作。

你工作较真吗?

认真

数据存储的几种方式：

普通文件

plist

NSUserDefaults

归档

数据库：Sqlite CoreData

介绍一下 socket。

套接字，是支持 TCP/IP 的网络通信的基本操作单元，可以看做是不同主机之间的进程进行双向通信的端点，简单的说就是通信的两方的一种约定，用套接字中的相关函数来完成通信过程

Cocos2d 中背景和精灵能否在一个类里加入到 CCSpriteBatchNode 中；

有没有看过 CCNode 这个类，简单介绍一下

用到过物理场景吗（大概就是重力场什么的）；

做 Cocos2D 有什么收获；

介绍一下播放音效（回答 simpleAudioEngine 还不行）；

Cocos2D 场景切换的知识（切换方式 fade 什么之类的）

融易通信面试题

- 1、 面向对象的特点
封装 继承 多态

- 2、 动画实现一个 view 从顶部移动到底部

```
UIView *view = [[UIView alloc]
initWithFrame:CGRectMake(0,0, 100, 100)];
[self.window addSubview:view];
```

```
[UIView animateWithDuration:1 animations:^(
    view.center =CGPointMake(50, 480-50);
} completion:nil];
```

- 3、 字符串 a-g, 截取 def 并逆序

```
NSString *str = @"abcdefg";
NSString *subStr = [str substringWithRange:NSMakeRange(3,
3)];
```

```
NSString*resStr = @"";
```

```
for (NSInteger i = subStr.length-1; i >= 0; i--) {
    unichar c = [subStr characterAtIndex:i];
    resStr = [resStr stringByAppendingFormat:@"%C",c];
}
```

- 4、 写一个获取日期的函数，输出格式 2012-06-06

```
NSDate *date = [NSDate date];
```

```
NSDateFormatter *dateFm = [[NSDateFormatter alloc] init];
```

```
dateFm.dateFormat = @"yyyy-MM-dd";
NSString *dateStr = [dateFm stringFromDate:date];
```

```
NSLog(@"dateStr:%@",dateStr);
```

尚为视讯

- 1、 iPhone 开发的两种方式是什么，各自的优缺点。

iPhone 开发可以有两种方式，即 **Web 开发**和 **SDK 编程**。这两种开发方式各有利弊，但他们都能为应用程序提供丰富、强大的功能。由于两种开发模型在本质上存在差异，所以他们各自适用的开发情形也不相同。**Web 开发**适合创建功能简单、动态且需要因特网连接的应该程序，而 **SDK 编程**适合创建功能复杂、强大的应用程序。

- 2、 OC 的两种内存管理方式，iOS 都支持吗。

手动内存管理 都支持

ARC 只有从 iOS5 之后才支持

- 3、 Notification 是什么，优缺点是什么，用什么可以代替。

Notification 是通知

观察者模式，controller 向 defaultCenter 添加自己的 notification，其他类注册这个 notification 就可以收到通知，这些类可以在收到通知时做自己的操作（多观察者默认随机顺序发通知给观察者们，而且每个观察者都要等当前的某个观察者的操作做完才能轮到他来操作，可以用 NotificationQueue 的方式安排观察者的反应顺序，也可以在添加观察者中设定反映时间，取消观察需要在 viewDidLoad 跟 dealloc 中都要注销）

可以用代理实现

- 4、 用 OC 写一个单例。

//定义静态全局变量

```
static Singleton * single = nil;
```

```
+ (Singleton *)sharedSingleton{
```

//考虑线程安全

```
@synchronized(self){  
    if (single == nil) {
```

```
        single = [[self alloc] init];
    }
}

return single;
}
```

5、iOS 是如何管理内存的。

6、OC 有多重继承吗，如何实现。

推博网

1、写出一些常见的库，你用的操作系统，各个操作系统的区别
asi, json, gdata, sdwebimage

1、iOS4.x 以后

1.1 多任务处理

参见 iOS 多任务.pdf

1.2 blocksObject

参见 block 编程要点. pdf

1.3 GCD

参见 GCD 并发处理.pdf

1.4 local Notification

参见 UILocalNotification.doc

1.5 Core Telephony Framework

参见 CoreTelephony.doc

1.6 iad 广告系统

参见 iad 文件夹里的文档

2、iOS5.x 以后

2.1 UIKit 中的控件增加外观属性修改的 API

用例请参考链接地址:

<http://www.cocoachina.com/newbie/tutorial/2012/0113/3876.html>

2.2 自动引用计数 (ARC)

参考链接:

<http://blog.csdn.net/zkdemon/article/details/7446385>

2.3 Storyboards

参见 Storyboards. pdf

2.4 NSJSONSerialization(iOS5 之后, 系统自带的 json 解析 API)

iOS5 以上版本应该选择系统的 API 进行 json 解析, 在解析 / 转化效率上完胜开源库.

不能使用系统 API 的应该选择开源解析库 JSONKit 进行解析工作.

2.5 Core Image Framework 图形渲染、图形处理

参考链接:

<http://www.cocoachina.com/iphonedev/sdk/2011/1115/3503.html>

参考 code: CoreImage

3、iOS6.x 以后

3.1 Maps 苹果自家地图

参见 LocationAwarenessPG. pdf

3.2 Social Framework (集成了新浪微博 API)

参见 Social_Framework. pdf

3.3 UICollectionView (内存重用机制, 图片展示瀑布流实现)

参见 CollectionViewPGforIOS. pdf

参考链接:

<http://www.raywenderlich.com/zh-hans/24595/%E5%9C%A8-ios-6%E5%BC%80%E5%A7%8B-uicollectionview-part-12>

参考 code: CollectionView-Simple

3.4 UIStatePreservation(app 状态保存)

3.5 Auto Layout (自动布局属性)

参见 AutolayoutPG. pdf

参考链接:

<http://www.raywenderlich.com/zh-hans/22873/ios-6-%E8%87%AA%E5%8A%A8%E5%B8%83%E5%B1%80-%E5%85%A5%E9%97%A8%E5%BC%8D1>

iOS7.x

系统

1, 全新的 ui 设计

2, Control Center 控制中心

3, 全新的多任务

4, AirDrop 共享文件

5, 全新的 Safari 浏览器交互体验

SDK

-
- 1, 动态 uikit 新增了 uidefinitionitem 委托 用来为 UIView 定制动态行为
 - 2, 新增游戏引擎 sprite kit Framework 框架 类似于 cocos2d

2、UITableView 有时候不太流畅, 有什么优化方案

- a 复用单元格
- b 使用不透明的师徒 单元格中尽量少用动画
- c 图片加载使用异步加载 并设置图片加载的并发数
- d 滑动时不加载图片 定制滑动开始加载
- e 文字 图片可以直接 drawinrect 绘制
- f 如果不是必须的, 减少使用 reloadData 全部 cell
- g 如果 cell 是动态行高, 计算出高度后缓存
- h cell 高度固定直接用 cell.rowHeight 设置高度

3、UIImageView 怎么使其有单击, 双击事件

打开用户交互属性, 使用点击手势, 设置单击和双击

并且如果既有单击 又有双击 那么要区分

```
for (UIGestureRecognizer *gest in self.gestureRecognizers)
{
    if ([gest isKindOfClass:[UITapGestureRecognizer
class]]) {
        //判断单击手势
        if (((UITapGestureRecognizer
*)gest).numberOfTapsRequired == 1) {

            //当tap手势 失败的时候 gest 才有效
            //增加这个函数之后 单击会延迟执行
            [gest requireGestureRecognizerToFail:tap];
            //只有双击失败了单击才有效

            //如果这里不设置 那么 在双击的时候 单击手势也会生效

            break;
        }
    }
}
```

6、将一组数据永久保存到手机里，怎样做？

NSUserDefaults, plist, 数据库 普通文件 归档

7、实现一个横向划动的表格，有什么思路

继承 UIScrollView

8、.m 里面的私有方法其他的类能不能调用，若强行调用会由什么问题

在非 ARC 下编译器会报警 强行执行可以调用

在 arc 下编译器会报错，没法调用

新浪乐居笔试题

1. 做 iOS 多久了，拥有哪些苹果设备？

4 年 itouch iPhone iPod iPad iPadmini MacBook MacBook Pro
macMini

2. 去年你最喜爱的技术图书是哪些？从中你学到了什么？

3. 平时你经常访问哪些编程类网站？

<http://www.w3school.com.cn>

代码库

<http://www.oschina.net/ios/codingList/>

<http://code4app.com>

最大的 iOS 中文开发网站

<http://www.cocoachina.com>

(源码共享网站)

<https://github.com/>

//stackoverflow flow :<http://www.stackoverflow.com> (有解决不了的难题时，这个网站可能会带给你惊喜)

4. 为什么选择 iOS 开发，你对它有什么想法？

5. 你有自己的开源项目吗？用什么托管的代码？

有 github

6. 在 OC 里类的方法和属性有几种访问权限，分别是什么？

OC 类中的方法都是公有的

属性有@private @protected @public

7. 在 OC 里 alloc 和 retain 语义相反的方法是?

Release

8. Release 的作用是什么和 autorelease 有什么区别? 什么时候用到 release, 什么时候用 autorelease?

Release 使对象计数器-1 放弃绝对使用权

Autorelease 延后释放对象 把对象地址放入自动释放池, 交给自动释放池管理对象释放

一般对象指针用完之后就应该立即 release, 但是有些时候不能用 release 比如实现一个方法创建对象后返回对象的地址, 这个时候是不能 release 的只能用 autorelease

9. 在简化的 Apple iOS 架构中, 总共有多少层?

iOS 的系统架构分为四个层次: 核心操作系统层 (Core OS layer)、核心服务层 (Core Services layer)、媒体层 (Media layer) 和可触摸层 (Cocoa Touch layer)。

10. 什么是 MVC, iOS 开发中起哪些作用?

MVC 就是 Model-View-Controller 的缩写, M 指的是业务模型, V 指的是用户页面, C 指的是控制器。MVC 是架构模式, 是讲 M 和 V 的代码分离, 从而使同那个一个程序可以使用不同的表现形式。 M: 表示数据和业务规则, V 是用户看到的并与之交互的页面, C 是接受用户的输入并调用 M 和 V 取完成用户需求的

11. 应用程序沙盒包含哪四个专门用于存储应用程序数据的目录?

documents, tmp, app, Library

12. 使用 Core Location 时, 如何避免电池续航时间急速缩短?

13. 使用哪个事件来检测用户请点击按钮？

14. 如何实现接触程序北京隐藏在 `textField` 变量的文本框显示的软键盘？
取消 `textField` 的第一响应

15. 写出你知道的设计模式，以及在实际开发当中用过哪些？

MVC设计模式
单例模式
代理模式
通知模式
工厂模式
适配器模式

16. 为什么要用单例模式？单例模式的好处是什么？
传值 共享数据，整个程序可以共享单例的数据

17. 执行代码后，每行 `string` 对象的 `retain count` 分别是多少？为什么？

```
NSString *string = [[NSString alloc] init];  
[string retain];  
[string release];  
[string release];
```

unsigned long 最大值 如果按照%ld 打印结果是-1

[[NSString alloc] init] 创建的是一个空字符串对象，没有必要在堆空间开辟空间，这个时候对象计数器是失效的 `retain` 或者 `release` 没有任何作用

18. 写出下列代码的输出结果

```
int a[5] = {1,2,3,4,5};  
int *b = (int *)(&a+1);  
printf("%d %d",*(a+1),*(b-1));
```

2 5

19. 使用 Objective-C 写一个冒泡算法

//冒泡

```
for (NSInteger i = 0; i < [stus count]-1; i++) {
    for (NSInteger j = 0; j < [stus count]-1-i; j++) {
        //获取第j j+1学生的地址
        Student *stu1 = [stus objectAtIndex:j];
        Student *stu2 = [stus objectAtIndex:j+1];
        if ([stu1 score] < [stu2 score]) {
            //交换
            [stus exchangeObjectAtIndex:j
withObjectAtIndex:j+1];
        }
    }
}
```

20. 写一段代码，用异步请求的方式抓取一个网站的内容，请考虑超时，状态码的异常情况。

1 简述工厂模式的原理

在基类中定义创建对象的一个接口，让子类决定实例化哪个类。工厂方法让一个类的实例化延迟到子类中进行。

工厂方法要解决的问题是对象的创建时机，它提供了一种扩展的策略，很好地符合了开放封闭原则

2 oc 支持多继承吗？支持多接口吗？重写一个类的方式用继承好，还是分类好？

Object-c 的类不可以多重继承；可以实现多个接口，通过实现多个接口可以完成 **C++** 的多重继承；**Category** 是类别，一般情况用分类好，用 **Category** 去重写类的方法，仅对本 **Category** 有效，不会影响到其他类与原有类的关系。

3 iPad 和 iPhone 开发有什么不同

开发 **iPad** 和 **iPhone** 应用使用的是相同的 **SDK**，它们之间的区别主要体现在设备硬件和 **UI** 操作上。(1)设备硬件：由于 **iPad** 不具备电话功能，**iPad1** 也没有相机，所以相关的特性如电话、**SMS** 等无法使用。(2)**UI** 操作:由于 **iPad** 屏幕大小为 **9.7** 英寸，而 **iPhone** 的屏幕是 **3.5** 英寸，这就决定了两者操作方式的不同。有些控件，如 **UIPopoverController**, **UISplitViewController** 只能在 **iPad** 中使用，而不能在 **iPhone** 中使用；而且当需要 **presentModalViewController** 时，**Window presentation style** 只对 **iPad** 有效；除此之外，**iPad** 产品开发中还需针对 **iPad** 大屏幕考虑加入多指手势以增加用户体验。

4 堆和栈的区别

5 关键字 volatile 有什么含义，并给出例子

6 iPhone 阅读器要读取个文本文件，是如何处理编码问题的？ 然后 pdf 格式的文件如何读取

7 iPhone 有哪些传值方式？ View 和 view 之间呢

属性传值 代理传值 **block** 传值 单例传值 通知传值 单例传值 等等

8 定义属性时，说明不同情况下如何使用 **copy** ， **retain** 和 **assign**？

一般都是使用 **retain**

字符串用 **copy**

设置代理用 **assign**

believe oneself

(深圳皆凯科技有限公司笔试题)

1、objective-c中是所有对象间的交互是如何实现的?

通过代理去进行通讯, 或者通过观察者消息模式, Blocks , appDelegate

2、如何将产品进行多语言发布, 开发?

程序国际化;比如:本地化应用程序名称(1、选中工程,Info—Localizations点击“+”

添加要国际化的语言。(2、在InfoPlist.strings右边会多出一个三角形,点击展开可

看到InfoPlist.strings(english)和

InfoPlist.strings(chinese)两个版本的文件;(3、在InfoPlist.strings(english)文件中加

入:

CFBundleDisplayName = "Program"; 其中“Program”为英文应用程序名称,同理在

InfoPlist.strings(chinese)文件中加入: CFBundleDisplayName = "应用程序"; 其中

“应用程序”为中文名称,注意:CFBundleDisplayName加不加双引号都行;

(4、编辑Info.plist,添加一个新的属性Application has localized display name, 设置

其类型为boolean,并将其 value设置为YES即可。

3、objective-c中是如何实现线程同步的?

Mutexlock, NSCondition lock, 消息传送

使用锁:锁是线程编程同步工具的基础。锁可以让你很容易保护代码中一大块区

域以便你可以确保代码的正确性。使用POSIX互斥锁;使用NSLock类;使用

@synchronized指令等。

4、编写函数_memmove说明如下: 实现C语言库函数memmove的功能: 将一块缓冲区中的数据移动到另一块缓冲区中。可能有重复。

//

5、什么叫数据结构? (面试顺风快递iphone开发)

计算机 = 编程+数据结构

6、编程中，保存数据有哪几种方式？数组、自定义类、文件操作、数据库如：SQLite

文件 plistNSUserDefaults sqlite

7、Sizeof与strlen的区别和联系？

8、什么是抽象类？抽象类有什么作用？能实例化吗？

NSObject 没有完全实现的类叫抽象类，C++，java

抽象类不能实例化，不能 NSObject *a =[[NSObject alloc] init];

9、用Objective-C写冒泡法。

//

10、你对@interface和@property的理解？

@interface声明一个类

@property就是申明2个函数，getter setter，编译器会自动声明2个函数

11、Iphone开发中控制类有什么作用？

传递数据 展示数据 数据模型和View的桥梁

12、线程中有哪些函数？写出来。

+ (NSThread *)currentThread;

+ (void)detachNewThreadSelector:(SEL)selector
toTarget:(id)target withObject:(id)argument;

+ (BOOL)isMultiThreaded;

@property (readonly, retain) NSMutableDictionary
*threadDictionary;

+ (void)sleepUntilDate:(NSDate *)date;

+ (void)sleepForTimeInterval:(NSTimeInterval)ti;

+ (void)exit;

+ (double)threadPriority;

+ (BOOL)setThreadPriority:(double)p;

@property double threadPriority NS_AVAILABLE(10_6, 4_0); //
To be deprecated; use qualityOfService below

@property NSQualityOfService qualityOfService
NS_AVAILABLE(10_10, 8_0); // read-only after the thread is
started

```
+ (NSArray *)callStackReturnAddresses NS_AVAILABLE(10_5,  
2_0);  
+ (NSArray *)callStackSymbols NS_AVAILABLE(10_6, 4_0);  
  
@property (copy) NSString *name NS_AVAILABLE(10_5, 2_0);  
  
@property NSUInteger stackSize NS_AVAILABLE(10_5, 2_0);  
  
@property (readonly) BOOL isMainThread NS_AVAILABLE(10_5,  
2_0);  
+ (BOOL)isMainThread NS_AVAILABLE(10_5, 2_0); // reports  
whether current thread is main  
+ (NSThread *)mainThread NS_AVAILABLE(10_5, 2_0);  
  
- (instancetype)init NS_AVAILABLE(10_5, 2_0)  
NS_DESIGNATED_INITIALIZER;  
- (instancetype)initWithTarget:(id)target  
selector:(SEL)selector object:(id)argument  
NS_AVAILABLE(10_5, 2_0);  
  
@property (readonly, getter=isExecuting) BOOL executing  
NS_AVAILABLE(10_5, 2_0);  
@property (readonly, getter=isFinished) BOOL finished  
NS_AVAILABLE(10_5, 2_0);  
@property (readonly, getter=isCancelled) BOOL cancelled  
NS_AVAILABLE(10_5, 2_0);  
  
- (void)cancel NS_AVAILABLE(10_5, 2_0);  
  
- (void)start NS_AVAILABLE(10_5, 2_0);  
  
- (void)main NS_AVAILABLE(10_5, 2_0); // thread body method  
  
@end  
  
FOUNDATION_EXPORT NSString * const  
NSWillBecomeMultiThreadedNotification;  
FOUNDATION_EXPORT NSString * const  
NSDidBecomeSingleThreadedNotification;  
FOUNDATION_EXPORT NSString * const  
NSThreadWillExitNotification;  
  
@interface NSObject (NSThreadPerformAdditions)
```

```

- (void)performSelectorOnMainThread:(SEL)aSelector
withObject:(id)arg waitUntilDone:(BOOL)wait modes:(NSArray
*)array;
- (void)performSelectorOnMainThread:(SEL)aSelector
withObject:(id)arg waitUntilDone:(BOOL)wait;
    // equivalent to the first method with
    kCFRunLoopCommonModes

- (void)performSelector:(SEL)aSelector onThread:(NSThread
*)thr withObject:(id)arg waitUntilDone:(BOOL)wait
modes:(NSArray *)array NS_AVAILABLE(10_5, 2_0);
- (void)performSelector:(SEL)aSelector onThread:(NSThread
*)thr withObject:(id)arg waitUntilDone:(BOOL)wait
NS_AVAILABLE(10_5, 2_0);
    // equivalent to the first method with
    kCFRunLoopCommonModes
- (void)performSelectorInBackground:(SEL)aSelector
withObject:(id)arg NS_AVAILABLE(10_5, 2_0);

```

1 3、有没有写过自定义的控件？

自己定义控件有2类，1类比较多的就是类似

```

RatingView
UIJewels : UIButton
{
    int x, y;
    int type;
    UIImage *image;
}
UIImageView和UIImage UILabel和NSString

```

2. 用CG CoreGraphics drawRect:

|-----| 每个UIView都有一个drawRect:

1 4、调用一个类的静态方法需不需要release？

静态方法就是类方法, 不需要, 类的方法
对象放在autorelease中

1 5、do-while 与 while-do的区别？

1 6、写出几个死循环？ while (1), for (;),

用预处理指令**#define** 声明一个常数，用以表明 1 年中有多少秒（忽略闰年问题）

```
#define SECONDS_PER_YEAR (60 * 60 * 24 * 365)UL
```

下面是笔试常考题

- a) `int a;` // An integer
- b) `int *a;` // A pointer to an integer
- c) `int **a;` // A pointer to a pointer to an integer
- d) `int a[10];` // 一个有 10 个指针的数组，该指针是指向一个整型数的。
- e) `int *a[10];` // 一个有 10 个指针的数组，该指针是指向一个整型数的。
- f) `int (*a)[10];` // 一个指向有 10 个整型数数组的指针
- g) `int (*a)(int);` // 一个指向函数的指针，该函数有一个整型参数并返回一个整型数。
- h) `int (*a[10])(int);` // 一个有 10 个指针的数组，该指针指向一个函数，该函数有一个整型参数并返回一个整型数。

1. assign 和 retain, copy 的区别

assign 就是和普通语言的赋值一样

retain 在一个对象中保存该对象需要拥有的对象
对该对象进行应用计数+1

@property (retain) 为了写程序方便

copy 主要是赋值

一般 代理的申明都是 **assign** 为什么
原因有 2 个，

- 1 根本就没有必要把代理的计数器+1
- 2. 主要是为了避免引用技术嵌套，递归

2. 如何持久保存数据

NSUserDefaults, Plist, SQLite3, file

3. 网络方面得 比如如何检测联网 http xml 等

ASIHTTPRequest Reacheable.[hm]

原理是 **ifconfig** 是不是有 ip 地址

Reachability

来判断是否有网络 3G Wifi

4. 类之间得继承关系 比如 **UIView** 得父类是哪一个等等？

UIResponder，处理触摸，

touchBegin, touchEnd, nextResponder

5. 异常处理

```
@try {} @catch {}
```

在同一个类 **A** 中，有方法 **a** 和方法 **b**，方法 **b** 如何调用方法 **a**？

```
[self a];
```

有 **A**、**B** 两个类，**B** 类继承于 **A** 类，都实现了方法 **xx**。创建 **B** 类实例 **b**，如何让 **b** 调用 **A** 类？

```
[super a];
```

const 和 Volatile 分别代表什么意思？

Volatile 表示每次都从内存取数据

类型 id 代表什么意思？

面向对象包含哪些元素？

继承和分类的区别？

Catogory

Catogory 实现起来麻烦一些，但是使用方便

NSSet 类的作用是什么？

集合

编写一个完整程序，在控制台打印字符串“HelloWorld!”

用递归实现计算 n 的阶乘。（用 objective-c 语法实现）

//

static 全局变量与普通的全局变量有什么区别？static 局部变量和普通局部变量有什么区别？static 函数与普通函数有什么区别？

C 语言中讲讲 static 变量和 static 函数有什么作用

static 关键字有两种意思，你看上下文来判断

1, 表示变量是静态存储变量

表示变量存放在静态存储区。

2, 表示该变量是内部连接

（这种情况是指该变量不在任何 {} 之内，就象全局变量那样，这时候加上 static）

，也就是说在其它的 .cpp 文件中，该变量是不可见的（你不能用）。

当 static 加在函数前面的时候

表示该函数是内部连接，之在本文件中有效，别的文件中不能应用该函数。

不加 static 的函数默认为是全局的。

也就是说在其他的 .cpp 中只要申明一下这个函数，就可以使用它。

1、static 全局变量与普通的全局变量有什么区别？static 局部变量和普通局部变量有什么区别？static 函数与普通函数有什么区别？

答：全局变量（外部变量）的说明之前再冠以 static 就构成了静态的全局变量。全局变量本身就是静态存储方式，静态全局变量当然也是静态存储方式。这两者在存储方式上并无不同。这两者的区别虽在于非静态全局变量的作用域是整个源程序，当一个源程序由多个源文件组成时，非静态的全局变量在各个源文件中都是有效的。而静态全局变量则限制了其作用域，即只在定义该变量的源文件内有效，在同一源程序的其它源文件中不能使用它。由于静态全局变量的作用域局限于一个源文件内，只能为该源文件内的函数公用，因此可以避免在其它源文件中引起错误。

从以上分析可以看出，把局部变量改变为静态变量后是改变了它的存储方式即改变了它的生存期。把全局变量改变为静态变量后是改变了它的作用域，限制了它的使用范围。

static 函数与普通函数作用域不同。static 函数仅在本文件中使用。只在当前源文件中使用的函数应该说明为内部函数（static），内部函数应该在当前源文件中说明和定义。对于可在当前源文件以外使用的函数，应该在一个头文件中说明，要使用这些函数的源文件要包含这个头文件

static 全局变量与普通的全局变量有什么区别：**static** 全局变量只初始化一次，防止在其他文件单元中被引用；

static 局部变量和普通局部变量有什么区别：**static** 局部变量只被初始化一次，下一次依据上一次结果值；

static 函数与普通函数有什么区别：**static** 函数在内存中只有一份，普通函数在每个被调用中维持一份拷贝

2、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，假定你将那个变写错了，那么在编译期间 会报错，如果你用 **extern** 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。

3、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。

什么是 UDP 和 TCP 的区别是什么？

TCP 的全称为传输控制协议。这种协议可以提供面向连接的、可靠的、点到点的通信。

UDP 的全称为用户数据报协议，它可以提供非连接的不可靠的点到多点的通信。用 **TCP** 还是 **UDP**，那要看你的程序注重哪一个方面？可靠还是快速？

TCP/IP 建立连接的过程

在 **TCP/IP** 协议中，**TCP** 协议提供可靠的连接服务，采用三次握手建立一个连接。

第一次握手：建立连接时，客户端发送连接请求到服务器，并进入 **SYN_SEND** 状态，等待服务器确认；

第二次握手：服务器收到客户端连接请求，向客户端发送允许连接应答，此时服务器进入 **SYN_RECV** 状态；

第三次握手：客户端收到服务器的允许连接应答，向服务器发送确认，客户端和服务器进入通信状态，完成三次握手。

（所谓的三次握手就是要有三次连接信息的发送/接收过程。**TCP** 连接的建立需要进行三次连接信息的发送/接收。）

```
-(void) print:(NSString*) msg
{
    NSLog(@"%@",msg);
}
```

```
printf("%d,%c\n",i,c)
```

```
char str[10]="",str2[]={ "China"};
strcpy(str1,str2);
```

```
printf("%d",max(a,b));
```

把 `max(a,b)` 作为 `printf` 函数的一个参数。

objc-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 objc-c 中通过委托来实现。

线程与进程的区别和联系?

进程和线程都是由操作系统所体现的程序运行的基本单元,系统利用该基本单元实现系统对应用的并发性。

程和线程的主要差别在于它们是不同的操作系统资源管理方式。进程有独立的地址空间,一个进程崩溃后,在保护模式下不会对其它进程产生影响,而线程只是一个进程中的不同执行路径。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,但在进程切换时,耗费资源较大,效率要差一些。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。

自动释放池是什么,如何工作?

当您向一个对象发送一个 `autorelease` 消息时,Cocoa 就会将该对象的一个引用放入到最新的自动释放池。它仍然是个正当的对象,因此自动释放池定义的作用域内的其它对象可以向它发送消息。当程序执行到作用域结束的位置时,自动释放池就会被释放,池中的所有对象也就被释放。

1. objc-c 是通过一种"referring counting"(引用计数)的方式来管理内存的,对象在开始分配内存(`alloc`)的时候引用计数为一,以后每当碰到有 `copy,retain` 的时候引用计数都会加一,每当碰到 `release` 和 `autorelease` 的时候引用计数就会减一,如果此对象的计数变为了 0,就会被系统销毁。

2. `NSAutoreleasePool` 就是用来做引用计数的管理工作的,这个东西一般不用你管的。

3. `autorelease` 和 `release` 没什么区别,只是引用计数减一的时机不同而已,`autorelease` 会在对象的使用真正结束的时候才做引用计数减一。

什么是 KVC 和 KVO? 答: KVC(Key-Value-Coding)内部的实现: 一个对象在调用 `setValue` 的时候,(1) 首先根据方法名找到运行方法的时候所需要的环境参数。(2) 他会从自己 `isa` 指针结合环境参数,找到具体的方法实现的接口。

(3) 再直接查找得来的具体的方法实现。KVO (Key-Value-Observing): 当观察者为一个对象的属性进行了注册,被观察对象的 `isa` 指针被修改的时候,`isa` 指针就会指向一个中间类,而不是真实的类。所以 `isa` 指针其实不需要指

向实例对象真实的类。所以我们的程序最好不要依赖于 **isa** 指针。在调用类的方法的时候，最好要明确对象实例的类名。

kvc 就是一种通过字符串去间接操作对象属性的机制。

iphone 中，自定义协议？

Protocol，看代理 viewController

协议在 oc 中主要用在代理中

绑定的概念？

代理的概念？对它怎么理解？

为了模块之间的松耦合

网络编程中协议的概念。

NetworkDataDelegate;

3 个函数理解 didReceive,

苹果上架流程？

怎么样编一个通话软件只能在 **iphone** 上运行,在 **itouch** 上不能运行 如何实现

```
UIDevice *dev = [UIDevice currentDevice];  
if ([[dev model] isEqualToString:@"iPhone"]) {  
  
} else if ([[dev model] isEqualToString:@"iPod touch"]) {  
  
}  
if([dev userInterfaceIdiom] == UIUserInterfaceIdiomPad) {  
  
}
```

两个上海的朋友给的他们公司自己的面试题，大部分也都是旧题，我自己整理了下答案。。

面试题：

1.viewWillAppear viewWillDisappear, viewDidLoad 他们在什么时候使用。

viewWillAppear:视图即将可见时调用，默认情况下不执行任何操作。

viewWillDisappear: 视图消失时，覆盖或以其他方式隐藏时调用，默认不执行任何操作。

viewDidLoad: 在视图加载后调用，如果是在代码中创建的视图加载器，它将在loadView方法后被调用，如果是从nib视图页面输出，它将会在视图设置好后被调用。

2.frame和bounds的区别

frame:该view在父view坐标系统中的位置和大小，参照点是父视图的坐标系统。

bounds: 该view在本身坐标系统中的位置和大小，参照点是本身坐标系统。

3深拷贝和浅拷贝的区别，并简述如何实现一个NSObject子类的深拷贝。

浅拷贝：复制一个对象为副本。

深拷贝：开闭一块新的内存来存储副本对象。

想要实现深拷贝需要遵循<NSCopying>和<NSMutableCopying>协议，并实现代理方法。

4描述UITableViewCell的复用机制。

简单来说我认为这个复用机制是为了节省系统资源的开支。

就是我们在展示tableView的时候系统并不会一次把所有的cell给创建出来，而是创建一部分（具体是多少，我测试过是你能同时看到最多几个cell，就会创建几个cell）。这样当你滑动tableView的时候，一端的cell会滚出视图，另一端会出现新的cell。这个滚出视图的cell，在滚出视图后并没有被闲置，而是被放进了tableView的复队列中，与此同时，另一端出现的新的cell，就会从这个复用队列中取，若无则创建一个新的，若有则使用之。

5.写出三种常用的设计模式。

代理模式：一般和协议一块使用完成某些传值。例如tableView的delegate。

观察者模式：例如通知中心，注册通知中心，任何位置可以发送消息，注册观察者的对象可以接收。

单例模式：确保应用整个生命周期中，某个类只有一份实例，用于共享资源数据的控制。

6.NSArray *array = @[@"a",@"b",@"c",@"a",@"d"....],里面有N个string元素，求出array中唯一元素，要求复杂度为N。

如，@[@"a",@"b",@"b",@"c"],输出结果为@"abc";

@[@"a",@"a",@"b"] 输出结果为@"ab";

@[@"a",@"a",@"a"] 输出结果为@"a";

(ps: 就是把重复的元素去掉拼成一个字符串)

```
代码实例: NSArray *arr = @[@"a",@"b",@"b",@"c"];
NSString *resultStr = [arr objectAtIndex:0];
for (NSString *str in arr) {
    if ([resultStr rangeOfString:str].location == NSNotFound) {
        resultStr = [resultStr stringByAppendingString:str];
    }
}
NSLog(@"%@",resultStr);
```

1. #include和#import有什么区别, @class呢, #import<>跟#import""又有什么区别

include和import都是导入某个文件, 但import可以防止重复导入的问题。

@class 值是告诉编译器, 这是个类, 可以声明一个此类的对象指针, 但是具体类里边有什么属性, 接口编译器不知道。

#import<> 适用于导入系统的头文件

#import""适用于导入自定义的头文件

2. OC的类可以多继承么? Category是什么, 使用过吗?

不可以。但是可以通过协议间接实现。

category, 我们也叫做类别, 就是可以为某个类增加额外的方法, 包括系统的类和自定义的类。

3. 编写一个blocks使用的例子, 尽可能体现出block编程的语法和优势

定义一个带参数的blocks:

```
void (^logBlock)(NSString *msg);
logBlock = ^(NSString *str){
    NSLog(@"%@",str);//line 3
};
```

使用blocks:

logBlock(@"hello");//相当于把line 3的代码放在这里执行。

4. 在iPhone应用总如何保存数据? 有几种方式。

临时数据: 数组, 字典等。

持久化存储: Plist, 数据库, 沙盒。

5. c/c++/oc怎么混编, 要注意什么?

可以参考 <http://blog.csdn.net/gf771115/article/details/9664393>

我之前公司的面试题，整理了下答案

1、ObjC中，与alloc语义相反的方法是dealloc还是release？与retain语义相反的方法是dealloc还是release，为什么？需要与alloc配对使用的方法是dealloc还是release，为什么？

alloc与dealloc语意相反,alloc是创建变量,dealloc是释放变量。 retain 对应 release,retain 保留一个对象。调用之后,变量的计数加1。或许不是很明显,在这有例为证:

```
- (void) setName : (NSString*) name { [name retain];  
[myname release];  
myname = name;  
}
```

我们来看一看newname的计数是怎么变化的。首先,它被alloc,count = 1; 然后,在setName中,它被retain, count = 2; 最后,用户自己释放newname,count = 1,myname指向了newname。这也解释了为什么需要调用[myname release]。我们需要在给myname赋新值的时候,释放掉以前老的变量。retain 之后直接dealloc对象计数器没有释放。alloc 需要与release配对使用,因为alloc 这个函数调用之后,变量的计数加1。所以在调用alloc 之后,一定要调用对应的release。另外,在release一个变量之后,他的值仍然有效,所以最好是后面紧接着再var = nil。

2、在一个对象的方法里面:

```
self.name = "object";和  
name = "object"
```

有什么不同吗?

self.name = "object"会调用对象的setName()方法,name = "object"会直接把object赋值给当前对象的name 属性。

3、这段代码有什么问题吗:

```
@implementation Person  
- (void)setAge:(int)newAge {  
self.age = newAge;  
}  
@end
```

会进入死循环

4、什么是retain count?

引用计数(ref count或者retain count)。对象的内部保存一个数字,表示被引用的次数。例如,某个对象被两个指针所指向(引用)那么它的retain count为2。需要销毁对象的时候,不直接调用dealloc,而是调用release。release会让retain count减1,只有retain count等于0,系统才会调用dealloc真正销毁这个对象。

5、以下每行代码执行后, person对象的retain count分别是多少

```
Person *person = [[Person alloc] init]; 1
```

```
[person retain]; 2
```

```
[person release]; 1
```

```
[person release]; 1
```

6、为什么很多内置类如UITableViewController的delegate属性都是assign而不是retain的?

会引起循环引用

7、定义属性时,什么情况使用copy, assign, 和retain

assign用于简单数据类型,如NSInteger,double,bool,retain 和copy用于对象,copy用于当 a指向一个对象,b也想指向同样的对象的时候,如果用assign,a如果释放,再调用b会crash,如果用copy 的方式,a和b各自有自己的内存,就可以解决这个问题。retain 会使计数器加一,也可以解决assign的问题。另外:atomic和nonatomic用来决 if (property != newValue) { [property release];
property = [newValue retain]; }

8、autorelease的对象是在什么时候被release的?

autorelease实际上只是把对release的调用延迟了,对于每一个Autorelease,系统只是把该Object放入了当前的Autorelease pool中,当该pool被释放时,该pool中的所有Object会被调用Release。对于每一个RunLoop,系统会隐式创建一个Autorelease pool,这样所有的release pool会构成一个象CallStack一样的一个栈式结构,在每一个RunLoop结束时,当前栈顶的Autorelease pool会被销毁,这样这个pool里的每个Object(就是autorelease的对象)会被release。那什么是一个RunLoop呢?一个UI事件,Timer call, delegate call, 都会是一个新的RunLoop。

9、这段代码有什么问题,如何修改

```
for (int i = 0; i < someLargeNumber; i++) {  
    NSString *string = @"Abc";  
    string = [string lowercaseString];  
    string = [string stringByAppendingString:@"xyz"];  
    NSLog(@"%@", string);  
}
```

内存泄露

```
NSAutorelease * pool1 = [[NSAutoreleasePool alloc] init];
```

```
NSString *string = @"Abc";  
string = [string lowercaseString];  
string = [string stringByAppendingString:@"xyz"];
```

11、autorelease和垃圾回收机制(gc)有什么关系？

12、iPhone OS有没有垃圾回收（gc）？

没有

13、什么是Notification？

观察者模式，controller向defaultNotificationCenter添加自己的notification，其他类注册这个notification就可以收到通知，这些类可以在收到通知时做自己的操作（多观察者默认随机顺序发通知给观察者们，而且每个观察者都要等当前的某个观察者的操作做完才能轮到他来操作，可以用NotificationQueue的方式安排观察者的反应顺序，也可以在添加观察者中设定反映时间，取消观察需要在viewDidLoad跟dealloc中都要注销）。参考链接：

<http://useyourloaf.com/blog/2010/6/6/delegation-or-notification.html>

14、什么时候用delegate，什么时候用Notification？

delegate针对one-to-one关系，并且receiver可以返回值给sender，notification可以针对one-to-one/many/none,receiver无法返回值给sender.所以，delegate用于sender希望接受到receiver的某个功能反馈值，notification用于通知多个object某个事件。

15、什么是KVC和KVO？

KVC(Key-Value-Coding)内部的实现：一个对象在调用setValue的时候，（1）首先根据方法名找到运行方法的时候所需要的环境参数。

（2）他会从自己isa指针结合环境参数，找到具体的方法实现的接口。（3）再直接查找得来的具体的方法实现。

KVO（Key-Value-Observing）：当观察者为一个对象的属性进行了注册，被观察对象的isa指针被修改的时候，isa指针就会指向一个中间类，而不是真实的类。所以isa指针其实不需要指向实例对象真实的类。所以我们程序最好不要依赖于isa指针。在调用类的方法的时候，最好要明确对象实例的类名。

16、Notification和KVO有什么不同？

17、KVO在ObjC中是怎么实现的？

18、ViewController 的 loadView, viewDidLoad, viewDidUnload 分别是在什么时候调用的？在自定义ViewController的时候这几个函数里面应该做什么工作？

当view为nil时调用loadView,view完成加载调用viewDidLoad，view释放时调用viewDidUnload。

viewDidLoad在view 从nib文件初始化时调用，loadView在controller的view为nil时调用。此方法在编程实现view时调用,view 控制器默认会注册memory warning notification,当view controller的任何view 没有用的时候，viewDidUnload会被调用，在这里实现将retain 的view release,如果是retain的IBOutlet view 属性则不要在这里release,IBOutlet会负责release。

19、ViewController 的 didReceiveMemoryWarning 是在什么时候被调用的？默认的操作是什么？

内存超过阈值，尝试释放view.

默认调用 [super didReceiveMemoryWarning]

20、UITableViewController中，创建cell时，initWithStyle:reuseIdentifier 中，reuseIdentifier有什么用？简述复用原理。

21、UIView和CALayer有什么区别？

一个UIView包括CALayer树，CALayer是一个数据模型，听说iPhone。包括了一些用于显示的对象，但本身不用于显示。CALayer相当于photoshop的一个层，很多动画可以通过设置CALayer来完成。据说有人用CALayer显示图片来播放视频。Coresome kind ofimation应该用CALayer来完成各种动画。

22、以UIView类 animateWithDuration:animation: 为例，简述UIView动画原理。

1、Object-C有多继承吗？没有的话用什么代替？

没，protocol. 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 来完成的 你不用去思索繁琐的多继承 虚基类的概念 ood的多态特性在 obj-c中通过委托来完成。

2、Object-C有私有方法吗？私有变量呢？

没，有。objective-c -类里面的方法唯有两种 静态方法和实例方法. 这似乎就不是完整的面向对象了按照OO的原则就是一个对象只暴露有用的东西.如果没有了私有方法的话 对付一些小局限的代码重用就不那么利市了. 在类里面声名一个私有方法

```
@interface Controller :NSObject { NSString *something; }
```

```
+(void)thisIsAStaticMethod;
```

```
-(void)thisIsAnInstsome kind ofceMethod;
```

```
@end
```

```
@interface Controller(private) -
```

```
(void)thisIsAPrivateMethod;
```

```
@end
```

```
@private
```

可以用来修饰私有变量在Objective-C中，所有实例变量默认都是私有的，所有实例方法默认都是公有的

3、#import和#include的区别，@class代表什么？

#import 能防止重复引用，#include 不能，@class 前置声明一个类。

4、Object-C的内存管理？

引用计数器。

5、为什么很多内置的类，如**TableViewController**的**delegate**的属性是**assign**而不是**retain**？

防止循环引用。

6、定义属性时，什么情况使用**copy**、**assign**、**retain**？

copy:拷贝对象。**assign**:传值，一般用于基础类型。**retain**:传地址，引用计数器加一。

9、请简述**iPhone**数据持久化的种方法？

答：属性列表、对象归档、**iPhone** 的嵌入式关系数据库(**SQLite3**)和持久性工具 **Core Data**。

IOS开发面试题

- 1、 请手工书写@property (nonatomic, retain) NSString *name; 的get与set方法。

```
getter
- (NSString *)name { return _name; }

setter
-(void)setName:(NSString *)aName
{
if (name != aName)
{
[name release];
name = [aName retain];
}
}
```

- 2、 请描述object c内存管理机制、以及一般原则?

- 1) 当你使用new, alloc和copy方法创建一个对象时, 该对象的保留计数器值为1. 当你不再使用该对象时, 你要负责向该对象发送一条release或autorelease消息. 这样, 该对象将在使用寿命结束时被销毁.
 - 2) 每retain一次会使引用系数就加1, 相应地要在适当位置release
- 内存管理原则

1. 谁创建, 谁释放 (“谁污染, 谁治理”)。如果你通过alloc、new或者(mutable)copy来创建一个对象, 那么你必须调用release或autorelease。或句话说, 不是你创建的, 就不用你去释放
2. 一般来说, 除了alloc、new或copy之外的方法创建的对象都被声明了autorelease (autorelease是延迟释放内存, 不用你自己去手动释放, 系统会知道在什么时候该去释放掉它。)
3. 谁retain, 谁release。只要你调用了retain, 无论这个对象是如何生成的, 你都要调用release

- 3、 nil与NULL的区别?

nil是一个对象, 而NULL是一个值

4、 如何判断一个对象是否属于某个类？

`respondsToSelector ,performSelector,conformstoProtocol,isKindof Class,isMemberofClass`

5、 object c如何实现私有方法？

在object-c中没有私有方法, 只有静态方法和实例方法. 但是可以用 `@private` 修饰私有变量. 用.m文件中使用category来实现私有方法.

6、 IBOutlet UIView *v:

`@property(nonatomic,retain)*V:`

这个v对象需要release吗? 如果需要怎么release?

需要release 在类的dealloc 中release

6、 BOOL与bool区别？

BOOL 只是一个宏，不是基本的数据类型，BOOL实际上=char，占1字节空间。

bool 是C++的基础数据类型，占用1个字节。

对于BOOL类型，当值为1时为真，即YES；当值为0时为假，即NO；

YES和NO都是宏。

对于bool类型，值只要不是0就是为真，即true；当值为0时为假，即false

7、 用NSLog函数输出一个浮点类型，结果四舍五入，并保留一位小数。

`NSLog(@"%.1f",3.14);`

8、 用object-c或者C语言写一个方法判断20120101！（阶乘）在十进制、

二进制下分别为几位数？

```
static double PI = 3.141592653589793239;
static double E = 2.718281828459045;
int mul10(int n){
    return (int)(log10(sqrt(2*PI*n))+n*log10(n/E))+1;
}
```

二进制下

```
int mul2(int n){  
    return (int)(log2(sqrt(2*PI*n))+n*log2(n/E))+1;  
}
```

10、用c语言写一个函数判断两个链表（每个链表不存在环）是否相交？

IOS开发工程师笔试题（请在30分钟内做完）

- 1、Objective-C有私有方法么？私有变量呢？
- 2、关键字const有什么含义？
- 3、关键字volatile有什么含义？并给出三个不同的例子；
- 4、线程与进程的区别和联系？
- 5、简单阐述堆和栈的区别；
- 6、简单阐述一下Objective-C内存管理；
- 7、为什么很多内置类如UITableViewController的delegate属性都是assign而不是retain的？

防止相互引用

仙掌软件C语言试题2011 (V0.01)

1、编写一个函数，要求输入年月日时分秒，输出该年月日时分秒的下一秒。如输入2004

年12月31日23时59分59秒，则输出2005年1月1日0时0分0秒

2、设 $x=2$ ，则表达式 $(++x)*3$ 的值是（ ）。

3、设 $x=2$ ，则表达式 $(++x)/3$ 的值是（ ）。

4、栈操作数据的原则是_____，队列操作数据的原则是_____。

5、用最有效率的办法来计算2乘以8等于几。

6、请写出代码使用位运算来交换整数变量 x ， y 的值。

7、编写代码实现同一平面内两圆是否碰撞，其中：

第一个圆圆心坐标为 (x_1, y_1) ，半径是 r_1 ，第二个圆圆心坐标为 (x_2, y_2) ，如果碰撞返回1，否则返回0

函数声明如下：

```
int colliswith(int x1,int y1,int r1,int x2,int y2,int r2){}
```

8、编写代码实现输入一个整数，求该整数的二进制表达中有多少个1。例如输入10，其二进制表示为1010，有两个1，因此输出2。

9、下面是C语言中两种if语句判断方式，请问哪种写法更好？为什么？

```
int n;
```

```
if (n==10) //第一种判断方式
```

```
if (10==n) //第二种判断方式
```

10、i最后等于多少？

```
int i = 1;
```

```
int j = i++;
```

```
if((i>j++)&&(i++==j) i+=j;
```

11、用预处理指令#define 声明一个常数，用以表明一年中有多少秒（忽略闰月）。

12、说说下面两个声明语句的区别？

1) char const *p

2) char * const p

13、int (*[10]) (int) 表示的是什么？

14、c和c++中的struct有什么不同？

15、有以下表达式：

```
int a=248; b=4;
```

```
int const c=21;
```

```
const int*d=&a;
```

```
int*const e=&b;
```

```
int const*fconst=&a;
```

请问下列表达式哪些会被编译器禁止？为什么？

```
*c=32; d=&b; *d=43; e=34; e=&a; f=0x321f;
```

16、你的程序需要用到一个数据文件，不行的是，由于容量的问题，你不得不事先将其压缩，游戏运行时再解压。

由于该数据文件的特点是，它的每个字节只用到了最后5位（即每个字节的值都小于32），并且连续相同的字节数不超过7个，你决定采取这样的压缩算法：对于连续相同的n个字节，压缩成一个字节，前3位存储n，后5为存储实际数据。

（1）现在已经从文件中读出了len个字节到字节数组file中，写一个函数把它解压到另一个字节数组data。

（2）在什么情况下，这种压缩算法的压缩率较低？

18、简述A*算法的核心思想（或者用伪码实现）

- 1、编写一个标准strep函数。
 - 2、描述vector,list,map各自使用情形。
 - 3、如何将单链表倒转？
 - 4、如何判断字符串是否为回文？比如“abcba”就是一个回文。
 - 5、找出一个整数数组中第二大的数，尝试处理各种输入异常。
 - 6、写段代码，去掉一个有英文字母组成的字符串中的重复字符。
 - 7、写出快速排序或冒泡排序的代码。
-

品科艺IOS开发面试题

- 1、 在objective-c中, 说明一下你如何让一个函数或类变量成为一个私有的函数或类变量。

@private 可以修饰 成员变量 成为私有的

成员方法可以在类的.m文件实现在.h不能声明就可以实现类似私有的方法

- 2、 描述一个UIView框架 (UIView frame) 和边界 (bounds) 之间的差异。

如果一个固定大小的UIView被做成动画来回移动, 这个框架或边界会不会变化?

frame: 该view在父view坐标系统中的位置和大小, 参照点是父视图的坐标系统。

bounds: 该view在本身坐标系统中的位置和大小, 参照点是本身坐标系统。

Frame会变 bounds不会变

- 3、 描述什么会在UIViewController中触发 (void)

didReceiveMemoryWarning函数, 你怎么在View controller中重写 (override) 这个函数。

当发生内存警告时会调用

iOS6之前需要调用didReceiveMemoryWarning 内部写self.view = nil;

并且再调用viewDidUnload 并且在里面释放子视图

iOS6之后在didReceiveMemoryWarning中不需要调用

viewDidUnload，需要手动写self.view = nil 并且释放子视图

- 4、 若有一个表有1000行，每一行包含一个不同的图像，每一个图像都是从不同的网址URL下载的（例如http://www.test.com/test123.jpg）,您将采取什么办法，以确保表内图像滚动尽可能顺利？

1. 使用第三方开源库SDWebimage

2. 异步下载，保存每一个图片的cell所在行，另外在本地以图片网

络地址加密进行本地保存，下次加载从本地获取

- 5、描述在声明一个@property时关键字之间的差异

weak

strong

copy

assing

retain

- 6、写一个函数，当你有一个字符串数组，例如：

Input NSArray content (@ "1apple" , @ "2banana" , @ "3oranges")

要输出一个NSDictionary object，其中每一个字符串的第一个字母是key，剩下的是value/object，例如：

NSDictionary content ("1" => "apple" , "2" => "banana" , "3"
=> "orange")

```
void arrayChangeToDictionary(NSArray *arr) {  
    NSMutableDictionary *dict = [[NSMutableDictionary alloc]  
init];  
    for (NSString *str in arr) {
```

```

        [dict setObject:[str substringFromIndex:1]
 forKey:[str substringToIndex:1]];
    }
    NSLog(@"dict:%@",dict);
}

```

7、当具有a class @interface RadioTuner: NSObject时，写出.h文件和.m文件，以证明使用Singleton设计模式。

```

@interface Singleton : NSObject
+ (Singleton *)sharedSingleton;
@end
#import "Singleton.h"

@implementation Singleton
//定义静态全局变量
static Singleton * single = nil;

+ (Singleton *)sharedSingleton{
    //考虑线程安全
    @synchronized(self){
        if (single == nil) {
            single = [[self alloc] init];
        }
    }
    return single;
}

```

9.写一个objective-c的函数来找到第N个素数。

```

- (BOOL)isPrimeNumber:(NSInteger)number {
    BOOL isPrime = YES;
    for (int i = 2; i <= number/2; i++) {
        if (number % i == 0) {
            isPrime = NO;
            break;
        }
    }
    return isPrime;
}

- (NSInteger)prime:(NSInteger)n {

```

```
NSInteger count = 0;
NSInteger i = 2;
while(1) {
    if ([self isPrimeNumber:i]) {
        count++;
    }
    if (count == n) {
        return i;
    }
    i++;
}
}
```

10、写一个objective-C的函数来测试一个字符串是否是回文（回文是一个字符串，在阅读是前进和后退是相同的，例如：racecar, level）。

```
- (BOOL)isLoopNumber:(NSString *)str {

    NSString *newStr = @"";

    for (NSInteger i = str.length-1; i >= 0; i--) {
        unichar c = [str characterAtIndex:i];
        newStr = [newStr stringByAppendingFormat:@"%C",c];
    }
    if ([newStr isEqualToString:str]) {
        return YES;
    }else{
        return NO;
    }
}
```

网络题

1、画出TCP/IP四层模型或IOS的七层模型。

2、IP地址头部都有哪些字段？简单描述其作用。

3、IPv6与IPv4的区别？

4、简述TCP协议链接的建立过程，断开链接的过程，以及端口的作用。

5、Qos用来干什么的？

6、请尽可能的写出你对WLAN的了解。

believe oneself

软通动力IOS开发面试题

- 一、用预处理指令#define定义求两个数中的较大值。
- 二、Objective-C有私有方法么？私有变量呢？
- 三、阐述static关键字的作用。
- 四、Objective-C有多重继承么？不是的话有什么替代方法？
- 五、#import跟#include的区别，@class呢？
- 六、简单阐述一下Objective-C内存管理。
- 七、定义属性时，什么情况使用copy，passing，和retain？
- 八、你是怎么理解MVC的
- 九、Objective-C中得protocol、selector、category是什么？

协议 选择器 类别(类目)

代码题：

- 一、这段代码的意义，有没有问题

```
- (void) setName: (NSString*) name {  
    [myname release];  
  
    [name retain];  
  
    myname=name;  
}
```

这段代码最好加一个判断

```
- (void) setName: (NSString*) name {  
    if(myname!=name){  
        [myname release];  
        [name retain];  
        myname=name;  
    }  
}
```

二、self.name= “object” ;和name= “object” 有何不同?

三、以下代码有没有问题? 有没有内存泄露?

```
for (inti=0;i<someLargeNumber;i++)  
{  
    NSString*string=@ “Abc” ;  
    string=[string lowercaseString];  
    string=[string stringByAppendingString:@ “xyz” ];  
    NSLog(@ “%@” ,string);  
}
```

string 一直重复打印abcxyz 如果someLargeNumber 特别大会存在内存方面的问题, 因为for循环中一直不断创建新的对象, 最好在for里面加一个自动释放池, 释放里面不用的对象

如:

```
for (inti=0;i<someLargeNumber;i++)
```

```
{  
  
    @autoreleasepool {  
  
        NSString*string=@ "Abc" ;  
  
        string=[string lowercaseString];  
  
        string=[string stringByAppendingString:@ "xyz" ];  
  
        NSLog(@ "%@" ,string);  
  
    }  
  
}
```

四、写一个iOS中委托的定义以及使用

实际上 委托就是代理，比如A委托B做一些事情，那么B就是A的代理，iOS中代理设计模式的应用常常会 and 协议结合起来使用，常常用协议来规范约束代理的行为

A.h

```
#import <Foundation/Foundation.h>  
@protocol Work <NSObject>  
- (void)work;  
@end  
@interface A : NSObject  
{  
    id <Work> _delegate;  
}  
@property (nonatomic,assign) id <Work>delegate;  
- (void)delegateGoToWork;  
@end
```

A.m

```
#import "A.h"

@implementation A
- (void)delegateGoToWork {
    if ([self.delegate respondsToSelector:@selector(work)])
    {
        [self.delegate performSelector:@selector(work)];
    }
}
@end

B.h

#import <Foundation/Foundation.h>
#import "A.h"
@interface B : NSObject <Work>

@end
B.m

#import "B.h"

@implementation B
- (void)work {
    NSLog(@"B is Working");
}
@end

#import "A.h"
#import "B.h"
int main(int argc, const char * argv[])
{
    @autoreleasepool {
        A *a = [[A alloc] init];
        B *b = [[B alloc] init];

        a.delegate = b;
        [a delegateGoToWork];
        [a release];
        [b release];
    }
    return 0;
}
```

上机题

- 1、写一个iPhone程序，有navigation 导航栏，内容是一个table，table的第一行放1行文字（文字内容来源于项目里新建的plist文件），第二行放一个文本框；点击每一行会弹出不同的提示信息。
 - 2、写一个iPhone程序，有2屏，可以通过滑动切换，第二屏有一个webview，读取本地的html文件。Html文件中会加载一个本地的xml文件，获取xml文件中的数据内容并显示。（可选：html中加载本地的js文件）
-

产研中心技术部招聘面试题

- 1、描述UITableView的重用机制。
- 2、ViewController 从创建到显示在屏幕上分别调用了哪些函数？
- 3、NSArray和NSMutableArray有什么区别？
- 4、编程题：定义一个委托，并调用、实现它。
- 5、编程题：写一个异常处理代码块。
- 6、编程题：写一个block代码块。
- 7、编程题：实现一个view使其可以在view controller 里使用，里面由上到下放标签，字体颜色蓝色；一个输入框；一个圆角按钮，按钮文字是“发送”；
- 8、Can we use one tableview with two different datasources? (可用中文回答)
- 9、Difference between frame and bounds ?(可用中文回答)
- 10、实例化一个UITableView对象 要求写出关键语句