

## OC 部分

1. Objective-C可以实现多继承吗？可以实现多个接口吗？Category是什么？重写一个类的方式是用继承好还是用分类好？为什么？

2. OC使用协议实现多继承，可以遵守多个协议实现多接口。category是OC中的类别，类别是用于给一个现有类添加新方法。重写类一般采用继承的方式，分类（类别）在给以个类添加同名方法后，会造成原有类中方法的实效，而继承重写的方法，依然可以在使用父类对象调用该方法。

3. 定义属性的时候，什么时候使用retain/copy/assign？写一个setter方法，用于完成@property(nonatomic,retain) NSString\* name；写一个setter方法，用于完成@property(nonatomic,copy) NSString\* name；

retain 用于保留对象的引用计数，在使用retain声明的属性做赋值的时候，成员变量指针会保留被赋值对象的引用计数。

copy 声明的属性，在使用属性赋值的时候会，成员指针会指向新副本，这个副本是一个不可变副本，不论赋值对象是不是可变的。

assign 用于基本数据类型的属性声明，不涉及到内存管理的问题，也是缺省参数

retain声明的属性的setter方法展开

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

copy声明的属性的setter方法展开

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

4. 什么时候使用NSMutableArray/什么时候使用NSArray

---

NSMutableArray 一般在需要随时更改数组结构的时候使用  
NSArray 一般用于保存一些不需要修改逻辑的数据

## 5. 实现字符串 “I LOVE CHINA” 反串成 “CHINA LOVE I”

## 6. pch 文件 的作用

.pch 表示“precompiled header”，这是一个你工程要用到的来自于外部框架的头文件列表。xcode 将编译这些头到文件，这将减少你在选择 Build 或 Build and Go 时编译项目的时间。通常用到的头文件已经自动包含了 pch，系统编译每个 cpp 文件前，都会先 include 这个文件。这样就节省了添加 include 的时间，相当于加速编译  
还有就是可以再这里面放入宏，在整个工程中都可以用

## 7. 怎样解决重复编译

```
#ifndef _DEBUG  
#ifdef USE_MYLIB  
.....  
#endif
```

## 8. awakeFromNib 与 viewDidLoad 区别

awakeFromNib

当.nib 文件被加载的时候，会发送一个 awakeFromNib 的消息到.nib 文件中的每个对象，每个对象都可以定义自己的 awakeFromNib 函数来响应这个消息，执行一些必要的操作。也就是说通过 nib 文件创建 view 对象是执行 awakeFromNib 。

viewDidLoad

当 view 对象被加载到内存是就会执行 viewDidLoad，所以不管通过 nib 文件还是代码的方式创建对象都会执行 viewDidLoad。

## 9. LayoutSubviews 何时会被调用

当要调整 subViews 时候，需要重写 layoutSubviews 方法。

1:初始化 init 方法时候不会触发。

2:滚动 UIScrollView 时会触发

3:旋转 UIScreen 时会触发

4:当改变 view 的值时候会触发，前提是 frame 前后值发生了变化

5:当改变 UIView 的大小时候会触发

## 10. public/private/protected 的具体区别

public 公共，加上这个修饰的类或属性，可以在同一个包或者别的包里面访问  
private 私有的，加上这个修饰的类或属性，只能在同类里访问，同包和别的包不能访问

---

protected 保护，加上这个修饰的类或属性，只能在类和同包访问，别的包不能访问

#### 11. ARC 是什么

ARC 是 iOS 5 推出的新功能，全称叫 ARC(Automatic Reference Counting)。简单地说，就是代码中自动加入了 retain/release，原先需要手动添加的用来处理内存管理的引用计数的代码可以自动地由编译器完成了。

该机能 在 iOS 5/ Mac OS X 10.7 开始导入，利用 Xcode4.2 可以使用该机能。简单地理解 ARC，就是通过指定的语法，让编译器(LLVM 3.0)在编译代码时，自动生成实例的引用计数管理部分代码。有一点，ARC 并不是 GC，它只是一种代码静态分析(Static Analyzer)工具。

#### 12. 写一个“标准”宏，这个宏输入两个参数并返回较小的

```
#define MIN(X,Y) ((X)>(Y)?(Y):(X))
```

#### 13. Objective-c 中有多重继承么？不是的话有声明替代方式？

没有多重继承，可以通过协议模拟多继承

#### 14. Objective-c 中有私有方法吗？私有变量呢？

没有私有方法，但可以将方法直接实现在.m 文件中不在.h 文件中声明时，外部也不能访问。

有私有变量

#### 15. iPhone OS 中有没有垃圾回收？

没有

#### 16. 常见的 object-c 的数据类型有哪些，和 c 的基本类型有什么区别

答：常见的 object-c 的数据类型有 NSInteger、CGFloat、NSString、NSNumber、NSArray、NSData，NSInteger 会根据系统是 32 位还是 64 位来决定是本身是 int 还是 Long，

CGFloat 会根据系统是 32 位还是 64 位来决定是本身是 float 还是 double，NSString、NSNumber、NSArray、NSData 都是指针类型的对象，在堆中分配空间，而 c 语言中的 char，[]等都是在栈中分配空间

#### 17. id 声明的对象有什么特性？

id 声明的对象具有运行时的特性，即可以指向任意类型的 objective-c 的对象；

#### 18. 想 nil 对象发送消息会发生什么？

答：在 Objective-C 中向 nil 发送消息是完全有效的，只是在运行时不会有任何作用。

#### 19. 什么是 block？block 实现原理？

答：block 是一个特殊的 OC 对象，它建立在栈上，而不是堆上，这么做一个是为性能考虑，还有就是方便访问局部变量。默认情况下 block 使用到的局部变量都会被复制，而不是保留。所以它无法改变局部变量的值。如果在变量面前加上

---

\_\_block, 那么编译器回去不会复制变量, 而是去找变量的地址, 通过地址来访问变量, 实际上就是直接操作变量。另外 block 是在栈上分配的, 所以一旦离开作用域, 就会释放, 因此如果你要把快用在别的地方, 必须要复制一份。block 是不能保留的, retain 对块没有意义。

## 20. C++和 OC, JAVA 和 OC 之间的区别?

C++是功能强大, 丰富的面向对象编程语言, 具有私有、公有、保护权限的三种成员变量和成员方法, 具有私有、公有、保护三种继承方式, 具有重写, 重载, 虚函数, 虚基类等多态方式, 通过虚基类实现代理回调。自定义类可以没有父类。另外具备向量, 模板, 友元, 重载运算符等多种独特语法

Obj-C 是针对 mac OS 和 iOS 设备应用程序开发的专属编程语言, 采用动态继承, 消息方法机制, 没有真正的重写机制, 没有私有方法, 继承方式为公有, 具备协议, 类别, Block 等独有的语法, 万用父类为 NSObject

JAVA 是老牌的面向对象语言, 编写的程序在 JAVA 虚拟机上运行, 真正实现了一次编译到处运行, 具有复杂的内存回收机制, 单继承模式, 接口语法类似 Obj-C 的协议

## 21. 抽象与接口的区别?

声明方法的存在而不去实现它的类被叫做抽象类 (abstract class), 它用于要创建一个体现某些基本行为的类, 并为该类声明方法, 但不能在该类中实现该类的情况。不能创建 abstract 类的实例。然而可以创建一个变量, 其类型是一个抽象类, 并让它指向具体子类的一个实例。不能有抽象构造函数或抽象静态方法。Abstract 类的子类为它们父类中的所有抽象方法提供实现, 否则它们也是抽象类为。取而代之, 在子类中实现该方法。知道其行为的其它类可以在类中实现这些方法。

接口 (interface) 是抽象类的变体。在接口中, 所有方法都是抽象的。多继承性可通过实现这样的接口而获得。接口中的所有方法都是抽象的, 没有一个有程序体。接口只可以定义 static final 成员变量。接口的实现与子类相似, 除了该实现类不能从接口定义中继承行为。当类实现特殊接口时, 它定义 (即将程序体给予) 所有这种接口的方法。然后, 它可以在实现了该接口的类的任何对象上调用接口的方法。由于有抽象类, 它允许使用接口名作为引用变量的类型。通常的动态联编将生效。引用可以转换到接口类型或从接口类型转换, instanceof 运算符可以用来决定某对象的类是否实现了接口。

## 22. nil 与 NULL 的区别?

从 Objective-C 语言的官方说法上看, nil 表示指向对象的指针即所谓对象的引用为空, NULL 表示指向基础类型变量即 C 语言变量的指针为空。如果在非 ARC 程序的编写过程中, 两个空是可以互换的, 但是在 ARC 环境下, 普通指针和对象引用被严格限制, 不能交换使用, 因此也应尽量不互换使用 nil 与 NULL

## 23. BOOL 与 bool 的区别?

bool 是 C 语言 C99 标准中增添的变量类型, Object-C 仅仅是从 C 语言继承了这种类型, 该类型有 true 和 false 两个值, 表示真和假。BOOL 是 Obj-C 独有的

---

布尔类型，有 YES 和 NO 两个值，分别是 1 和 0 的宏。Obj-C 中同时认为所有非 0 的值都是真值，0 为假值

#### 24. OC 如何实现私有方法？

Obj-C 语法中并没有私有方法的概念，但是由于 Obj-C 是通过导入其他类的头文件来获取其他类所拥有的成员方法的声明，因此可以采用编写方法时，不声明，或仅在.m 文件中的匿名类别中声明的方式，使方法对外不可见，即可达到方法私有化的目的。但是外部类仍然可以通过@selector 来访问确实存在的私有方法，因此严格来讲 Obj-C 确实不能真正实现方法私有化

#### 25. #import 和#include 的区别 @ class 代表什么 预编译指令

Objective-C: #import

C, C++: #include

#import 由 gcc 编译器支持

在 Objective-C 中，#import 被当成 #include 指令的改良版本来使用。除此之外，#import 确定一个文件只能被导入一次，这使你在递归包含中不会出现

问题。

使用哪一个还是由你来决定。一般来说，在导入 Objective-C 头文件的时候使用 #import，包含 C 头文件时使用 #include。比如：

```
#import
```

```
#include
```

```
#include
```

#import 比起#include 的好处就是不会引起交叉编译

#### 二、@class 是用来做类引用的

@class 就是告诉编译器有这么一个类，至于类的定义是啥不知道

@class 一般用于头文件中需要声明该类的某个实例变量的时候用到，在 m 文件中还是需要使用#import

#### 26. NSString 和 NSMutableString 的区别

NSString 是一个不可变的字符串对象。这不是表示这个对象声明的变量的值不可变，而是表示它初始化以后，你不能改变该变量所分配的内存中的值，但你可以重新分配该变量所处的内存空间。而 NSMutableString 是可变的，意味着你可以追加它的内存空间，或者修改它所分配的内存空间中的值。

---

27. . 关于语句 `NSString * str= [NSData alloc]init`, 编译和运行分别 `str` 代表什么对象?

首先, 声明 `NSString *str` 是告诉编译器, `str` 是一个指向某个 Objective-C 对象的指针。因为不管指向的是什么类型的对象, 一个指针所占的内存空间都是固定的, 所以这里声明成任何类型的对象, 最终生成的可执行代码都是没有区别的。这里限定了 `NSString` 只不过是告诉编译器, 请把 `str` 当做一个 `NSString` 来检查, 如果后面调用了非 `NSString` 的方法, 会产生警告。

接着, 你创建了一个 `NSData` 对象, 然后把这个对象所在的内存地址保存在 `str` 里。那么运行时, `str` 指向的内存空间就是一个 `NSData` 对象。你可以把 `str` 当做一个 `NSData` 对象来用。

28. socket 通信的几个关键步骤

面向连接的 socket 通信就像与对方打电话, 首先需要通过电话建立一个连接, 连接建立好之后, 彼此才能双向通信。它有几个关键步骤

服务器端通常以守护进程的方式实现:

- 1: 创建守护进程
- 2: 获取或注册服务
- 3: 创建 socket 并绑定地址
- 4: 开始监听
- 5: 接收客户端连接请求
- 6: 进行数据传输

客户端

- 1: 获取或注册服务
- 2: 创建 socket
- 3: 发送连接请求

29. 类别意义? 与继承的区别

当我们添加头文件以后, 对已知的类, 会自动提示你对这个类添加的方法  
主要用途, 对于原生不会造成破坏, 使用原生就可以提示出你的方法

30. Core Foundation 中提供了哪几种操作 Socket 的方法?

CFNetwork、CFSocket 和 BSD Socket

31. 用 `id` 声明的对象有什么特性?

Ø 没有 \* 号

Ø 动态数据类型

Ø 可以指向任何类的对象(设置是 `nil`), 而不关心其具体类型

Ø 在运行时检查其具体类型

Ø 可以对其发送任何(存在的)消息



---

32. self.name= “object” name= “object” 有什么区别？

前者实际上是调用了set方法给变量赋值 而后者是直接给变量赋值

33. shell中，将command1的输出作为command2的输出应该使用的命令是？

重定向命令>

command1>command2

34. 下面的数据结构中不属于线性结构的是：栈，链表，二叉树，线性表

线性结构：栈，链表，线性表

非线性结构：二叉树

35. oc中有没有多继承，如果没有用什么方法替代？

没有 用协议代替多继承

36. 常见的 Objective-C 的数据类型有哪些，和 C 的基本数据类型有什么区别

O C 中常用数据类型有

NSArray, NSDictionary, NSData, NSString, NSMutableString 等等，和 C 的最大区别为 OC 中的类型是类类型，需要实例化对象才能用。C 中是一般数据类型直接操作内存空间

37. self.name=@ “aa” 和 \_name=@ “aa” 的区别

答：self.name=@ “aa” 是通过 set 方法进行赋值，\_name=@ “aa” 是直接复制给成员变量

38. C 语言中指针与数组的区别

答：指针是变量可以修改指向的方向，数组名是地址常量，不能被修改

39. new delete malloc free 的含义

c++:new 申请内存， delete 释放掉指针指向的内存

c:malloc 动态申请内存，free 释放指针指向的内存

40. 常引用什么时候使用

如果既要利用引用提高程序的效率，又要保护传递给函数的数据不在函数中被改变，就要使用常引用

41. c/oc/c++ 有什么区别和联系

C 相对于 C++ 和 OC 而言更偏重于逻辑算法，这是因为 C 是面向过程，C++ 和 OC 都是面向对象。C 和 C++ 的联系：C 是 C++ 的一个子集，C++ 是 C 的超集，OC 是 C 的扩展，C++ 和 OC 基本兼容 C 的语法。

42. const 的用法

const 修饰变量表示该变量是只读变量（有些人管它叫常量），即只能引用而不能修改

---

`const int *p;`表示指针变量 P 指向的数据不能改  
`int *const p;`表示指针变量 P 的值不能改，或者说是指针 P 的指向不能改

43. [pool release]和[pool drain]有什么区别

drain 和 release 都会促使自动释放池对象向池内的每一个对象发送release消息来释放池内对象的引用计数，但是release触发的这个操作，不会考虑对象是否需要release，而drain会在自动释放池向池内对象发送release消息的时候，考虑对象是否需要release

44. 自动释放池和 GC 一样吗，iphone 有没有 GC

在引用计数环境下，  
ios 是没有垃圾回收的，自动释放池是 oc 中管理内存的一种方式，它和 gc 是本质区别的，自动释放池管理内存的前提是，必须把要管理内存的对象加入池内，才会生效。而 gc 是不断检测当前程序中是否有不再使用的内存进而释放。  
45. 当A类 中的某个方法执行到某处时，这时想在B类中执行某个方法，如何做？并做简单说明

用代理执行代理方法

说明：在b类中实现协议方法，设置a的代理为b，在指定方法内调用代理的协议方法

46. 类别的作用？

答案：category 可以在不获悉，不改变原来代码的情况下往里面添加新的方法，只能添加，不能删除修改。

并且如果类别和原来类中的方法产生名称冲突，则类别将覆盖原来的方法，因为类别具有更高的优先级。

类别主要有3个作用：

- (1) 将类的实现分散到多个不同文件或多个不同框架中。
- (2) 创建对私有方法的前向引用。
- (3) 向对象添加非正式协议。

47. 简述extern C的作用

可以在 C++中使用 C 的已编译好的函数模块, 在 c++中么用到 c 语言写的函数, 声明一下, 在 DLL 中经常看到, 避免 C++ name mangling, 主要用于动态链接库, 使得导出函数名称与 C 语言规则一致（不改变），方便不同的编译器甚至是不同的开发语言调用。

extern "C"是告诉C++编译器以C Linkage方式编译，也就是抑制C++的name mangling机制。

**编程：**

1. 写出@property(nonatomic, retain)Person \*person;@synthesize person 具体实现。

```
-(void)setPerson:(Person *)person
{
```



---

```

        if(_person != person) {
            [_person release];
            [_person = person retain];
        }
    }
}

```

```

- (Person *)person
{
    return _person;
}

```

2. 从普通 id 类型对象转换成数字对象，因为配置了限定词.1f，所以结果四舍五入，并保留一位小数

```

NSDictionary* rowData =
[NSDictionary dictionaryWithObjectsAndKeys:@"46.95",@"price",nil];
NSLog(@"a double value:%.1f",[(NSNumber*)[rowData
objectForKey:@"price"] doubleValue]);

```

输出：

a double value:47.0

3, 写一个委托的interface

```

#import
@protocol MyDelegate;//声明
@interface MyClass:NSObject
{
    id    delegate;
}
@end
@protocol MyDelegate//委托方法
-(void) selector:(NSString *) args;
@end

```

4:请看下面一段代码

```

static int a=1;
int main() {
    int b=2;
    char* c=NULL;
    c=(char*)malloc(100*sizeof(char));
    return 0;
}

```

问：1，访问abc三者的效率从高到低依次是：

bca

- 
- 2, 在最坏情况下排序速度最快的是: 归并排序  
复杂度最高的是: 快排  
a, 快排, 冒泡, 堆, 归并

5. 看下面的程序

```
=====
NSMutableArray *arr = [[NSMutableArray array] retain];
NSString *str = [NSString stringWithFormat:@"%test"];

[str retain];
[arr addObject:str];
NSLog(@"%%lu", str, [str retainCount]);
[str retain];
[str release];
[str release];
NSLog(@"%%lu", str, [str retainCount]);
[arr removeObject:str];
NSLog(@"%%lu", str, [str retainCount]);
=====
```

三次打印的retainCount分别是什么, 为什么?

答:

分别是3, 2, 1,

初始化的时候的为1, retain的时候+1, 往数组里add的时候+1, release的时候-1, 从数组里移除的时候-1