

1. 下面是深拷贝还是浅拷贝 当str变化的时候strCopy是否发生改变

```
NSString *str = @"hello word!";
NSString *strCopy = [str copy]
```

2. 叙述一下哈希原理和哈希存储过程

3. 下面两段代码输出是什么

```
@implementation Son : Father
- (id)init {
    if (self = [super init]) {
        NSLog(@"%@", NSStringFromClass([super class]));
    }
    return self;
}
@end

+ (void)viewDidLoad {
    [super viewDidLoad];
    NSLog(@"1");
    dispatch_sync(dispatch_get_main_queue(), ^{
        NSLog(@"2");
    });
    NSLog(@"3");
}
```

4. 简述怎样实现一个精确的NSTimer.

5. NSInteger和Int的区别是什么. NSInteger是系统对long 的别名, 并根据系统进行中位数的进行了区分, Int 默认是32位

6. 编译过程做了哪些事情 使用BitCode的意义是什么.

7. 使用过那些类库,是否看过源码,简述其实现过程.

8. 对于新技术什么态度.

9. 一万个整形数去重和排序. 一万个整数去重和排序;去重, 把数据放到set中, 去重,对去重的通过快排进行排序. (时间复杂度: $O(n) + \log(n)$)

10.一亿个整形数去重和排序. 如果使用上述方法,可能会造成内存溢出. 可以使用1&0来表示数字是否存在, 定义一个大数组,如果对应的下标为数字,比如如果数组中存在12,那么在数组中第11位置标记1,最后会得到一个大数组,存储的[1,0,1,0,0,1,1,.....];这样的结构,最后for寻边遍历一下数组就可以得到去重的,并排序好的数组

11.在 Queue 中, 获得 Label.text.

```
let queue = DispatchQueue(label: "abc")
queue.async {
    let m: String = method()
    var n = "" // 获得 Label.text
    print(m+n)
}
```

12. 实现一个间隔 20ms 的主动轮询机制. 实现一个20s的轮询: 开辟一个子线程,并加入一个timer,定时20执行,让子线程加入到事件循环中(runloop),保活线程.然后在定时器里加入事件,进行处理任务.

13. 生产速率不定, 消费速率恒定, 如何设计数据的存储和访问.