

360

一面:

1.String 内容存放在哪儿 [运行时常量池]

2.然后讲了一下 jvm 的内存结构,堆、栈、程序计数器、方法区、网赚博客等等,每块区域的作用之类的

3.jvm 的回收机制是什么 [感觉问得很宽泛,只好从分代回收算法说起了,从 GC ROOTS 的引用链机制到新生代的小空间复制算法再到老生代的标记清除算法,最后就是新生代和老生代之间的关系以及跃迁等]

4.再讲讲 jvm 的类加载机制吧

5.反射机制是如何,利用类加载机制的呢?

6.懂哪些设计模式 [单例,建造者,观察者,模板方法模式,迭代器模式]

7.你在项目中用到过观察者模式吗?怎么用的? [做统一配置组件的时候使用到了 zookeeper 的发布订阅,创建了监听链表,完成应用系统的注册和同步,然后边写代码边讲细节。]

8.你写个单例模式吧,把你认为效率最高的单例实现写出来。 [这当然是通过 jvm 的内部类机制实现单例了,又快又安全]

9.然后说再写一个双检查的懒汉单例模式吧

10.为什么要用 volatile 关键字 [在高并发时会出现并发模式异常,volatile 可以防止指令重排,创建对象操作并不是一个原子操作,分为三个步骤]

11.能不能讲讲线程的内存模型

12.谈谈你对 cas 的理解。

二面:

1.场景模拟:假设有一个场景,系统需要某个特定时间内响应用户请求,比如说 100ms 内完成用户请求,但是 www.818mulu.cn 在最高峰的时候每单位时间几百万的用户请求,也就是高并发,但我必须要实现系统响应及时,而且高可用,假如你是架构师,你该如何架构这个系统,聊聊你的方案。

2.然后向我介绍了一下他们部门是做什么业务的

3.为什么离职等

4.聊了聊薪资待遇

陌陌

一面:

1.先写代码,写一个二叉树的深度遍历 [递归搞定]

2.memcache 的内存管理机制 [slab 分块管理,最终以 chunk 分配,维护了一个 LRU 的双向列表等等]

3.问了问 zookeeper 的一些知识

二面问的比较杂,大部分不记得了,只记得问了 vi 命令。