

## 7 日复习计划

以下是针对你的简历和Java开发岗位校招的 **7天高效准备计划**，每天3小时左右，覆盖核心考察点，强化优势并弥补潜在短板：

### Day 1-2: Java基础与框架深度复习（每天3小时）

目标：巩固Java核心知识点和框架底层原理。

#### 1. Java基础（1.5小时）

- 重点复习：集合框架（HashMap、ConcurrentHashMap源码）、多线程（线程池参数、锁机制、volatile/CAS）、JVM（内存模型、GC算法、类加载机制）。
- 结合项目：思考你项目中如何应用多线程（如Kafka异步处理）和JVM调优（如缓存优化）。

#### 2. 框架与中间件（1.5小时）

- Spring Boot：自动配置原理、AOP实现（参考你的敏感词过滤项目）、事务管理。
- Redis：数据类型、缓存穿透/雪崩解决方案（结合你的缓存优化经历）。
- Kafka：消息可靠性（如ACK机制）、如何保证顺序性？

### Day 3-4: 算法与数据结构（每天3小时）

目标：熟练解决高频算法题，掌握优化思路。

#### 1. 刷题策略：

- 每日刷15题，重点覆盖：数组/字符串、链表、二叉树、动态规划（如背包问题）、排序（快排/归并）。
- 推荐平台：LeetCode热题HOT 100（优先前50题）。

#### 2. 手撕代码技巧：

- 白板编码时注意边界条件（如分页优化中的PageHelper逻辑）。
- 结合项目：思考如何用算法优化高并发场景（如Redis缓存淘汰策略LRU实现）。

### Day 5: 系统设计与项目复盘（3小时）

目标：深入梳理项目细节，应对系统设计问题。

#### 1. 项目复盘（2小时）：

- 高并发论坛项目：
  - 如何设计缓存分层（Redis + Guava）？
  - 为什么选择Kafka异步处理？若消息堆积如何解决？
  - Elasticsearch倒排索引原理？如何优化搜索性能？
- 园区数字系统：
  - 策略模式在租金计算中的应用（画UML图说明）。

- 分页查询优化（PageHelper原理？如何防止深分页？）。

## 2. 系统设计（1小时）：

- 准备经典题：短链系统、秒杀系统、分布式ID生成。
- 结合你的经验：尝试设计一个高并发的点赞系统（参考你项目中的消息通知逻辑）。

---

## Day 6：模拟面试与行为问题（3小时）

目标：模拟真实面试场景，提升表达逻辑。

### 1. 技术模拟（2小时）：

- 找同学或在线平台模拟面试，重点考察：
  - 项目难点（如“如何将QPS提升到3000+”）。
  - 框架原理（如Spring Security的RBAC实现细节）。

### 2. 行为问题（1小时）：

- 准备高频问题：
  - “遇到最难的技术挑战是什么？如何解决？”（可答缓存雪崩问题）。
  - “为什么从能源专业转计算机？”（突出自学能力和兴趣）。

---

## Day 7：查漏补缺与简历优化（3小时）

目标：完善简历细节，调整心态。

### 1. 简历优化：

- 量化成果：例如“接口响应时间降低50%”改为“从200ms优化至100ms”。
- 补充技术关键词：如“分布式事务（Seata）”“设计模式（策略模式）”。

### 2. 查漏补缺：

- 复习薄弱点（如计算机网络：TCP三次握手、HTTP/HTTPS区别）。
- 准备1分钟自我介绍，突出技术亮点（如高并发优化经验）。

---

## 每日附加任务

- 早间30分钟：背诵3个面试高频题（如HashMap vs ConcurrentHashMap）。
- 晚间30分钟：复盘当日学习内容，记录未掌握知识点。

---

### 注意事项：

1. 面试中遇到不会的问题时，先拆解问题，结合项目经验尝试推导答案。
2. 保持自信，重点展示你的“高并发优化”和“系统设计”经验，这是简历最大亮点！

祝你在校招中斩获心仪Offer！🚀