

# 面试知识大汇总

面试考察的知识点多而而杂，要完全掌握需要花费大大量量的时间和精力力。但是面面试中经常被问到的知识点却没有多少，你完全可以用用 20% 的时间去掌握 80% 常问的知识点。在这里里里我将这 80% 常问的知识点整理理出来，方方便便 大大家快速地掌握。这些知识点也标注了了重要程度，从而而让大大家可以根据知 识点的重要程度去制定学习计划。

如果你想更更好的学习这些知识点，可以参考 [知识总结方方法](#) 一一文文，这篇文文 章的最后给出了了比比本文文更更详细的思维导图。

下图列列出了了面面试考察的九大大知识点，也标出了了重要程度和复习难度。

常见数据结构与算法 刷 100 题 Leetcode 或者刷剑指 Offer 重要程度：★★★ 复习难度：★★★	操作系统、Linux 重要程度：★★★ 复习难度：★★☆	计算机网络、HTTP、Socket 重要程度：★★★ 复习难度：★★☆
分布式与系统设计 重要程度：★★★ 复习难度：★★☆	语言基础 重要程度：★★☆ 复习难度：★★★	设计模式、面向对象 重要程度：★★☆ 复习难度：★★☆
数据库 重要程度：★★☆ 复习难度：★★☆	中间件和框架 重要程度：★★☆ 复习难度：★★☆	项目 重要程度：★★☆ 复习难度：★★★

其实重要程度根据不同的公司会有所不同，针对上图做一一下几点说明：

分布式与系统设计部分对于大大厂厂面面试来说至至关重要，但是小小厂厂面面试考察的会少一点。

语言基础部分，阿里里里喜欢问 Java、腾讯喜欢问 C++，其它大大厂厂面面试考察会少一些，小小厂厂面面试会考察多一些。数据库、中间件和框架部分对于大大厂厂面面试来说不是那么重要，但是对

●●●

于小小厂厂面面试会重要一些。

最后再说一一下项目目部分，它在实习招聘阶段中不是特别重要，但是在校园招聘阶段重要程度会增高。但是如果你的学历不好，而而且没有博客、开源项目目、竞赛等加分项，那么最好做个比较出色色的项目目来增加简历通过率。

## 1. 算法

---

### GitBook 内容大纲

1. 冒泡排序
2. 选择排序
3. 插入排序
4. 希尔排序
5. 归并排序
6. 快速排序
7. 堆排序
8. 计数排序
9. 桶排序
10. 基数排序

数组 171

动态规划 139

数学 129

字符串 123

树 114

哈希表 104

深度优先搜索 99

二分查找 67

双指针 56

贪心算法 53

广度优先搜索 49

栈 49

回溯算法 41

设计 38

链表 35

堆 34

位运算 32

排序 30

图 26

并查集 22

分治算法 18

字典树 16

Sliding Window 15

递归 14

Ordered Map 10

线段树 10

队列 9

极小化极大 8

树状数组 6

Random 6

拓扑排序 5

脑筋急转弯 4

几何 3

Line Sweep 3

Rejection Sampling 2

蓄水池抽样 2

二叉搜索树 1

记忆化 1

## 2. 操作系统 2.1 基础

★★★ 进程与线程的本质区别、以及各自的使用场景。

★☆☆ 进程状态。

★★★ 进程调度算法的特点以及使用场景。

★☆☆ 线程实现的方式。

- ★★☆☆ 协程的作用。

下面是关于算法的拓展，具体的指导分类

📁 swift	Fixing the size getter leads to infinite loop.	3 months ago
📁 typescript	add insert sort and bubble sort	7 days ago
📄 .gitignore	java中public类名必须跟文件名一致，将不一致的StackBasedOnLinkedList修改成一致；在...	2 months ago
📄 DynamicStackBaseArray.java	fix the bug	3 months ago
📄 LICENSE	Initial commit	8 months ago
📄 README.md	Update README.md	a day ago
📄 StackBaseArray.java	顺序栈实现	3 months ago
📄 f21	Create f21	7 months ago

 README.md

# 数据结构和算法必知必会的50个代码实现

微信搜索我的公众号“小争哥”，或者微信扫描下面二维码, 获取更多压箱底的干货分享

前Google工程师，5万+人跟着学的《数据结构和算法之美》专栏作者



## 数组

---

- 实现一个支持动态扩容的数组
- 实现一个大小固定的有序数组，支持动态增删改操作
- 实现两个有序数组合并为一个有序数组

## 链表

---

- 实现单链表、循环链表、双向链表，支持增删操作
- 实现单链表反转
- 实现两个有序的链表合并为一个有序链表
- 实现求链表的中间结点

## 栈

---

- 用数组实现一个顺序栈

- 用链表实现一个链式栈
- 编程模拟实现一个浏览器的前进、后退功能

## 队列

---

- 用数组实现一个顺序队列
- 用链表实现一个链式队列
- 实现一个循环队列

## 递归

---

- 编程实现斐波那契数列求值 $f(n)=f(n-1)+f(n-2)$
- 编程实现求阶乘 $n!$
- 编程实现一组数据集合的全排列

## 排序

---

- 实现归并排序、快速排序、插入排序、冒泡排序、选择排序
- 编程实现 $O(n)$ 时间复杂度内找到一组数据的第K大元素

## 二分查找

---

- 实现一个有序数组的二分查找算法
- 实现模糊二分查找算法（比如大于等于给定值的第一个元素）

## 散列表

---

- 实现一个基于链表法解决冲突问题的散列表
- 实现一个LRU缓存淘汰算法

## 字符串

---

- 实现一个字符集，只包含a~z这26个英文字母的Trie树
- 实现朴素的字符串匹配算法

## 二叉树

---

- 实现一个二叉查找树，并且支持插入、删除、查找操作
- 实现查找二叉查找树中某个节点的后继、前驱节点
- 实现二叉树前、中、后序以及按层遍历

## 堆

---

- 实现一个小顶堆、大顶堆、优先级队列
- 实现堆排序
- 利用优先级队列合并K个有序数组
- 求一组动态数据集合的最大Top K

## 图

---

- 实现有向图、无向图、有权图、无权图的邻接矩阵和邻接表表示方法
- 实现图的深度优先搜索、广度优先搜索
- 实现Dijkstra算法、A\*算法
- 实现拓扑排序的Kahn算法、DFS算法



## 回溯

---

- 利用回溯算法求解八皇后问题
- 利用回溯算法求解0-1背包问题

## 分治

---

- 利用分治算法求一组数据的逆序对个数

## 动态规划

---

- 0-1背包问题
- 最小路径和
- 编程实现莱文斯坦最短编辑距离
- 编程实现查找两个字符串的最长公共子序列
- 编程实现一个数据序列的最长递增子序列

- ★★☆☆ 常见进程同步问题。
- ★★★★★ 进程通信方法的特点以及使用场景。
- ★★★★★ 死锁必要条件、解决死锁策略，能写出和分析死锁的代码，能说明在数据库管理系统或者 Java 中如何解决死锁。
- ★★★★ 虚拟内存的作用，分页系统实现虚拟内存原理。
- ★★★★ 页面置换算法的原理，特别是 LRU 的实现原理，最好能手写，再说明它在 Redis 等作为缓存置换算法。
- ★★★★ 比较分页与分段的区别。
- ★★★★ 分析静态链接的不足，以及动态链接的特点。

## 2.2 Linux

- ★★☆☆ 文件系统的原理，特别是 inode 和 block。数据恢复原理。
- ★★★★ 硬链接与软链接的区别。
- ★★☆☆ 能够使用常用的命令，比如 cat 文件内容查看、find 搜索文件，以及 cut、sort 等管线命令。了解 grep 和 awk 的作用。
- ★★★★ 僵尸进程与孤儿进程的区别，从 SIGCHLD

分析产生僵尸进程的原因。

### 3. 网络 3.1 基础

★★★ 各层协议的作用，以及 TCP/IP 协议的特点。

- ★★☆☆ 以太网的特点，以及帧结构。
- ★★☆☆ 集线器、交换机、路由器的作用，以及所属的网络层。
- ★★☆☆ IP 数据报常见字段的作用。

★☆☆ ARP 协议的作用，以及维护 ARP 缓存的过程。

★★☆☆ ICMP 报文种类以及作用；和 IP 数据报的关系；Ping 和 Traceroute 的具体原理。

★★★ UDP 与 TCP 比较，分析上层协议应该使用 UDP 还是 TCP。★★★ 理解三次握手以及四次挥手具体过程，三次握手的原因、四次挥手原因、TIME\_WAIT 的作用。

★★★ 可靠传输原理，并设计可靠 UDP 协议。

★★☆ TCP 拥塞控制的作用，理解具体原理。

★★☆ DNS 的端口号;TCP 还是 UDP;作为缓存、负载均衡。

## 3.2 HTTP

★★★ GET 与 POST 比较:作用、参数、安全性、幂等性、可缓存。

★★☆ HTTP 状态码。

★★★ Cookie 作用、安全性问题、和 Session 的比较。

★★☆ 缓存的Cache-Control 字段，特别是 Expires 和 max-age 的区别。ETag 验证原理。

★★★ 长连接与短连接原理以及使用场景，流水线。

★★★ HTTP 存在的安全性问题，以及 HTTPs 的加密、认证和完整性保护作用。

★★☆ HTTP/1.x 的缺陷，以及 HTTP/2 的特点。 ★★★ HTTP/1.1 的特性。

★★☆ HTTP 与 FTP 的比较。

## 3.3 Socket

★★☆ 五种 IO 模型的特点以及比比较。

★★★ select、poll、epoll 的原理理、比比较、以及使用用场景;epoll 的 水水平触发与边缘触发。

## 4. 数据库 4.1 SQL

- ★★☆ 手手写 SQL 语句句，特别是连接查询与分组查询。
- ★★☆ 连接查询与子子查询的比比较。
- ★★☆ drop、delete、truncate 比比较。
- ★★☆ 视图的作用用，以及何时能更更新视图。

★☆☆ 理理解存储过程、触发器器等作用用。

## 4.2 系统原理理

★★★ ACID 的作用用以及实现原理理。

★★★ 四大大隔离级别，以及不不可重复读和幻影读的出现原因。 ★★☆ 封锁的类型以及粒度，两段锁协议，隐式和显示锁定。 ★★★ 乐观锁与悲观锁。

★★★★ MVCC 原理理，当前读以及快照读，Next-Key Locks 解决幻影 读。

★★☆ 范式理理论。

★★★★ SQL 与 NoSQL 的比比较。

## 4.3 MySQL

★★★★ B+ Tree 原理理，与其它查找树的比比较。★★★★ MySQL 索引以及优化。

★★★★ 查询优化。

★★★★ InnoDB 与 MyISAM 比比较。

★★☆ 水水平切分与垂直切分。

★★☆ 主从复制原理理、作用用、实现。★☆☆ redo、undo、binlog 日日志的作用用。

## 4.4 Redis

★★☆ 字典和跳跃表原理理分析。★★★★ 使用用场景。

★★★★ 与 Memchached 的比比较。★☆☆ 数据淘汰机制。

★★☆ RDB 和 AOF 持久化机制。★★☆ 事件驱动模型。

★☆☆ 主从复制原理理。

★★★★ 集群与分布式。

★★☆ 事务原理理。★★★★ 线程安全问题。

## 5. 面面向对象 5.1 思想

★★★ 面面向对象三大大特性 ★☆☆ 设计原则

## 5.2 设计模式

★★☆ 设计模式的作用用。

★★★ 手手写单例例模式，特别是双重检验锁以及静态内部类。★★★ 手手写工工厂厂模式。

★★★ 理理解 MVC，结合 SpringMVC 回答。

★★★ 理理解代理理模式，结合 Spring 中的 AOP 回答。

★★★ 分析 JDK 中常用用的设计模式，例例如装饰者模式、适配器器模 式、迭代器器模式等。

## 6 系统设计以及基本工具使用



### 系统设计

---

- 系统设计基础
- 分布式
- 集群
- 攻击技术
- 缓存
- 消息队列



### 工具

---

- Git
- Docker