

特别放送 | 大厂面试到底在考些什么？

2020-06-17 陶辉

系统性能调优必知必会

[进入课程 >](#)



讲述：陶辉

时长 08:54 大小 8.16M



你好，我是陶辉。这节课我们换换脑子，聊一个相对轻松点的话题——大厂面试。

2004 年我毕业于西安交通大学计算机科学与技术专业，此后 16 年来既在华为、腾讯、思科、阿里巴巴这样的大厂工作过，也在两家几十人的创业公司工作过，在这种对比下，我对大厂面试的考核点很有心得体会。

作为候选人我拿到过很多大厂 offer，作为面试官也考核过数百位同学的技术水平，因此，今天这节课我会兼顾面试官与候选人的视角，分享如何拿下一线大厂的技术面试。



大厂面试到底在考些什么？

相信绝大多数同学都经历过技术面试，你肯定发现，小厂与大厂的面试题差距很大，其中，**大厂特别关注程序性能**，为什么呢？在我看来有这样 3 个原因：

首先，大厂产品的用户基数大，任何微小的性能提升都会被庞大的用户数放大。因此，员工具备性能优先的思维，有利于提升产品竞争力。

其次，大厂经常会重新造轮子，不管原因是什么，造轮子都需要深厚的底层知识，而性能是其中的核心要素。而且，愿意花时间去掌握底层知识的候选人，学习动力更强，潜力也会更好。

最后，大厂待遇好，成长空间大，是典型的稀缺资源，大家都打破了头往里挤。在这样优中选优的情况下，有区分度的性能题就是最好的面试题，通过快速筛选不同档次的候选人，可以节约招聘成本。

那么，对于候选人来说，到底怎样才能答好性能面试题呢？首先，背网上流传的大厂面试题，绝对不是个好主意，这是因为大厂的面试题并不是固定的，往往都是考官自备的面试题，这与每位考官的个人经历有关，所以你押中面试题的概率非常低。

而且，面试并不是为了找出最优秀的那位候选人（这样的候选人手里往往拿着许多优厚的 offer，签下他并不容易），而是将大量的候选人分出层次，再按照团队的业务发展、技术方向、薪资规划来发放 offer。这样的话，面试题就必须是开放的，在各个层次上都有考核点，有内涵更有外延，任何一个点考官都可以展开了聊上个把小时。你背的知识点，未必是考官感兴趣会展开了问的点。

所以，大厂面试考核的是技能、潜力，而不是知识，面试前的刷题不是为了背答案，而是通过练习来提升技能！

下面我就以 1 道算法题为例，带你看看大厂面试中都在考哪些技能点。

举例：1 道算法题可以考核多少知识点？

题目：请用你熟悉的一门编程语言实现 Fibnacci 函数。

Fibnacci 是中学代数提过的一个函数，在自然界中广泛存在，美学中的黄金分割点也与它相关。可能有些同学还不熟悉 Fibnacci 函数，它的定义如下：

$$\text{Fib}(n) = \begin{cases} 0 & \text{if } n = 0, \\ 1 & \text{if } n = 1, \\ \text{Fib}(n - 1) + \text{Fib}(n - 2) & \text{otherwise.} \end{cases}$$


比如 $\text{Fib}(6) = \text{Fib}(5) + \text{Fib}(4) = 5 + 3 = 8$ 。

我个人非常喜欢用这道面试题，因为它有很好的区分度，至少能考核候选人 6 个方面的能力。

首先它像所有编码题一样，可以判断候选人是否至少熟练使用一门编程语言，特别是在不依赖编辑器错误提示的情况下，能不能在白板上手写出高质量的代码。这通常是大厂的基本要求。

其次，Fibonacci 函数很显然非常适合用递归函数实现，大多数候选人都可以写出递归函数，比如：

```
1 Fib(n) {  
2     if (n <= 0) return 0;  
3     if (n == 1) return 1;  
4     return Fib(n-1)+Fib(n-2);  
5 }
```

 复制代码

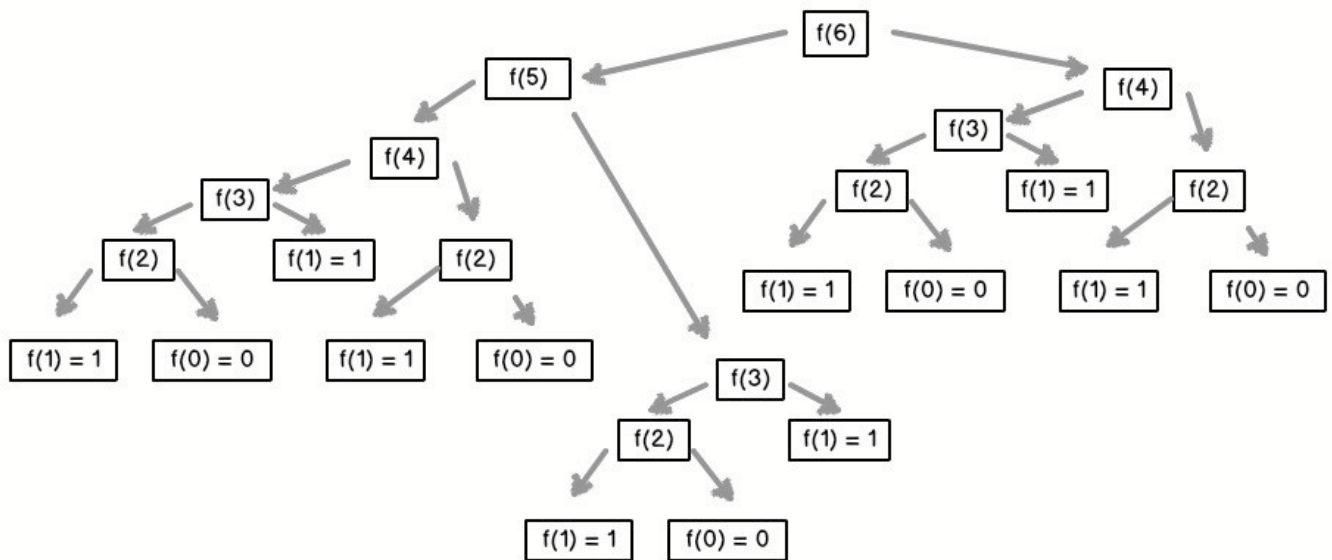
我们可以看到递归函数其实并不难，然而面试官会很自然地追问 2 个问题，这才是考核点：

首先，递归函数在系统层面有什么问题？

这其实是在考你是否知道栈溢出问题。每调用一次函数，需要将函数参数、返回值压栈，而操作系统为每个线程分配的栈空间是有限的，比如 Linux 通常只有 8MB。因此，当数字 n 过大时，很容易导致 StackOverflow 错误。

其次，这段递归代码的时间复杂度是多少？

如果你还没有时间复杂度的概念，请再次阅读🔗[第 3 讲]。《算法导论》中的递归树很适合用于猜测算法的时间复杂度，下图是 Fibonacci(6) 展开的计算量，可见，递归树中所有节点的数量，就是递归函数的时间复杂度：



图片来源：<https://medium.com/launch-school/recursive-fibonacci-method-explained-d82215c5498e>

如果你仔细数一数，会发现随着 n 的增大，节点数大致接近 2^n 个，因此其时间复杂度为 $O(2^n)$ 。当然，树中的节点数与 Fibonacci 数列相关，大致为 $O(1.618^n)$ 个（书中还介绍了 2 种求解递归函数时间复杂度的方法：代入法和主定理，请参考书中第 4 章）。

最后，这道题还可以考察候选人能否运用逆向思维，通过一个循环实现递归函数的效果。递归法的时间复杂度之所以达到了 $O(2^n)$ ，是因为做了大量的重复运算。比如求 Fibonacci(6) 时，Fibonacci(2) 重复执行了 4 次。如果对 n 从小向大做递推运算，重复使用已经计算完成的数字，就能大大减少计算量，如下所示：

📄 复制代码

```
1 Fib(n) {
2   if (n <= 0) return 0;
3   if (n == 1) return 1;
4   prev = 0;
5   cur = 1;
6   for (i = 2; i <= n; i = i + 1) {
7     next = cur + prev;
8     prev = cur;
9     cur = next;
10  }
```

```
11     return next;
12 }
```

这样，只通过 1 次循环，就能以 $O(n)$ 的时间复杂度完成 Fibonacci 数列的计算。这与动态规划的实现思路是一致的。注意，使用递推法时如果不小心，会采用数组存放计算出的每个结果，这样空间复杂度就从 $O(1)$ 到了 $O(n)$ ，这也是一个考核点。

当然，我们还可以通过公式直接计算出数列的值：

$$F_n = \frac{1}{\sqrt{5}} \left[\left(\frac{1 + \sqrt{5}}{2} \right)^n - \left(\frac{1 - \sqrt{5}}{2} \right)^n \right]$$

但纯粹背下这个公式并没有意义，因为你写出来后，会有 2 个问题等着你：

首先，如何使用高等数学推导出这个公式？如果你能够答出来，那么证明你的数学功底很好，在数据挖掘、人工智能人才短缺的当下，这对你在大厂内部的职业发展很有好处！

其次，上面这个公式有大量的浮点运算，在数学中数字可以是无限长的，但在计算机工程体系中，任何类型都有最大长度（比如浮点类型通常是 64 个比特位），所以对于根号 5 这样的无理数，小数点后的数字会出现四舍五入而不精确，而且当 n 非常大时，有限的内存还会导致数据溢出。因此上述的公式法并不能直接使用，如果你能回答出适合计算机使用的矩阵解法（请参考 [@wiki](#)，这里不再列出矩阵法的细节），那就更完美了。

可见，这么一道简单的题目，就可以考察递归编码能力、递推解法、公式解法、矩阵解法、时间复杂度的推算、计算机浮点运算特性等许多知识点。而且，随着你回答时涉及到更多的知识点，面试官会基于自己的经验进一步延伸提问。所以我不推荐你押题，把基础技能掌握好才是最有效的面试备战法。

这道题目只是入门的算法题，如果你应聘的是 Google、头条之类非常重视算法的公司，那么你还必须掌握动态规划、贪心算法、图算法等高级算法等等。

小结

今天我们只是详细讲解了算法题，其实从系统工程、网络协议上也可以从性能优化这个方向快速区分出候选人的能力水平。比如：

我在开课直播时提出的关于并发的的问题，多进程和多线程、协程实现的并发编程，各自的优势和劣势是什么（你可以参考 [🔗\[第 5 讲\]](#)）？

或者 TCP 连接的 `close_wait` 状态出现时应当如何解决（你可以参考 [🔗\[第 10 讲\]](#)）？

这些都在考察你如何通过操作系统，协调使用系统资源的能力。

另外，除了硬核的知识技能外，你也不要忽略软技能，这也能在面试中加分。比如，任何大厂都非常强调团队协作，如果员工遇到难题时，只会闷头冥想，这样的时间成本太高，既有可能延迟项目进度，也不利于充分发挥大厂高手如林、资源丰富的优势。所以，如果你在面试中，表现出善于沟通、乐于求助的特性，都是加分项。

以上就是我对大厂面试的一些沉淀和思考，不知道你有没有感同身受呢？如果你在面试中也遇到过一些特别开放、有区分度的面试题，或者作为面试官你有哪些喜欢用的面试题，欢迎分享出来，我们一起探讨。

感谢阅读，如果你觉得这节课有所收获，也欢迎把它分享给你的朋友。

618 课程特惠

618 好课 5 折起

优惠口令立减 ¥15

618gogogo



© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 18 | 如何通过gRPC实现高效远程过程调用？

下一篇 期中考试 | 行至半程，你的收获如何呢？

精选留言 (2)

写留言



忆水寒

2020-06-17

写的很好👍

展开



1



leslie

2020-06-19

不光大厂，其实小企业同样会去看中这点。一个团队其实各种人都是需要的，如果只是会单打独斗反而会给团队/项目带来许多问题，毕竟人是属于社会。

展开



