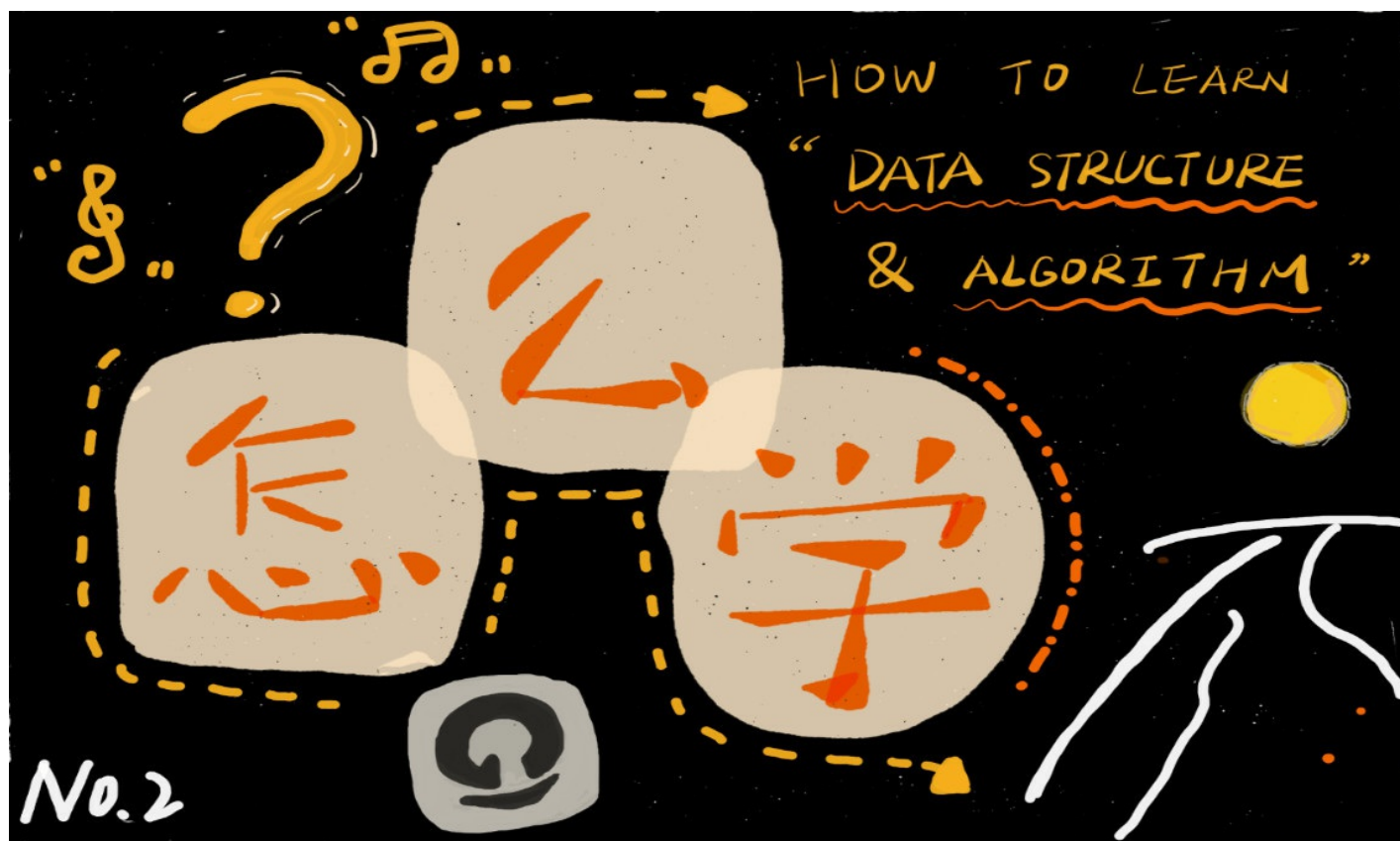


02讲如何抓住重点，系统高效地学习数据结构与算法



你是否曾跟我一样，因为看不懂数据结构和算法，而一度怀疑是自己太笨？实际上，很多人在第一次接触这门课时，都会有这种感觉，觉得数据结构和算法很抽象，晦涩难懂，宛如天书。正是这个原因，让很多初学者对这门课望而却步。

我个人觉得，其实真正的原因是你**没有找到好的学习方法，没有抓住学习的重点**。实际上，数据结构和算法的东西并不多，常用的、基础的知识点更是屈指可数。只要掌握了正确的学习方法，学起来并没有看上去那么难，更不需要什么高智商、厚底子。

还记得大学里每次考前老师都要划重点吗？今天，我就给你划划我们这门课的重点，再告诉你一些我总结的学习小窍门。相信有了这些之后，你学起来就会有有的放矢、事半功倍了。

什么是数据结构？什么是算法？

大部分数据结构和算法教材，在开篇都会给这两个概念下一个明确的定义。但是，这些定义都很抽象，对理解这两个概念并没有实质性的帮助，反倒会让你陷入死抠定义的误区。毕竟，我们现在学习，并不是为了考试，所以，概念背得再牢，不会用也就没什么用。

虽然我们说没必要深挖严格的定义，但是这并不等于不需要理解概念。下面我就从广义和狭义两个层面，来帮你理解数据结构与算法这两个概念。

从广义上讲，数据结构就是指一组数据的存储结构。算法就是操作数据的一组方法。

图书馆储藏书籍你肯定见过吧？为了方便查找，图书管理员一般会将书籍分门别类进行“存储”。按照一定规律编号，就是书籍这种“数据”的存储结构。

那我们如何来查找一本书呢？有很多种办法，你当然可以一本一本地找，也可以先根据书籍类别的编号，是人文，还是科学、计算机，来定位书架，然后再依次查找。笼统地说，这些查找方法都是算法。

从狭义上讲，也就是我们专栏要讲的，是指某些著名的数据结构和算法，比如队列、栈、堆、二分查找、动态规划等。这些都是前人智慧的结晶，我们可以直接拿来用。我们要讲的这些经典数据结构和算法，都是前人从很多实际操作场景中抽象出来的，经过非常多的求证和检验，可以高效地帮助我们解决很多实际的开发问题。

那数据结构和算法有什么关系呢？为什么大部分书都把这两个东西放到一块儿来讲呢？

这是因为，数据结构和算法是相辅相成的。**数据结构是为算法服务的，算法要作用在特定的数据结构之上。**因此，我们无法孤立数据结构来讲算法，也无法孤立算法来讲数据结构。

比如，因为数组具有随机访问的特点，常用的二分查找算法需要用数组来存储数据。但如果我们选择链表这种数据结构，二分查找算法就无法工作了，因为链表并不支持随机访问。

数据结构是静态的，它只是组织数据的一种方式。如果不在它的基础上操作、构建算法，孤立存在的数据结构就是没用的。

现在你对数据结构与算法是不是有了比较清晰的理解了呢？有了这些储备，下面我们来看看，究竟该怎么学数据结构与算法。

学习这个专栏需要什么基础？

看到数据结构和算法里的“算法”两个字，很多人就会联想到“数学”，觉得算法会涉及到很多深奥的数学知识。那我数学基础不是很好，学起来会不会很吃力啊？

数据结构和算法课程确实会涉及一些数学方面的推理、证明，尤其是在分析某个算法的时间、空间复杂度的时候，但是这个你完全不需要担心。

这个专栏不会像《算法导论》那样，里面有非常复杂的数学证明和推理。我会由浅入深，从概念到应用，一点一点给你解释清楚。你只要有高中数学水平，就完全可以学习。

当然，我希望你最好有些编程基础，如果有项目经验就更好了。这样我给你讲数据结构和算法如何提高效率、如何节省存储空间，你就会有很直观的感受。因为，对于每个概念和实现过程，我都会从实际场景出发，不仅教你“**是什么**”，还会教你“**为什么**”，并且告诉你遇到同类型问题应该“**怎么做**”。

学习的重点在什么地方？

提到数据结构和算法，很多人就很头疼，因为这里面的内容实在是太多了。这里，我就帮你梳理一下，应该先学什么，后学什么。你可以对照看看，你属于哪个阶段，然后有针对性地进行学习。

想要学习数据结构与算法，**首先要掌握一个数据结构与算法中最重要的概念——复杂度分析。**

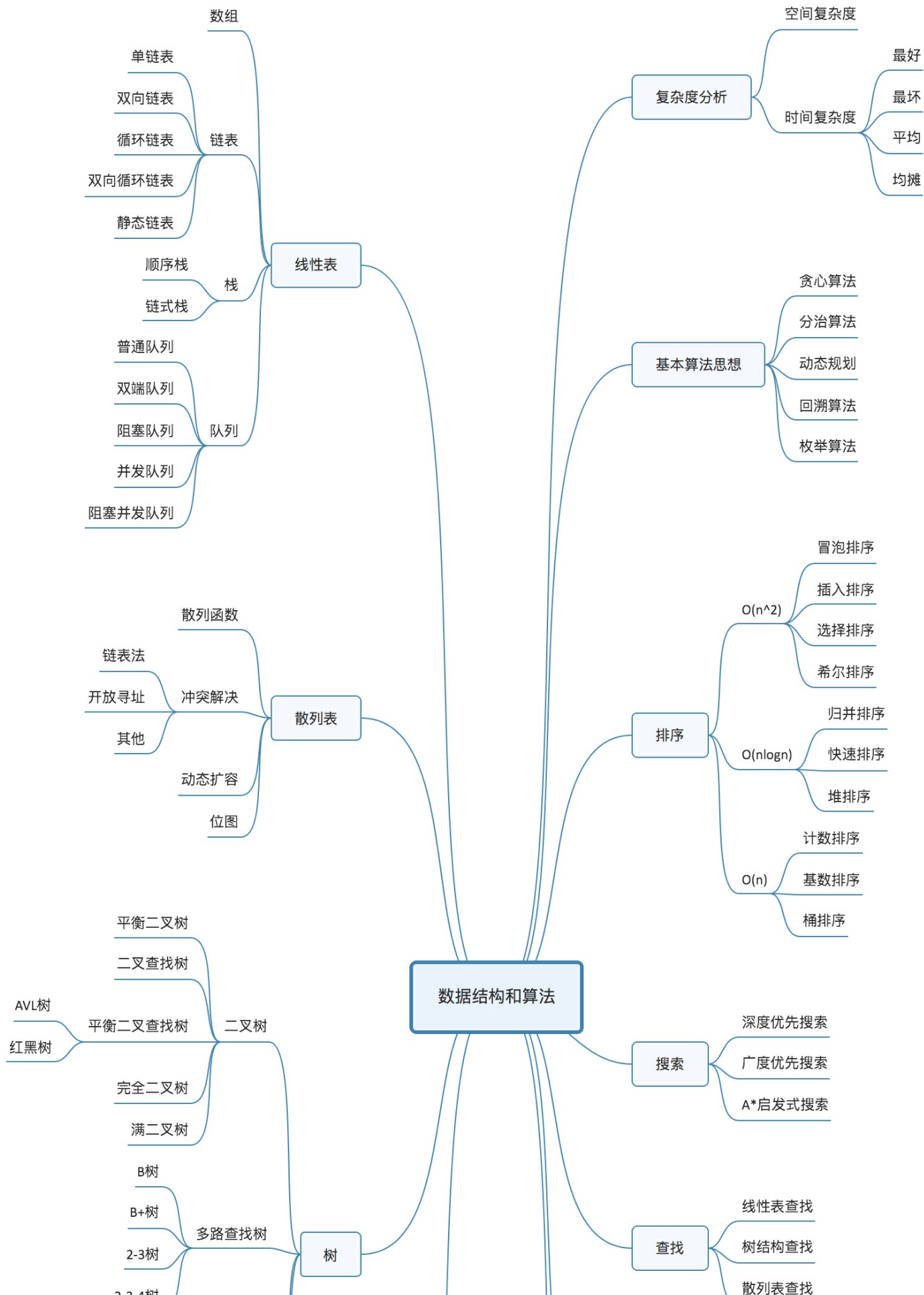
这个概念究竟有多重要呢？可以这么说，它几乎占了数据结构和算法这门课的半壁江山，是数据结构和算法学习的精髓。

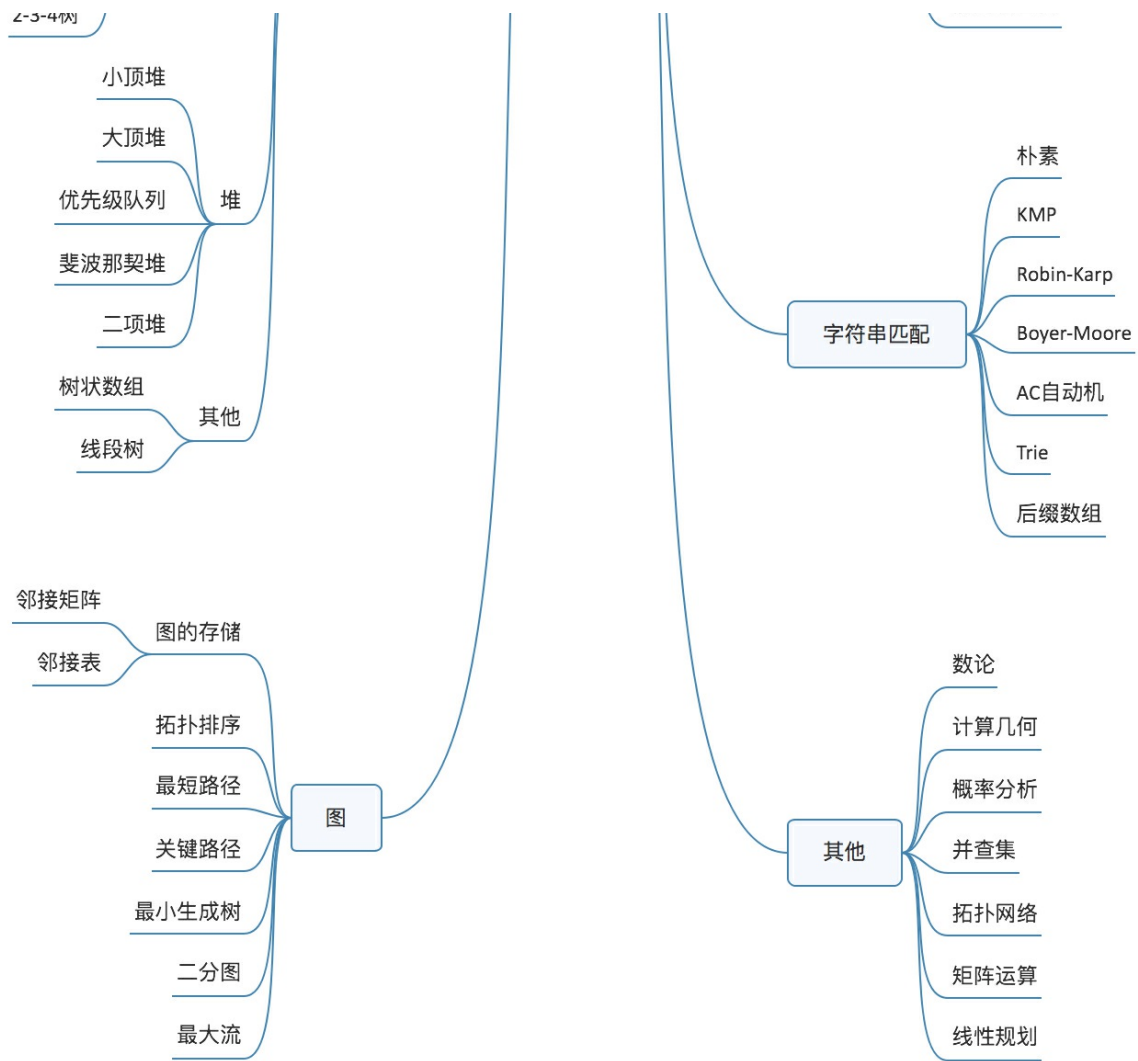
数据结构和算法解决的是如何更省、更快地存储和处理数据的问题，因此，我们就需要一个考量效率和资源消耗的方法，这就是复杂度分析方法。所以，如果你只掌握了数据结构和算法的特点、用法，但是没有学会复杂度分析，那就相当于只知道操作口诀，而没掌握心法。只有把心法了然于胸，才能做到无招胜有招！

所以，复杂度分析这个内容，我会用很大篇幅给你讲透。你也一定要花大力气来啃，必须要拿下，并且要搞得非常熟练。否则，后面的数据结构和算法也很难学好。

搞定复杂度分析，下面就要进入**数据结构与算法的正文内容**了。

为了让你对数据结构和算法能有个全面的认识，我画了一张图，里面几乎涵盖了所有数据结构和算法书籍中都会讲到的知识点。





(图谱内容较多，建议长按保存后浏览)

但是，作为初学者，或者一个非算法工程师来说，你并不需要掌握图里面的所有知识点。很多高级的数据结构与算法，比如二分图、最大流等，这些在我们平常的开发中很少会用到。所以，你暂时可以不用看。我还是那句话，咱们学习要学会找重点。如果不分重点地学习，眉毛胡子一把抓，学起来肯定会比较吃力。

所以，结合我自己的学习心得，还有这些年的面试、开发经验，我总结了**20个最常用的、最基础**数据结构与算法，不管是应付面试还是工作需要，只要集中精力逐一攻克这20个知识点就足够了。

这里面有10个数据结构：**数组、链表、栈、队列、散列表、二叉树、堆、跳表、图、Trie树**；10个算法：**递归、排序、二分查找、搜索、哈希算法、贪心算法、分治算法、回溯算法、动态规划、字符串匹配算法**。

掌握了这些基础的数据结构和算法，再学更加复杂的数据结构和算法，就会非常容易、非常快。

在学习数据结构和算法的过程中，你也要注意，不要只是死记硬背，不要为了学习而学习，而是要学习它的“来历”“自身的特点”“适合解决的问题”以及“实际的应用场景”。对于每一种数据结构或算法，我都会从这几个方面进行详细讲解。只要你掌握了我每节课里讲的内容，就能在开发中灵活应用。

学习数据结构和算法的过程，是非常好的思维训练的过程，所以，千万不要被动地记忆，要多辩证地思考，多问为什么。如果你一直这么坚持做，你会发现，等你学完之后，写代码的时候就会不由自主地考虑到很多性能方面的事情，时间复杂度、空间复杂度非常高的垃圾代码出现的次数就会越来越少。你的编程内功就真正得到了修炼。

一些可以让你事半功倍的学习技巧

前面我给你划了学习的重点，也讲了学习这门课需要具备的基础。作为一个过来人，现在我就给你分享一下，专栏学习的一些技巧。掌握了这些技巧，可以让你化被动为主动，学起来更加轻松，更加有动力！

1.边学边练，适度刷题

“边学边练”这一招非常有用。建议你每周花1~2个小时的时间，集中把这周的三节内容涉及的数据结构和算法，全都自己写出来，用代码实现一遍。这样一定会比单纯地看或者听的效果要好很多！

有面试需求的同学，可能会问了，那我还要不要去刷题呢？

我个人的观点是可以“适度”刷题，但一定不要浪费太多时间在刷题上。我们学习的目的还是掌握，然后应用。除非你要面试Google、Facebook这样的公司，它们的算法题目非常非常难，必须大量刷题，才能在短期内提升应试正确率。如果是应对国内公司的技术面试，即便是BAT这样的公司，你只要彻底掌握这个专栏的内容，就足以应对。

2.多问、多思考、多互动

学习最好的方法是，找到几个人一起学习，一块儿讨论切磋，有问题及时寻求老师答疑。但是，离开大学之后，既没有同学也没有老师，这个条件就比较难具备了。

不过，这也就是咱们专栏学习的优势。专栏里有很多跟你一样的学习者。你可以多在留言区写下自己的疑问、思考和总结，也可以经常看看别人的留言，和他们进行互动。

除此之外，如果你有疑问，你可以随时在留言区给我留言，我只要有空就会及时回复你。你不要担心问的问题太小白。因为我初学的时候，也常常会被一些小白问题困扰。不懂一点都不丢人，只要你勇敢提出来，我们一起解决了就可以了。

我也会力争每节课都最大限度地给你讲透，帮你扫除知识盲点，而你要做的就是，避免一知半解，要想尽一切办法去搞懂我讲的所有内容。

3.打怪升级学习法

学习的过程中，我们碰到最大的问题就是，坚持不下来。是的，很多基础课程学起来都非常枯燥。为此，我自己总结了一套“打怪升级学习法”。

游戏你肯定玩过吧？为什么很多看起来非常简单又没有乐趣的游戏，你会玩得不亦乐乎呢？这是因为，当你努力打到一定级别之后，每天看着自己的经验值、战斗力在慢慢提高，那种每天都在一点一点成长的成就感就不由自主地产生了。

所以，我们在枯燥的学习过程中，也可以给自己设立一个切实可行的目标，就像打怪升级一样。

比如，针对这个专栏，你就可以设立这样一个目标：每节课后的思考题都认真思考，并且回复到留言区。当你看到很多人给你点赞之后，你就会为了每次都能发一个漂亮的留言，而更加认真地学习。

当然，还有很多其他的目标，比如，每节课后都写一篇学习笔记或者学习心得；或者你还可以每节课都找一下我讲得不对、不合理的地方……诸如此类，你可以总结一个适合你的“打怪升级攻略”。

如果你能这样学习一段时间，不仅能收获到知识，你还会有意想不到的成就感。因为，这其实帮你改掉了一点学习的坏习惯。这个习惯一旦改掉了，你的人生也会变得不一样。

4.知识需要沉淀，不要想试图一下子掌握所有

在学习的过程中，一定会碰到“拦路虎”。如果哪个知识点没有怎么学懂，不要着急，这是正常的。因为，想听一遍、看一遍就把所有知识掌握，这肯定是不可能的。学习知识的过程是反复迭代、不断沉淀的过程。

如果碰到“拦路虎”，你可以尽情地在留言区问我，也可以先沉淀一下，过几天再重新学一遍。所谓，书读百遍其义自见，我觉得是很有道理的！

我讲的这些学习方法，不仅仅针对咱们这一个课程的学习，其实完全适用任何知识的学习过程。你可以通过这个专栏的学习，实践一下这些方法。如果效果不错，再推广到之后的学习过程中。

内容小结

今天，我带你划了划数据结构和算法的学习重点，复杂度分析，以及10个数据结构和10个算法。

这些内容是我根据平时的学习和工作、面试经验积累，精心筛选出来的。只要掌握这些内容，应付日常的面试、工作，基本不会有问题。

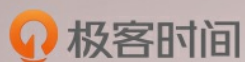
除此之外，我还给你分享了我总结的一些学习技巧，比如边学边练、多问、多思考，还有两个比较通用的学习方法，打怪升级法和沉淀法。掌握了这些学习技巧，可以让你学习过程中事半功倍。所以，你一定要好好实践哦！

课后思考

今天的内容是一个准备课，从下节开始，我们就要正式开始学习精心筛选出的这20个数据结构和算法了。所以，今天给你布置一个任务，对照我上面讲的“打怪升级学习法”，请思考一下你自己学习这个专栏的方法，让我们一起在留言区立下Flag，相互鼓励！

另外，你在之前学习数据结构和算法的过程中，遇到过什么样的困难或者疑惑吗？

欢迎留言和我分享，我会第一时间给你反馈。



数据结构与算法之美

为工程师量身打造的数据结构与算法私教课

王争

前 Google 工程师



新版升级：点击「👤请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

精选留言



强

1、所有数据结构与算法用C++实现一遍；

2、所有数据结构与算法用Python实现一遍；

3、学完就辞职。

2018-09-24 12:39



wean

是什么：

数据结构指的是“一组数据的存储结构”，算法指的是“操作数据的一组方法”。

数据结构是为算法服务的，算法是要作用再特定的数据结构上的。

学什么：

1. 效率和资源消耗的度量衡--复杂度分析。

2. 最常用、最基础的20个数据结构与算法，学习他们的：“来历”、“特点”、“适合解决什么问题”和“实际的应用场景”。

数据结构：数组、链表、栈、队列、散列表、二叉树、堆、跳表、图、Trie树

算法：递归、排序、二分查找、搜索、哈希算法、贪心算法、分治算法、回溯算法、动态规划、字符串匹配算法

怎么学

1. 边学边练，每周花 1~2 小时集中攻关节课涉及的数据结构和算法，全部写出来。

2. 主动提问、多思考、多互动。在留言区增加自己的留言。

3. 自我激励，每次学习完做一篇学习笔记。

4. 沉下心不要浮躁，先把这些基础的数据结构和算法，还有学习方法熟练掌握后，再追求更高层次。

flag：每篇文章必写学习笔记，每周必实现一次该周代码。

2018-09-24 06:54



Jason X

BAT我明年年初一定进入一个

2018-09-24 00:34



一条咸鱼

非科班。。马上就28岁了，一直有个计算机梦。来这学习不为别的，只为了能慢慢成为一位合格的计算机人

2018-09-24 10:15



zyzheng

“学习最好的方法是，找到几个人一起学习，一块儿讨论切磋，有问题及时寻求老师答疑。”

老师说的这个方法非常赞同，之前在学其他内容时经常遇到个人单独学习坚持不下来的问题。这次和公司里有兴趣的同事一起成立了一个五人学习小组，计划每周组织一个小时交流会，每人讲10分钟，自由讨论10分钟，希望能对自己和大家的学习都有帮助，希望能坚持下去

2018-09-24 07:28

作者回复

太赞太用心了

2018-09-24 09:55



Eid

敲代码，回想起自己大学的时候，课本上的每一段代码都敲一边！甚至敲完了整本《C++ primer》

2018-09-24 00:33



觉良

当我们要谈一个事物/概念的时候，需要问自己三个终极问题--是什么？为什么？怎么样？

什么是数据结构和算法

数据结构，就是一组数据的存储结构。

算法，就是操作数据的一组方法。

数据结构是为算法服务的，算法要作用在特定的数据结构之上。

为什么需要数据结构和算法

来谈谈应用层面的原因。在计算机科学和互联网迅猛发展下，需要计算的数据量越来越庞大。但是计算机的计算能力是有限的，这么多的数据计算，需要越来越多的计算机，需要越来越长的计算时间，注重效率的我们需要尽可能的提高计算效率。其中重要的一项，就是使用合适的数据结构和算法。选用合适的数据结构和算法，特别是在处理体量非常庞大的数据的时候，可以极大提高计算效率。那么，第三个问题来了，我们怎么选用合适的数据结构和算法？有什么衡量标准吗？

怎么样衡量数据结构和算法

需要引入一个衡量的标准(metric)---时间复杂度和空间复杂度。

学习数据结构和算法的基石，就是要学会`复杂度分析`。知道怎么去分析复杂度，才能作出正确的判断，在特定的场景下选用合适的正确的算法。而不是盲目的死记烂背，机械操作。

在本专栏中，重点学习20个最常用的最基础的数据结构和算法，需要我们逐一攻克。

10个数据结构: 数组，链表，栈，队列，散列表，二叉树，堆，跳表，图，Trie树

10个算法: 递归，排序，二分查找，搜索，哈希算法，贪心算法，分治算法，回溯算法，动态规划，字符串匹配算法

2018-09-24 16:53

作者回复

写的好!

2018-09-24 18:10



吉祥

Flag: 课程内容用 python 实现一遍，有不懂的及时留言提问，每周写总结。

2018-09-24 07:59



墨墨

对于比较笨的我来说，我的学习方法就是边学边练，这个方法务实有效。在此立下flag，不学会数据结构与算法，就掉光头发！哈哈

2018-09-24 02:03



与路同飞

完美主义在作怪。在看《算法》这本书的时候，总想着做完所有习题在看后面的章节。导致后面没坚持下去

2018-09-24 02:27



黄海娜

学习了，也理解了，当时可以写出代码来，但是长时间就忘记了，这个难道真的如果日常不用的话，的确会忘记，那不是就要先当公式一样记下来再说？

2018-09-24 10:54

作者回复

完全不需要死记硬背的，我也记不住快排，红黑树，但是只要你掌握分析的能力，等你真的需要的时候，花不到半个小时就弄懂了。你要记住的是，这些算法的特点，应用场景，用到了能想到他就好了。

2018-09-24 11:01



Joshua 兆甲

需要教材吗？老师？

自己学习的方法: 排个队想到队列，火车进站想到堆栈，看到树上开花想到树的某些指标，导航想到贪心，将生活场景与算法和数据结构的场景结合起来。

遇到的困难: 应用和实践不行。你看树是数据结构，而大众有更约定俗成的固有看法，切磋估计不大可行，且自己学习，首先需要有一个代码开发的条件，不断思考练习，不易坚持。

数据结构不容易可视化呈现。靠画图建立初始的结构，靠脑补数据变化的过程，再靠心算将结果呈现为“可视”图景。中间，一般在机器上就是简单plain text输入输出.不是第一信号系统，接受收效率慢。

没有大图景，只有片面观察。即便会了一招半式，获得敢不强，信心不足，真正应用中，或许没有直接上两个循环嵌套来得直接，明了，易懂。容易生出百无一用是书生的瞎想

以上。老师中秋快乐！循环链表代表我的心。

2018-09-25 07:28

作者回复

写得好 你可以看看大话数据结构和算法图解。里面的学习方法跟你的很相似

2018-09-25 09:51



王宇

我的学习法：

- 1.从老师这里获取知识
- 2.学习理解知识，继续思考，提出疑问，先自己解决，实在想不出了问老师
- 3.总结知识
- 4.运用知识，解决问题
- 5.写博客教别人学到的知识

已经意识到算法对面试工作的重要性，也喜欢锻炼自己的思维，想体验不知道哪一天解决了个什么事，突然发现自己变聪明了的快感

2018-09-24 01:19

187J3X1

<https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>

此网站可以将指定算法可视化，能方便理解。麻烦顶上去让更多人看到，谢谢。

2018-10-01 17:17



确认过眼神

本人学的Java，想通过这个课程学习如何把数据结构和算法切实的用到编程中，来实现用空间换时间。

- 1、提高每个接口的访问速度，可否实现？
 - 2、时间复杂度越高，执行效率越低吗？
 - 3、两张表联查后再遍历查询第三张表是否比三张表联查效率高？利用数据结构和算法是否有更优的解决方案？
- 希望老师给出三个问题的答案，伙伴们一起讨论。

2018-09-24 06:59

作者回复

1. 学习的过程可以锻炼你的性能意识，写代码的时候会不自觉的考虑性能问题，所以写出低效代码的情况就少了
2. 理论上是，但实际上时间复杂度与执行时间并不划等号，我后面会讲到，有时候 $O(\log n)$ 的算法要比 $O(1)$ 算法还快
3. 这个没有确切的场景我也不好说，你可以自己测试一下。一般来说，海量数据的存储一般都是分库分表，所以join操作可能就无法用上了。

2018-09-24 11:07



lxb

我想问问用JavaScript真能实现这些算法跟数据结构吗？

2018-09-24 01:13

作者回复

我觉得能 后面如果你觉得不能你就再给我留言吧

2018-09-24 10:04



Yafei

大学学这门课的时候，一碰到那种递归求复杂度就懵逼，这次一定搞定它！

2018-09-24 07:26



五岳寻仙

老师给的思路非常非常清晰，不愧是IT大牛，我很喜欢这种结构化的思维方式，相信同学也有同感！

已经决定严格跟随老师的思路这门课。在执行前遇到三个困难：

1. 想请老师推荐刷题的地方；
2. 想知道每种算法和数据结构在现实中的应用场景，因为很多时候不是学不懂，是积极性不够，所以我觉得只有意识到它的重要作用，才会深入地钻研。
3. 没有交流的对象和途径，希望咱们专栏能有个微信群，方便交流。(我微信jjjzzz2014，期待和各位同学一起交流成长！)

2018-09-24 10:14

作者回复

1. 刷题的话leetcode比较火
2. 会讲的 放心
3. 暂时就在留言区交流吧

2018-09-24 10:53



Silence

Flag：在这几个月内一定要掌握并学会运用文中提到的算法，不管多难都要坚持看下去，这次，不想再逃了！

2018-09-24 07:56



objcoding

把我自己课后练习做过的题目建个仓库丢到GitHub上，慢慢累积。

2018-09-24 01:04