

前言

目前，信息技术已被广泛应用于互联网、金融、航空、军事、医疗等各个领域，未来的应用将更加广泛和深入。并且，很多中小学都开设了计算机语言课程，越来越多的中小学生对编程、算法感兴趣，甚至在 NOIP、NOI 等算法竞赛中大显身手，进入名校深造。对信息技术感兴趣的大学生通常会参加 ACM-ICPC、CCPC、蓝桥杯等算法竞赛，其获奖者更是被各大名企所青睐。

学习算法，不仅可以帮助我们具备较强的思维能力及解决问题的能力，还可以帮助我们快速学习各种新技术，拥有超强的学习能力。

写作背景

很多读者都觉得算法太难，市面上晦涩难懂的各种教材更是“吓退”了一大批读者。实际上，算法并没有我们想象中那么难，反而相当有趣。

每当有学生说看不懂某个算法的时候，笔者就会建议其画图。画图是学习算法最好的方法，因为它可以把抽象难懂的算法展现得生动形象、简单易懂。笔者曾出版《算法训练营：海量图解+竞赛刷题》（入门篇）和《算法训练营：海量图解+竞赛刷题》（进阶篇），很多读者非常喜欢其中的海量图解，更希望看到这两本书的全彩版。经过一年的筹备，笔者对上述书中的所有图片都重新进行了绘制和配色，并精选、修改、补充和拆分上述书中的内容，形成了《算法训练营：入门篇》（全彩版）、《算法训练营：提高篇》（全彩版）和《算法训练营：进阶篇》（全彩版），本书就是其中的《算法训练营：入门篇》（全彩版）。在此衷心感谢各位读者的大力支持！

本书详细讲解常用的算法知识，特别增加了 C++ 基础知识和 STL 部分的内容。如果读者已经熟悉 C++，则可跳过其中的基础章节。本书不是知识点的堆砌，也不是粘贴代码而来的简单题解，而是将知识点讲解和对应的竞赛实例融会贯通，读者可以在轻松阅读本书的同时进行刷题实战，在实战中体会算法的妙处，感受算法之美。

学习建议

学习算法的过程，应该是通过大量实例充分体会遇到问题时该如何分析：用什么数据结构，用什么算法和策略，算法复杂度如何，是否有优化的可能，等等。这里有以下几个建议。

第 1 个建议：学经典，多理解。

算法书有很多，初学者最好选择图解较多的入门书，当然，也可以选择多本书，从多个角度进行对比和学习。先看书中的图解，理解各种经典问题的求解方法，如果还不理解，则可以看视频讲解，理解之后再看代码，尝试自己动手上机运行。如有必要，则可以将算法的求解过程通过图解方式展示出来，以加深对算法的理解。

第 2 个建议：看题解，多总结。

在掌握书中的经典算法之后，可以在刷题网站上进行专项练习，比如练习贪心算法、分治算法、动态规划等方面的题目。算法比数据结构更加灵活，对同一道题目可以用不同的算法解决，算法复杂度也不同。如果想不到答案，则可以看题解，比较自己的想法与题解的差距。要多总结题目类型及最优解法，找相似的题目并自己动手解决问题。

第 3 个建议：举一反三，灵活运用。

通过专项刷题达到“见多识广”，总结常用的算法模板，熟练应用套路，举一反三，灵活运用，逐步提升刷题速度，力争“bug free”（无缺陷）。

本书特色

本书具有以下特色。

（1）完美图解，通俗易懂。本书对每个算法的基本操作都有全彩图解。通过图解，许多问题都变得简单，可迎刃而解。

（2）实例丰富，简单有趣。本书结合了大量竞赛实例，讲解如何用算法解决实际问题，使复杂难懂的问题变得简单有趣，可帮助读者轻松掌握算法知识，体会其中的妙处。

（3）深入浅出，透析本质。本书透过问题看本质，重点讲解如何分析和解决问题。本书采用了简洁易懂的代码，对数据结构的设计和算法的描述全面、细致，而且有算法复杂度分析及优化过程。

（4）实战演练，循序渐进。本书在讲解每个算法后都进行了实战演练，使读者在实战中体会算法的设计思路和使用技巧，从而提高独立思考、动手实践的能力。书中



有丰富的练习题和竞赛题，可帮助读者及时检验对所学知识的掌握情况，为从小问题出发且逐步解决大型复杂性工程问题奠定基础。

(5) 网络资源，技术支持。本书为读者提供了配套源码、课件、视频，并提供了博客、微信群、QQ 群技术支持，可随时为读者答疑解惑。

建议和反馈

写书是极其琐碎、繁重的工作，尽管笔者已经竭力使本书内容、网络资源和技术支持接近完美，但仍然可能存在很多漏洞和瑕疵。欢迎读者反馈关于本书的意见，因为这有利于我们改进和提高，以帮助更多的读者。如果对本书有意见和建议，或者有问题需要帮助，则都可以加入 QQ 群 281607840，也可以致信 rainchxy@126.com 与笔者交流，笔者将不胜感激。

对于本书提供的读者资源，可通过本书封底的“读者服务”获取。

致谢

感谢笔者的家人和朋友在本书写作过程中提供的大力支持。感谢电子工业出版社工作严谨、高效的张国霞编辑，她的认真、负责促成了本书的早日出版。感谢提供了宝贵意见的同事们，感谢提供了技术支持的同学们。感恩遇到这么多良师益友！

前言

第 1 章 C++基础知识.....1

1.1 开启算法之旅.....	1
1.2 常用的数据类型.....	2
1.3 玩转输入和输出.....	2
1.4 常用的运算符.....	3
1.5 选择结构语句.....	5
1.5.1 if 条件语句.....	5
1.5.2 switch 条件语句.....	9
1.6 循环结构语句.....	10
1.6.1 for 语句.....	10
1.6.2 while 语句.....	13
1.6.3 do while 语句.....	14
1.7 巧用数组.....	15
1.7.1 一维数组.....	15
1.7.2 二维数组.....	17
1.8 玩转字符串.....	18
1.8.1 C 风格的字符串.....	19
1.8.2 C++ string 类型的 字符串.....	20
1.9 结构体的应用.....	21
1.10 指针的应用.....	22

第 2 章 算法之美.....24

2.1 算法复杂度.....	24
2.1.1 时间复杂度.....	27

2.1.2 空间复杂度.....	27
2.2 函数.....	30
2.2.1 标准函数.....	30
2.2.2 传值参数.....	31
2.2.3 引用参数.....	31
2.2.4 数组参数.....	32
2.3 递归.....	33
2.3.1 递归函数.....	33
2.3.2 递归的原理.....	33

第 3 章 线性表的应用.....37

3.1 顺序表.....	37
3.1.1 插入.....	38
3.1.2 删除.....	39
3.2 链表.....	40
3.2.1 单链表.....	40
3.2.2 双向链表.....	43
3.2.3 循环链表.....	45
3.2.4 静态链表.....	46
3.3 栈.....	49
3.3.1 入栈.....	49
3.3.2 出栈.....	49
3.3.3 取栈顶元素.....	50
3.4 队列.....	50
3.4.1 顺序队列.....	51
3.4.2 循环队列.....	53



3.5 STL 中的常用函数和容器	56	4.5 二叉搜索树	112
3.5.1 sort()	57	4.5.1 二叉搜索树原理详解	112
3.5.2 vector (向量)	58	4.5.2 查找	112
训练 角谷猜想	59	4.5.3 插入	115
3.5.3 stack (栈)	60	4.5.4 创建	116
训练 数字游戏	60	4.5.5 删除	117
3.5.4 queue (队列)	61	训练 1 落叶	122
训练 骑士移动	61	训练 2 完全二叉搜索树	124
3.5.5 list (双向链表)	63		
训练 新兵队列训练	64		
第 4 章 树的应用	66	第 5 章 图论基础	127
4.1 树	66	5.1 图的存储	128
4.1.1 树的存储	68	5.1.1 邻接矩阵	128
4.1.2 树、森林与二叉树的 转换	71	5.1.2 边集数组	129
4.2 二叉树	73	5.1.3 邻接表	130
4.2.1 二叉树的性质	74	5.1.4 链式前向星	133
4.2.2 满二叉树和完全二 叉树	75	5.1.5 图的存储技巧	136
4.2.3 二叉树的存储结构	78	5.2 图的遍历	136
4.3 二叉树遍历	80	5.2.1 广度优先遍历	136
4.3.1 先序遍历	80	5.2.2 深度优先遍历	140
4.3.2 中序遍历	83	训练 1 最大的节点	144
4.3.3 后序遍历	86	训练 2 油田	145
4.3.4 层次遍历	90		
训练 1 新二叉树	92	第 6 章 算法入门	149
训练 2 二叉树遍历	93	6.1 贪心算法	149
4.4 哈夫曼树	95	6.1.1 贪心算法秘籍	149
4.4.1 哈夫曼编码	95	6.1.2 最优装载问题	150
4.4.2 哈夫曼编码的长度 计算方法	108	训练 1 部分背包问题	152
训练 1 围栏修复	109	训练 2 排队接水	153
训练 2 信息熵	110	训练 3 线段覆盖	154
		6.2 分治算法	156
		6.2.1 分治算法秘籍	156
		6.2.2 合并排序	156
		6.2.3 快速排序	161
		训练 1 排序 (模板)	168
		训练 2 求第 k 小的数	169

第7章 高精度计算 171

7.1 高精度加法	171
7.1.1 接收和存储数据	171
7.1.2 处理进位	171
训练 A+B Problem	174
7.2 高精度减法	175
7.2.1 比较大小	175
7.2.2 接收和存储数据	175
7.2.3 处理借位	175
训练 A-B Problem	177
7.3 高精度乘法	178
7.3.1 接收和存储数据	178
7.3.2 处理进位	178
训练 A*B Problem	179
7.4 高精度除法	180
7.4.1 接收和存储数据	180
7.4.2 按位相除	181
训练 A/B Problem	181

第8章 搜索算法入门 183

8.1 二分算法	183
8.1.1 二分查找	183
8.1.2 二分答案	186
训练 1 查找	187
训练 2 跳石头游戏	189
训练 3 花环	193
8.2 深度优先搜索	195
8.2.1 回溯法的原理	195
8.2.2 回溯法模板	197

训练 1 01 背包问题	198
--------------------	-----

训练 2 图的 m 着色问题	205
------------------------	-----

训练 3 n 皇后问题	213
---------------------	-----

8.3 广度优先搜索	227
------------------	-----

8.3.1 分支限界法的原理	227
----------------------	-----

8.3.2 分支限界法秘籍	227
---------------------	-----

训练 1 迷宫问题	228
-----------------	-----

训练 2 01 背包问题	229
--------------------	-----

第9章 动态规划入门 235

9.1 动态规划秘籍	235
------------------	-----

9.1.1 动态规划的三个要素	236
-----------------------	-----

9.1.2 动态规划的设计方法	236
-----------------------	-----

9.2 背包问题	237
----------------	-----

9.2.1 01 背包问题	238
---------------------	-----

9.2.2 完全背包问题	246
--------------------	-----

训练 1 骨头收藏家	246
------------------	-----

训练 2 存钱罐	248
----------------	-----

9.3 线性动态规划	250
------------------	-----

训练 1 超级楼梯	250
-----------------	-----

训练 2 数字三角形	251
------------------	-----

训练 3 最长上升子序列	253
--------------------	-----

训练 4 最长公共子序列	256
--------------------	-----

训练 5 最大连续子段和	257
--------------------	-----

9.4 区间动态规划	259
------------------	-----

训练 1 回文	259
---------------	-----

训练 2 括号匹配	261
-----------------	-----

训练 3 乘法难题	263
-----------------	-----

训练 4 猴子派对	265
-----------------	-----