

第四章 查找

4.1 练习

Problem 4.1.1

散列表 H 的类型在闭散列方法下是一个表的引用数组，而在开散列方法下是一个关键字的数组。假设一个关键字需要一个单位的内存，而一个连接表节点需要两个单位的内存，其中一个存放关键字，另一个存放表的引用。考虑在闭散列下如下的负载因子：0.25, 0.5, 1.0, 2.0。设 h_C 是闭散列下散列单元的数量。

1. 给出总体的空间需求，包括在闭散列下的表所需要的空间。假设同样的空间用于开散列表，它的负载因子会是多少？
2. 假设一个关键字要占据4个单位的内存，而一个表节点需要5个单位内存（4个用于关键字，一个用于余下表的引用），重复上一问中的问题。

Problem 4.1.2

对下面每个问题给出符合要求的算法，可以在你的算法中使用已有的排序算法。

- (a) S 是由 n 个数组成的数组，并且未排序。请给出算法用来找到整数对 $x, y \in S$ 并且 $|x - y|$ 最大，最坏情况下，你的算法的复杂度为 $O(n)$ 。
- (b) S 是由 n 个数组成的有序数组。请给出算法用来找到整数对 $x, y \in S$ 并且 $|x - y|$ 最大，最坏情况下，你的算法的复杂度为 $O(1)$ 。
- (c) S 是由 n 个数组成的数组，并且未排序。请给出算法用来找到整数对 $x, y \in S$ 并且 $|x - y|$ ($x \neq y$)最小，最坏情况下，你的算法的复杂度为 $O(n \log n)$ 。
- (d) S 是由 n 个数组成的有序数组。请给出算法用来找到整数对 $x, y \in S$ 并且 $|x - y|$ ($x \neq y$)最小，最坏情况下，你的算法的复杂度为 $O(n)$ 。

Problem 4.1.3

给定由 n 个实数组成的集合 S 和一个实数值 a 。请给出算法判定在 S 中是否存在两个元素的和为 a 。

- (a) 假设 S 是未排序的。给出复杂度为 $O(n \log n)$ 的算法。
- (b) 假设 S 是排完序的。给出复杂度为 $O(n)$ 的算法。

Problem 4.1.4

下面两个问题都是求集合 A, B 的并集，其中 $n = \max\{|A|, |B|\}$ 。

- (a) 假设集合 A, B 中的元素都是乱序的。请给出复杂度为 $O(n \log n)$ 的算法。

(b) 假设集合 A, B 中的元素都是排好序的。请给出复杂度为 $O(n)$ 的算法。

Problem 4.1.5

任意的一棵 RB_h 或者 ARB_h 的 black height 是良定义的, 为 h 。现在另 T 为一棵 RB_h , A 为一棵 ARB_h , 请证明下列命题:

- (a) T 至少有 $2^h - 1$ 个内部黑色节点。
- (b) T 至少有 $4^h - 1$ 个内部节点。
- (c) 任意黑色节点的深度至少是它的 black depth 的两倍。
- (d) A 至少有 $2^h - 2$ 个内部黑色节点。
- (e) A 至少有 $(4^h)/2 - 1$ 个内部节点。
- (f) 任意黑色节点的深度至少是它的 black depth 的两倍。

Problem 4.1.6

证明: 对于一个并查集程序, 长度为 m , 一组中有 n 个元素, 那么它如果使用无权值的合并和直接查找的方法来实现的话, 需要最多 $n + mn$ 次连接操作。

Problem 4.1.7

说明一个大小为 m 的并查集需要 $\Omega(n + (m - n)\log n)$ 的时间来进行查找和合并, 如果查找使用直接查找 (无路径压缩), 合并使用加权值的合并 (wUnion)。

Problem 4.1.8

S_k 树是如下定义的:

- S_0 如果只有一个节点, 它是一棵树。
- 对于 $k > 0$, 一个 S_k 是由两棵不相交的 S_{k-1} 树构成, 通过将其中一棵的根连接到另一棵的根上¹。

证明: 如果 T 是一棵 S_k 树, 则 T 有 2^k 个顶点, 高度为 k , 并且在第 k 层有唯一的顶点, 这个在第 k 层的顶点称为 S_k 树的柄。

Problem 4.1.9

使用练习 4.1.8 中的定义和结果, 证明 S_k 树的以下性质: 设 T 是一棵 S_k 树, 其树柄为 v 。有不相交的树 T_0, T_1, \dots, T_{k-1} , 不包含 v , 它们的根分别是 r_0, r_1, \dots, r_{k-1} , 这样,

1. T_i 是一棵 S_i 树, 其中 $0 \leq i \leq k-1$, 且
2. T 是通过将 v 加到 r_0 , r_i 加到 r_{i-1} 上得到的, 其中 $0 \leq i \leq k-1$

这个分解过程在 S_4 的情形下请参考图 6.29²。

Problem 4.1.10

使用在练习 4.1.8 中的定义和结果, 证明 S_k 树的以下性质: 设 T 是一棵 S_k 树, 其根为 r 而树柄为 v 。有互不相交的树 $T'_0, T'_1, \dots, T'_{k-1}$, 不包含 r , 它们的根分别是 $r'_0, r'_1, \dots, r'_{k-1}$, 这样,

¹请参考 Sara Baase and Allen Van Gelder. Computer Algorithms-Introduction to Design and Analysis (算法设计与分析) 图 6.28

²Sara Baase and Allen Van Gelder. Computer Algorithms-Introduction to Design and Analysis (算法设计与分析) 图 6.29

1. T'_i 是一棵 S_i 树, $0 \leq i \leq k-1$,
2. T 是通过如下方式构造的: 即将每个 r'_i 添加到 r 上, 对于所有的 $0 \leq i \leq k-1$,
3. v 是 T'_{k-1} 的树柄。

这一过程在树为 S_4 的时候请参考图6.30³。

4.2 问题

Problem 4.2.1

采用闭地址散列的方式避免哈希表的冲突问题。现在假设有 k 个关键字需要插入该哈希表中 (有 n 个槽), 每个关键字等可能地插入每个槽中。求恰好有 k 个关键字插入某个特定槽的概率。

Problem 4.2.2

以下是程序自动分析中的一个问题。对于一组变量 x_1, \dots, x_n , 给定一些形如 “ $x_i = x_j$ ” 的等式约束和形如 “ $x_i \neq x_j$ ” 的不等式约束。这些约束是否能同时满足?

例如, 如下一组约束:

$$x_1 = x_2, x_2 = x_3, x_3 = x_4, x_1 \neq x_4$$

是无法同时满足的。请给出一个有效的算法, 判断关于 n 个变量的 m 个约束是否可同时满足。

Problem 4.2.3

给定一个由 n 个数组成的集合 S 和一个整数 T , 请给出复杂度为 $O(n^{k-1} \log n)$ 的算法用来判定在 S 中是否存在 k 个整数的和为 T 。

Problem 4.2.4

请给出高效的算法在 n 个值之中找到第三大的, 并回答最坏情况下所需要的比较次数是多少和你所给出的算法在执行过程中是否必须找到最大的和次大的元素。

Problem 4.2.5

如果数组中的第1个元素到第 i 个元素值是单调递增的, 而从第 i 个元素到最后一个元素值是单调递减的, 我们称其为 “单峰” 数组。现在给定一个大小为 n 的 “单峰” 数组 E , 请设计复杂度为 $O(\log n)$ 的找出 E 中最大元素的算法。

Problem 4.2.6

数组 $A[1..n]$ 存放的是 n 个有序的不相同的正整数。请设计算法找出 A 中缺少的最小的正整数。例如, $A = [1, 2, 4, 5]$, 缺少的最小的正整数是3。

Problem 4.2.7

假设有一个数组 $A[1..n]$, 它满足以下两个条件: $A[1] \geq A[2]$; $A[n-1] \leq A[n]$ 。当元素 $A[x]$ 并不大于它的两个邻居 ($A[x-1] \geq A[x], A[x+1] \geq A[x]$) 时, 我们称 $A[x]$ 为局部最小。例如, 在下面的数组中有六个局部最小元素。

- (a) 我们可以用 $O(n)$ 的时间通过扫描一遍数组找到一个局部最小元素。请设计一个用 $O(\log n)$ 的时间找到一个局部最小元素的算法。

³Sara Baase and Allen Van Gelder. Computer Algorithms-Introduction to Design and Analysis(算法设计与分析) 图6.30

9	7	7	2	1	3	7	5	4	7	3	3	4	8	6	9
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(b) 在给定的边界条件下，请说明至少存在一个局部最小元素。

Problem 4.2.8

给定一个有 n 个互不相同的有序整数 $\{a_1, a_2, \dots, a_n\}$ 的序列。请设计算法判断是否存在某个下标 i 满足 $a_i = i$ 。例如， $\{-10, -3, 3, 5, 7\}$ ，有 $a_3 = 3$ ；在 $\{2, 3, 4, 5, 6, 7\}$ 不存在满足条件的 i 。

Problem 4.2.9

假设你只可以使用加法和移位操作。请给出一个高效的算法计算 $\lceil \sqrt{N} \rceil$ 。

1. 给定一个自然数 N 有 n 个比特，请给出复杂度为 $O(n^2)$ 的算法计算 $\lceil \sqrt{N} \rceil$ 。
2. (选做) 请将你的算法改进到 $o(n^2)$ 。

Problem 4.2.10

数组 A 和 B 分别有 n 个整数。给定一个整数 x ，请设计算法找到是否存在 $a \in A$ 和 $b \in B$ 并且满足 $x = a + b$ ，并分析算法的复杂度。

Problem 4.2.11

编写一个客户端用来创建一个符号表完成对字母表示的成绩和数值学分之间的映射，如下表所示，然后从标准输入一个字母成绩的列表，算出并输出对应的GPA（平均成绩）。

A+	A	A-	B+	B	B-	C+	C	C-	D	F
4.33	4.00	3.67	3.33	3.00	2.67	2.33	2.00	1.67	1.00	0.00

Problem 4.2.12

现在有一个由数组组成的集合，数组 i 的大小是 $2i$ 。每个数组不是空（没有元素）就是满的（填满元素）。例如，11个元素存储在下面的数组里：

$A_0 : [a_1]$
 $A_1 : [a_2, a_3]$
 $A_2 : \text{empty}$
 $A_3 : [a_4, a_5, \dots, a_{11}]$

现在插入一个新的元素，称作 a_{12} 。首先创建一个新的大小为1的数组存放 a_{12} ，现在我们查看 A_0 是否为空，如果 A_0 为空，那么就另这个新数组成为 A_0 ；如果不为空（如上面的例子），就将这个新数组和 A_0 合并为一个新的数组（在上面的例子中 A_0 就变为 $[a_1, a_{12}]$ ），并且再继续查看 A_1 是否为空，如果 A_1 为空，就另这个新的 A_0 成为 A_1 ；如果不为空，则将其和 A_1 合并，并且继续查看 A_2 ，反复执行上述操作。所以，在上面的例子中插入 a_{12} ，我们最终得到的新的数组的集合是：

$A_0 : \text{empty}$
 $A_1 : \text{empty}$
 $A_2 : [a_1, a_2, a_3, a_{12}]$
 $A_3 : [a_4, a_5, \dots, a_{11}]$

现在我们另创建一个新的大小为1的数组的开销是1，合并两个大小分别为 m 的数组的开销为 $2m$ ，所以上面的例子所需要的总开销是 $1+2+4$ 。请用平摊分析的方法分析插入操作的复杂度。