

问题求解（二）作业（第二周）

161180162 许致明

2018 年 3 月 14 日

DH 第六章

6.1

(a) 将此过程改为如下形式:

```
SALARY-COMPUTATION( $N, BT[l..N]$ )
1  for  $I = 1$  to  $N$ 
2      if  $BT[I] \leq M$ 
3           $tmp = BT[I] \times Rl$ 
4           $AT[I] = BT[I] - tmp$ 
5           $Tl = Tl + tmp$ 
6      else
7           $tmp = BT[I] \times Rh$ 
8           $AT[I] = BT[I] - tmp$ 
9           $Th = Th + tmp$ 
```

共减少了 N 次比较和 $2N$ 次乘法, 时间复杂度大约下降了一半。

(b) 在这种情况下, 可以只使用一个数组 $B[N]$ 来储存税后的工资信息, 这种做法在 N 较大时, 可以使空间复杂度下降到原来的一半。

6.8

(a) 此算法进行了一次循环, 即将指针从需搜索的文本头移动到文本尾, 因此为线性复杂度。

(b) 此算法在进行旋转时, 至多贴合多边形的每一条边一次, 因此为线性复杂度。

6.10

设所有节点的个数为 n :

4.2(a)(b)(c) 中最坏情况的复杂度均为 $O(n)$, 即需要访问所有节点以获得所需信息

4.3(a) 对于一个合理的较小的常数 K , 打印第 K 层的节点数字和, 则复杂度为 $O(\log_2 N)$ 。4.3(b) 最坏情况下需要遍历整棵树, 故复杂度为 $O(n)$ 。

6.13

证明. 基于比较的排序可以被抽象为决策树, 它是一颗满二叉树, 表示排序算法作用于给定输入的所有比较。考虑一棵高度为 h , 具有 l 个可达叶节点的决策树, 它对应于 n 个元素所做的比较排序。因为 n 个元素的输入共有 $n!$ 种排序, 每一种都作为一个叶子出现在树中, 所以有 $n! \leq l$ 。又因为在一棵高度为 h 的二叉树中, 叶子的数目不多于 2^h , 则有:

$$n! \leq l \leq 2^h$$

取对数有:

$$\begin{aligned} h &\geq \log_2(n!) \\ &= O(n \log_2 n) \end{aligned}$$

因此, 最少的比较次数 (对应于树的高度) 为 $O(n \log_2 n)$ \square

6.18

LG1(m, n)// 一种比较 naive 的方法

```
1  ret = m
2  while ( $n > ret$ )
3      ret = ret  $\times$  m
4  return ret
```