

数据结构中各种排序算法的比较与分析

项丽萍

(晋城职业技术学院,山西 晋城 048000)

[摘要] 由于排序方法的广泛使用,使得研究排序算法成为计算机处理数据的重要内容。同时在教学过程中发现许多学生对于各种排序算法的理解与掌握并不尽如人意,所以本文在分析了各种排序算法的基础上,从算法评价的角度对各种算法做出了效率评价。

[关键词] 排序 算法 时间复杂度 性能评价

[中图分类号] TP274

[文献标识码] A

[文章编号] 1673-0046(2008)12-0154-02

排序在实际中应用很广,据统计,计算机处理的25%的机时是用于排序的。因此,研究高效率的排序方法是数据结构的一个重要内容。而且在日常的教学过程中,笔者发现许多学生对于各种各样的排序算法的掌握不尽如人意,常常张冠李戴,且对于各种排序算法的效率也是模棱两可。鉴于此种情况,笔者将在计算机专业课本中可能出现的各种排序算法及其效率作了小结,供爱好算法研究的专业教师及学生作一参考。

一、排序的基本概念

排序的确切定义为:假设含 n 个记录的序列为 $\{R_1, R_2, \dots, R_n\}$ 其相应的关键字序列为 $\{K_1, K_2, \dots, K_n\}$ 这些关键字相互之间可以进行比较,即在它们之间存在着这样一个关系:

$$K_{p1} \leq K_{p2} \leq \dots \leq K_{pn}$$

按此固有关系将上式记录序列重新排列为 $\{R_{p1}, R_{p2}, \dots, R_{pn}\}$ 的操作称作排序。

根据排序所涉及的存储器,可将排序分为内部排

序和外部排序两大类。在排序过程中,所有记录都放在内存中处理的称为内部排序;当待排序的记录很多,排序时只能有部分记录进入内存,其他大量记录仍存于外部存储器的排序方法称为外部排序。内部排序主要包括插入类、交换类、选择类、归并类及基数类等。

二、排序算法分析

1. 插入类排序

插入类排序的基本思想是将无序子序列中的一个或几个记录“插入”到有序序列中,从而增加记录的有序子序列的长度。它主要有直接插入排序、折半插入排序、表插入排序及希尔插入排序。

(1) 直接插入排序。直接插入排序是把记录逐一按其关键字的大小插入到已排好序的记录序列中的适当位置,直到全部插入完为止。这很像打扑克牌,一边抓牌,一边理牌的过程,每抓一张牌就把它插入到适当的位置上去。

(2) 折半插入排序。当直接插入排序进行到某一趟

1. 外墙保温体系面层产生裂缝的原因分析

由于构造设计的原因,外墙内保温构造设计存在缺陷,内保温是将保温系统置于外墙内侧从而使内、外墙体分处于两个温度场,建筑物结构受温度应力的影响而从始至终处于不稳定的状态,使结构寿命缩短。在相同气候条件下,内保温在外界各种作用力的影响与外保温相比非常显著。在冬采暖、夏制冷的建筑中,外墙和屋面受太阳辐射热及室外温度的作用而引起的温度变化幅度较大(昼夜温差可达到 $20 \sim 40^\circ\text{C}$,年温差可达到 $80 \sim 100^\circ\text{C}$),外墙的温度变形速度与变形量远大于保温层;由于保温层的内置使梁板等部位出现热桥,导致局部温差过大产生结露现象。

外墙外保温构造设计的不足:外保温体系由于被置于外墙外侧,直接承受来自自然界的各种因素影响。就太阳辐射及环境温度变化对其影响,置于保温层之上的抗裂保护层只有 $3 \sim 20\text{mm}$,且与抗裂层相比保温层有较大的热阻,在相同的条件下,外保温抗裂防护层温度变化速度比无保温情况下主体结构外墙温度变化速度高 $8 \sim 30$ 倍。因此抗裂保护层的柔韧性和耐候性对外保温体系的抗裂性能起着关键作用。

2. 构造设计上的防治措施

外墙保温体系面层裂缝防止主要思路是,外保温体系优于内保温体系,逐层渐变柔性释放应力,急剧变化的温差产生的温度应力集中发生在外保温的外表面,根据温度应力、变形能量释放的原理,保证保温体

系内各相邻构造层性能、弹性模量变化指标相匹配、逐层渐变,抗裂砂浆应保证一定的柔韧性与变形协调,同时,在抗裂防护层中采用多种纤维传递应力方向,防止发生应力集中的可能;普通水泥砂浆不应作为保温体系的表面找平及保护层材料;无空腔或小空腔构造可提高体系稳定性;采用无空腔或小空腔构造体系能够提高体系的稳定性,提高体系抵御风及重力荷载的能力,风压长期作用于建筑物外保温层及隔热层,且风压随高度正比增大,因保温墙面局部受负压时,空腔内气体膨胀,反之,气体压缩,正负压反复作用造成保温层疲劳就会产生裂缝,无或小空腔构造使外保温体系具有较大的抗风压能力、整体性好、应力传递稳定、安全性好,并能有效地传递外保温面层荷重引起的应力,保持保温系统的整体稳定性;保温体系的保护层的抗裂性能是关键问题必须认真对待;外墙应尽量采用涂料饰面。

参考文献:

- [1] 刘明明,王吉霖. 围护结构隔热性能评价及计算机算法[J]. 建筑节能, 2003(41).
- [2] 杨善勤. 民用建筑节能设计手册[M]. 北京:中国建筑工业出版社, 1997.

时,对于 $R[i].key$ 来讲,前边 $i-1$ 个记录已经有序,此时可改为折半查找,找出 $R[i].key$ 应插的位置,然后插入。

(3) 表插入排序。为了减少在排序过程中进行的“移动”记录的操作,必须改变排序过程中采用的存储结构。利用静态链表进行排序,并在排序完成之后,一次性地调整各个记录相互之间的位置,即将每个记录都调整到它们所应该在的位置上。算法如下:

```
void LInsertionSort (Elem SL[ ], int n) {
    SL[0].key = MAXINT;
    SL[0].next = 1; SL[1].next = 0;
    for ( i = 2; i <= n; ++i )
        for ( j = 0, k = SL[0].next; SL[k].key <=
              SL[i].key; j = k, k = SL[k].next )
            { SL[j].next = i; SL[i].next = k; }
    void Arrange ( Elem SL[ ], int n ) {
        p = SL[0].next;
        for ( i = 1; i < n; ++i ) {
            while ( p < i ) p = SL[p].next;
            q = SL[p].next;
            if ( p != i ) {
                SL[p] ← SL[i];
                SL[i].next = p;
            }
            p = q;
        }
    }
}
```

(4) 希尔排序。希尔排序不是每次一个元素挨一个元素的比较,而是初期选用较大增量比较,使记录跳跃式接近它的排序位置,然后增量缩小,直至为 1,这样记录移动次数大大减少,提高了排序效率。

2. 交换类排序

交换类排序的基本思想是通过“交换”无序序列中的记录从而得到其中关键字最小或最大的记录,并将它加入到有序子序列中,以此方法增加记录的有序子序列的长度。主要包括冒泡排序和快速排序。

(1) 冒泡排序。冒泡排序的基本思想是将被排序的每个记录 $R[i]$ 看作是重量为 $R[i].key$ 的气泡,根据轻上重下的原则,按大小顺序扫描数组,凡扫描到轻气泡,就使其向上“飘浮”。如此反复进行,直到最后任何两个气泡都是轻者在上面,重者在下为止。

(2) 快速排序。快速排序的基本思想是:首先对无序的记录序列进行“一次划分”,之后分别对分割所得两个子序列“递归”进行快速排序。

3. 选择类排序

选择类排序的基本思想是从记录的无序子序列中“选择”关键字最小或最大的记录,并将它加入到有序子序列中,以此方法增加记录的有序子序列的长度。主要包括简单选择排序和树型选择排序和堆排序。

(1) 简单选择排序。对 n 个记录进行简单选择排序,所需进行的关键字间的比较次数为:

$$\sum_{i=1}^{n-1} (n-i) = \frac{n(n-1)}{2}$$

(2) 树型选择排序。树型选择排序的思想是将 n 个要排序的记录看成完全二叉树的叶结点,则该完全二叉树有 $2n-2$ 或 $2n-1$ 个结点。首先,两两进行比较(树中兄弟进行比较,否则轮空,直接进入下一轮),胜出的再两两进行比较,直到产生第一名;接下来,将作为第一名的结点看成最差的,并从该结点开始,沿该结点到根路径上,依次进行各分支结点的比较,胜出的就是第二名。如此继续进行下去,直到所有选手的名次确定。

(3) 堆排序。堆排序利用了大根堆(或小根堆)堆顶记录的关键字最大(或最小)这一特征,使得在当前无序区中选取最大(或最小)关键字的记录变得简单。用大根堆排序的基本思想是:①先将初始文件 $R[1..n]$ 建成一个大根堆,此堆为初始的无序区;②再将关键字最大的记录 $R[1]$ (即堆顶)和无序区的最后一个记录 R

$[n]$ 交换,由此得到新的无序区 $R[1..n-1]$ 和有序区 $R[n]$,且满足 $R[1..n-1].keys \leq R[n].key$;③由于交换后新的根 $R[1]$ 可能违反堆性质,故应将当前无序区 $R[1..n-1]$ 调整为堆。然后再次将 $R[1..n-1]$ 中关键字最大的记录 $R[1]$ 和该区间的最后一个记录 $R[n-1]$ 交换,由此得到新的无序区 $R[1..n-2]$ 和有序区 $R[n-1..n]$,且仍满足关系 $R[1..n-2].keys \leq R[n-1..n].keys$,同样要将 $R[1..n-2]$ 调整为堆。直到无序区只有一个元素为止。

4. 归并类排序

归并类排序的基本思想是通过“归并”两个或两个以上的记录有序子序列,逐步增加记录有序序列的长度。常用的是二路归并算法。

5. 基数排序

基数排序是一种借助于多关键字排序的思想,是将单关键字按基数分成“多关键字”进行排序的方法。分为多关键字排序和链式基数排序。

(1) 多关键字排序。多关键字排序按照从最主位关键字到最次位关键字或从最次位到最主位关键字的顺序逐次排序,分两种方法:最高位优先法,简称 MSD 法和最低位优先法,简称 LSD 法。MSD 是先按 k_1 排序分组,同一组记录中,主关键字 k_1 相等,再对每组按 k_2 排序分成子组,之后,对后面的关键字继续这样的排序分组,直到按最次位关键字 k_d 对各子组排序后。再将各组连接起来,便得到一个有序序列。LSD 是先从 k_d 开始排序,再对 k_d-1 进行排序,依次重复,直到对 k_1 排序后便得到一个有序序列。

(2) 链式基数排序。链式基数排序的基本思想是对于数字型或字符型的单关键字,可以看成是由多个数位或多个字符构成的多关键字,此时可以采用这种“分配——收集”的办法进行排序,称作基数排序法。

三、各种排序方法的综合比较

1. 时间性能分析

(1) 快速排序、堆排序和归并排序的时间复杂度为 $O(n \log n)$;直接插入排序、折半折半插入排序、起泡排序和简单选择排序的时间复杂度为 $O(n^2)$;基数排序的时间复杂度为 $O(n)$ 。

(2) 当待排记录序列按关键字顺序有序时,直接插入排序和起泡排序能达到 $O(n)$ 的时间复杂度。

(3) 二路归并算法的时间复杂度为 $O(n \log_2 n)$ 。

(4) 快速排序的时间性能蜕化为 $O(n^2)$,简单选择排序、堆排序和归并排序的时间性能不随记录序列中关键字的分布而改变。

(5) 基数排序的时间复杂度为 $O(d(n+rd))$ 其中:分配为 $O(n)$,收集为 $O(rd)$ (rd 为“基”) d 为“分配——收集”的趟数。

2. 空间性能分析

空间性能指的是排序过程中所需的辅助空间大小。

(1) 所有的简单排序方法(包括:直接插入、起泡和简单选择)和堆排序的空间复杂度为 $O(1)$ (2) 快速排序为 $O(\log n)$,为递归程序执行过程中,栈所需的辅助空间 (3) 归并排序所需辅助空间最多,其空间复杂度为 $O(n)$ (4) 链式基数排序需附设队列首尾指针,则空间复杂度为 $O(rd)$ 。

3. 排序方法的稳定性

快速排序、堆排序和希尔排序是不稳定的排序方法。直接插入排序、折半插入排序、冒泡排序等排序方法是稳定的排序方法。

参考文献:

- [1] 郑宗汉. 算法设计与分析[M]. 北京:清华大学出版社, 2005.
- [2] 严蔚敏. 数据结构[M]. 北京:清华大学出版社, 1995.