数据结构

内部排序（二）

深圳大学电子与信息工程学院 周飞

2 第八章 内部排序

第四节 希尔排序

- 一、希尔排序（思路）
  - 从直接插入排序可以看出，当待排序列为一个正序时，时间复杂度为O(n)
  - 若待排序列基本有序时，排序效率会提高
  - 希尔排序：
    - 先将待排序列分成若干子序列，分别进行插入排序；
    - 待整个序列基本有序时，再对全体记录进行一次直接插入排序。

3 第八章 内部排序

第四节 希尔排序

- 二、希尔排序（算法）
  - 首先，以一个整数gap（小于待排序的记录数）作为间隔，
    将全部记录分为gap个子序列，
    每个子序列中存放的记录再原序列中的距离都是gap；
  - 其次，在每一个子序列中分别施行直接插入排序；
  - 然后逐步缩小gap，例如gap = gap/2，或者gap = gap-1；
  - 重复上述步骤，直到gap=1。

4 第八章 内部排序

第四节 希尔排序

- 三、希尔排序（举例）

已知待排序的一组记录的初始排列为：

21，25，49，25*，16，08

第八章 内部排序

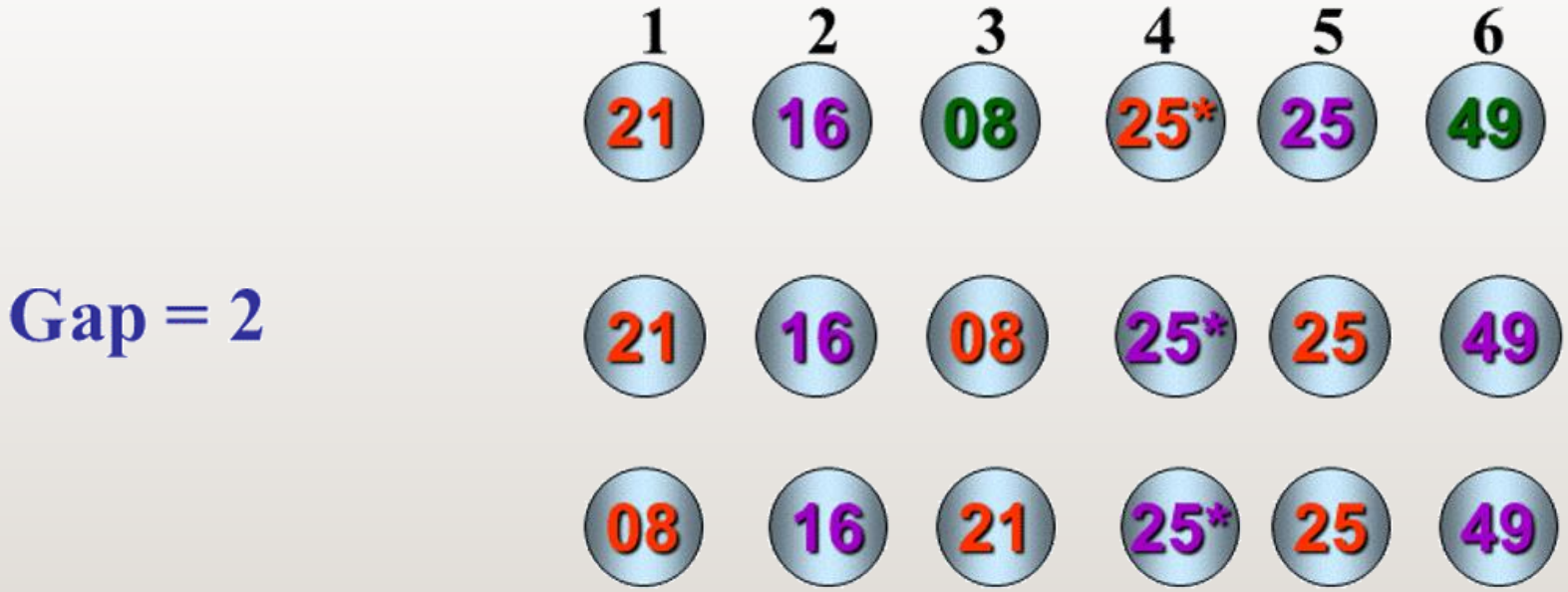
第四节 希尔排序

- 三、希尔排序（举例）

6 第八章 内部排序

第四节 希尔排序

- 三、希尔排序（举例）



第八章 内部排序

第四节 希尔排序

- 三、希尔排序（举例）

Gap = 1

8 第八章 内部排序

第四节 希尔排序

- 四、希尔排序（算法分析）
  - 一般情况下，初始时，序列较乱；

    这时，gap比较大，子序列记录比较少，排序耗时少；

    并且记录一次移动的间隔较大。
  - 随着排序进展，

    gap的值逐渐变小，子序列中的记录个数逐渐变多，

    但是经过前面的排序，已经基本有序，所以排序仍然很快。

9 第八章 内部排序

第四节 希尔排序

- 四、希尔排序（算法分析）
  - gap的取法有多种。
  - Shell提出，

    gap = ⌊n/2⌋

    gap = ⌊gap/2⌋

    直到gap = 1。

第八章 内部排序

第四节 希尔排序

- 四、希尔排序（算法分析）
  - 希尔排序的性能分析是一个复杂的问题；
  - 只对特定待排序的序列，可以准确地估算出关键字地比较次数和记录的移动次数。
  - 一般认为，希尔排序所需的比较次数和移动次数约为$n^{1.3}$
  - 当n趋于无穷时可减小到$n(\log n)^2$
  - 希尔排序的时间复杂度约为$O(n(\log n)^2)$

第八章 内部排序

第四节 希尔排序

- 四、希尔排序（算法分析）
  - 是否稳定？

希尔排序是一种不稳定的排序方法

第八章 内部排序

第五节 简单选择排序

- 一、简单选择排序（思路）
  - 快速排序是给定一个枢轴，通过算法确定它的位置。
  - 能不能给定一个位置，找到应该放在这个位置上的记录？
  - 给定什么位置？
  - 怎么找到这个位置上的记录？
    - 简单选择排序
    - 起泡排序

第八章 内部排序

第五节 简单选择排序

- 一、简单选择排序（思路）
  - 考察序列长度为n的序列
  - 第一趟：第一个位置放置序列中最小的值，
  - 首先，找到最小值：需要进行n-1次比较。
  - 然后，把找到的最小值和第一个位置上存放的记录互换一下。

第八章 内部排序

第五节 简单选择排序

- 二、简单选择排序（算法）

每一趟(例如第i趟, i=1, 2, ..., n-1)在后面
n-i+1个待排序记录中通过n-i次比较，选出
关键字最小的记录, 与第i个记录交换

第八章 内部排序

第五节 简单选择排序

- 二、简单选择排序（举例）

第八章 内部排序

第五节 简单选择排序

- 二、简单选择排序（举例）

第八章 内部排序

第五节 简单选择排序

- 三、简单选择排序（性能分析）
  - 比较次数：
    - 简单选择排序中，关键字的比较次数KCN与记录的初始排列无关。
    - 第*i*趟中，所需的比较次数恒定为n-*i*次。
    - 总的关键字比较次数为

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

第八章 内部排序

第五节 简单选择排序

- 三、简单选择排序（性能分析）
  - 移动次数：
    - 简单选择排序中，记录的移动次数RMN与记录的初始排列有关。
    - 如果是有序排列，RMN=0；
    - 如果每次都要交换，总的移动次数是

$$RMN = 3(n-1)$$

第八章 内部排序

第五节 简单选择排序

- 三、简单选择排序（性能分析）

简单选择排序时间复杂度为$O(n^2)$

- 是否稳定？

简单选择排序是一种不稳定的排序方法

第八章 内部排序

第六节 起泡排序

- 一、起泡排序（思路）
  - 与简单选择排序类似的思路：
    - 先把关键字最大（或最小）的记录安排到最后（或最前）；
    - Repeat…
  - 但是简单选择排序不稳定
  - 是否有其他思路让排序稳定？

第八章 内部排序

第六节 起泡排序

- 一、起泡排序（思路）

依次比较相邻两个记录的关键字，如果发生逆序，则交换之

- i=1时，为第一趟排序，关键字最大的记录将被交换到最后一个位置
- i=2时，为第二趟排序，关键字次大的记录将被交换到最后第二个位置
- ：

第八章 内部排序

第六节 起泡排序

- 二、起泡排序（算法）

第i趟起泡排序从1到n−i+1

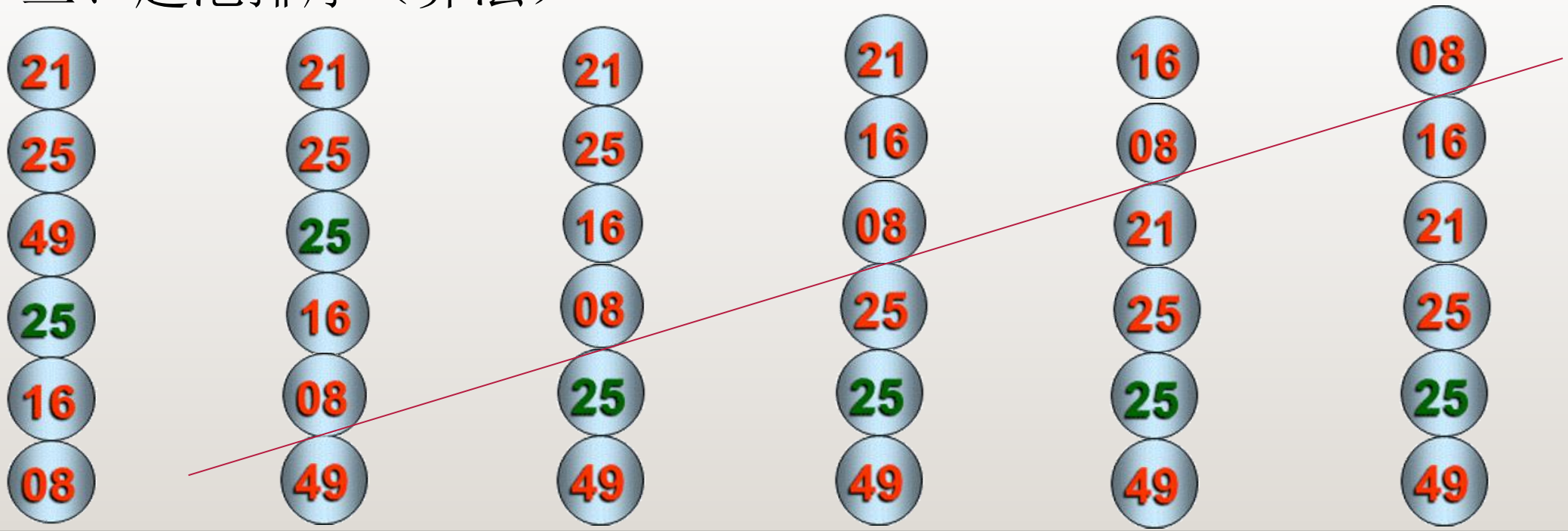
其结果是这n−i+1个记录中，关键字最大的记录被交换到第n−i+1的位置上

最多作n−1趟

第八章 内部排序

第六节 起泡排序

- 三、起泡排序（算法）



第八章 内部排序

第六节 起泡排序

- 四、起泡排序（性能分析）
  - 最好情况：
    - 已经排好序。
    - 第一趟就没有发生交换，因此算法只执行一趟；
    - 没有移动记录；
    - 只进行n-1次关键字的比较。

第八章 内部排序

第六节 起泡排序

- 四、起泡排序（性能分析）
  - 最坏情况：
    - 执行n-1趟起泡；
    - 第i趟做n-i次关键字的比较，执行n-i次记录交换；

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

$$RMN = 3\sum_{i=1}^{n-1}(n-i) = \frac{3}{2}n(n-1)$$

起泡排序是一种稳定的排序方法

第八章 内部排序

第七节 归并排序

- 一、归并（概念和思路）
  - 归并是将两个或两个以上有序表合成一个新的有序表的操作过程。
  - 当考察2个有序表时，被称为2路（或二路）归并。
    - 在二路归并中，假设两个有序表的表长分别为m和L，则归并后表长为m+L
    - 二路归并最多需要多少次移动和比较？
      - m+L次比较和m+L次移位

因此两路归并的时间复杂度为O(m+L)

第八章 内部排序

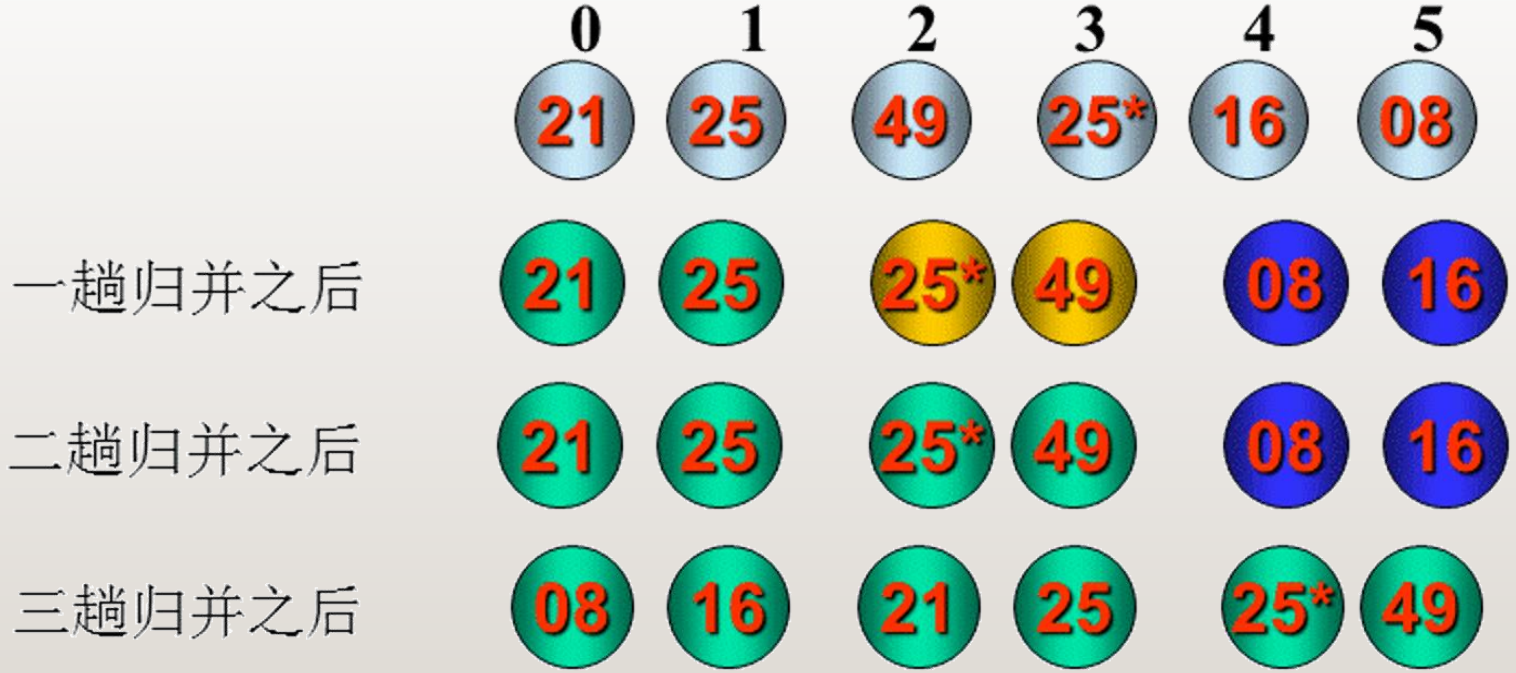
第七节 归并排序

- 二、2路—归并排序（思路和算法）
  - n个记录的某待排序序列看成n有序的序列（每个序列只有一个记录）
  - 将每前后两个有序序列视为一对，归并成一个有序序列（2路归并）；
  - 重复执行上述操作，直到只有一个有序序列为止。

第八章 内部排序

第七节 归并排序

- 二、2路—归并排序（举例）



第八章 内部排序

第七节 归并排序

- 二、2路—归并排序（性能分析）
  - 如果待排序的序列有n个记录，那么需要多少趟2路归并？
    - O($log(n)$)
  - 每一趟2路归并的时间复杂度是？
    - O(n)
  - 2路—归并排序的时间复杂度是？
    - O($nlog(n)$)

第八章 内部排序

第七节 归并排序

- 二、2路—归并排序（性能分析）
  - 稳定吗？

归并排序是一种稳定的排序方法

第八章 内部排序

第八节 各种排序的比较

排序方法	平均时间	最坏情况	辅助存储	稳定排序
插入排序	$O(n^2)$	$O(n^2)$	$O(1)$	稳定
希尔排序	$O(n(\log_2 n)^2)$	$O(n^2)$	$O(1)$	不稳定
起泡排序	$O(n^2)$	$O(n^2)$	$O(1)$	稳定
快速排序	$O(n\log_2 n)$	$O(n^2)$	$O(\log_2 n)$	不稳定
简单选择排序	$O(n^2)$	$O(n^2)$	$O(1)$	不稳定
归并排序	$O(n\log_2 n)$	$O(n\log_2 n)$	$O(n)$	稳定

在平均情况下，排序速度最快