



软件设计师

--数据结构基础

高级项目经理 任铄

QQ: 1530841586

第一章 数据结构基础

- 1.1 线性表
- 1.2 树和二叉树
- 1.3 图
- 1.4 排序
- 1.5 查找

四、交换排序

交换排序的基本思想是两两比较待排序记录的关键码，如果发生逆序（排列顺序与排序后的次序相反），则交换之，直到所有记录都排好序为止。

交换排序的主要算法有：

- 1) 冒泡排序
- 2) 快速排序

高级项目经理 任铄

QQ: 1530841586

(1) 冒泡排序

基本思路：每趟不断将记录两两比较，并按“前小后大”（或“前大后小”）规则交换。

- 第一趟：第1个与第2个比较，大则交换；第2个与第3个比较，大则交换，……关键字最大的记录交换到最后一个位置上；
- 第二趟：对前 $n-1$ 个记录进行同样的操作，关键字次大的记录交换到第 $n-1$ 个位置上；
- 依次类推，则完成排序

高级项目经理 任铎

QQ: 1530841586

25	25	25	25	11	11	11
56	49	49	11	25	25	25
49	56	11	49	41	36	36
78	11	56	41	36	41	
11	65	41	36	49		
65	41	36	56			
41	36	65				
36	78					

初
始
关
键
字
 第
一
趟
排
序
后
 第
二
趟
排
序
后
 第
三
趟
排
序
后
 第
四
趟
排
序
后
 第
五
趟
排
序
后
 第
六
趟
排
序
后

高级项目经理 任铄
 QQ: 1530841586

向上人生路!

最好情况：初始排列已经有序，只执行一趟起泡，做 $n-1$ 次关键码比较，不移动对象。

最坏情形：初始排列逆序，要执行 $n-1$ 趟起泡。

- 时间效率： $O(n^2)$ 因为要考虑最坏情况
- 空间效率： $O(1)$ 只在交换时用到一个缓冲单元
- 稳定性：稳定，因为25和25*在排序前后的次序未改变

高级项目经理 任铎

QQ: 1530841586

向上人生路!

(2) 快速排序

从待排序列中任取一个元素 (例如取第一个) 作为中心, 所有比它小 (或相等) 的元素一律放前, 所有比它大的元素一律放后, 形成左右两个子表;

然后再对各子表重新选择中心元素并按此规则调整, 直到每个子表的元素只剩一个。此时便为有序序列了。

- 优点: 因为每趟可以确定不止一个元素的位置, 而且呈指数增加, 所以特别快!
- 前提: 顺序存储结构

高级项目经理 任铄

QQ: 1530841586

快速排序过程：

对 $r[0.....n-1]$ 中记录快速排序，设两个指针 i 和 j ，基准记录 $x=r[0]$ ，初始时令 $i=0+1, j=n-1$

- 1、从 j 所指位置向前搜索第一个关键字小于 x 的记录，并和 $r[j]$ 交换
- 2、从 i 所指位置起向后搜索，找到第一个关键字大于 x 的记录，和 $r[i]$ 交换
- 3、重复上述两步，直至 $i=j$ 为止
- 4、分别对两个子序列进行快速排序，直到每个子序列只含有一个记录为止

向上人生路！

例

初始关键字: $\begin{matrix} x \\ \downarrow \\ \downarrow \end{matrix}$ 27 38 13 49 76 97 65 50

$\begin{matrix} \uparrow \\ \uparrow \\ \uparrow \\ \uparrow \\ \uparrow \\ \uparrow \\ \uparrow \\ \uparrow \end{matrix}$

完成一趟排序: (27 38 13) 49 (76 97 65 50)

分别进行快速排序: (13) 27 (38) 49 (50 65) 76 (97)

快速排序结束: 13 27 38 49 50 65 76 97

高级项目经理 任铄

QQ: 1530841586

向上人生路!

若待排记录的初始状态为按关键字有序时，快速排序将蜕化为冒泡排序，最坏时间复杂度为 $O(n^2)$ ，最好时间复杂度为 $O(n\log_2 n)$

快速排序是一种不稳定的排序方法

高级项目经理 任铄

QQ: 1530841586

向上人生路!

五、归并排序

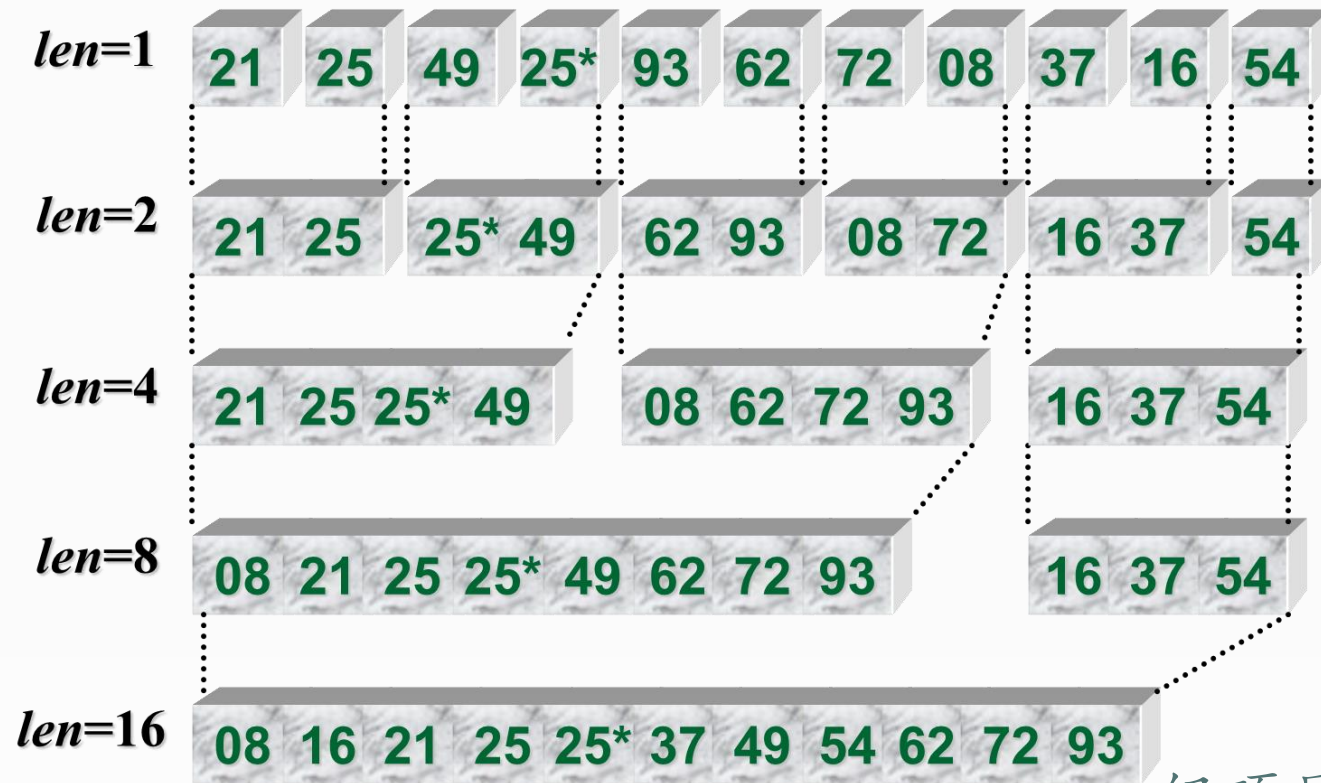
可以把一个长度为 n 的无序序列看成是 n 个长度为1的有序子序列，首先做两两归并，得到 $n/2$ 个长度为2的有序子序列；再做两两归并，...，如此重复，直到最后得到一个长度为 n 的有序序列。

高级项目经理 任铄

QQ: 1530841586

向上人生路!

例：关键字序列T= (21 , 25 , 49 , 25* , 93 , 62 , 72 , 08 , 37 , 16 , 54) ，请给出归并排序的具体实现过程。



整个归并排序仅需 $\lceil \log_2 n \rceil$ 趟

高级项目经理 任铄

QQ: 1530841586

归并排序

- 时间效率： $O(n\log_2 n)$
- 空间效率： $O(n)$
- 稳定性：稳定

高级项目经理 任铄

QQ: 1530841586

高级项目经理 任铄

QQ: 1530841586

算法复杂性比较

表 1-3 排序算法时间复杂度表

排序方法	最好情况	平均时间	最坏情况	辅助空间	稳定性
直接插入排序	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	✓
简单选择排序	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(1)$	✓
冒泡排序	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	✓
快速排序	$O(n\log_2n)$	$O(n\log_2n)$	$O(n^2)$	$O(\log_2n)$	×
堆排序	$O(n\log_2n)$	$O(n\log_2n)$	$O(n\log_2n)$	$O(1)$	×
归并排序	$O(n\log_2n)$	$O(n\log_2n)$	$O(n\log_2n)$	$O(n)$	✓
基数排序	$O(d(n+rd))$	$O(d(n+rd))$	$O(d(n+rd))$	$O(rd)$	✓

向上人生路!

可以通过下列渠道沟通联系：

- 1、QQ:1530841586
- 2、QQ群：164955673

向上人生路！