

10.6 基数排序(Radix Sort)

- 基数排序是采用“分配”与“收集”的办法，用对多关键字进行排序的思想实现对单关键字进行排序的方法。
- 以扑克牌排序为例。每张扑克牌有两个“关键字”：花色和面值。其有序关系为：
 - 花色：♣ < ♦ < ♥ < ♠
 - 面值：2 < 3 < 4 < 5 < 6 < 7 < 8 < 9 < 10 < J < Q < K < A

- 如果我们把所有扑克牌排成以下次序：

♣ 2, ..., ♣ A, ♦ 2, ..., ♦ A, ♥ 2, ..., ♥ A, ♠ 2, ...,
♠ A。

- 这就是多关键字排序。排序后形成的有序序列叫做词典有序序列。
- 对于上例两关键字的排序，可以先按花色排序，之后再按面值排序；也可以先按面值排序，再按花色排序。
- 一般情况下，假定有一个 n 个元素的序列 $\{\text{elem}_0, \text{elem}_1, \dots, \text{elem}_{n-1}\}$ ，且每个元素 elem_i 中含有 d 个关键字：

$$(K_i^1, K_i^2, \dots, K_i^d)$$

- 如果对于序列中任意两个元素 elem_i 和 elem_j ($0 \leq i < j \leq n-1$) 都满足:

$$(K_i^1, K_i^2, \dots, K_i^d) < (K_j^1, K_j^2, \dots, K_j^d)$$

- 则称序列对关键字 (K^1, K^2, \dots, K^d) 有序。其中, K^1 称为最高位关键字, K^d 称为最低位关键字。
- 实现多关键字排序有两种常用的方法:
 - 最高位优先MSD (Most Significant Digit first)
 - 最低位优先LSD (Least Significant Digit first)

■ 最高位优先法通常是一个递归的过程：

- 先根据最高位关键字 K^1 排序, 得到若干元素组, 元素组中各元素都有相同关键字 K^1 。
- 再分别对每组中元素根据关键字 K^2 进行排序, 按 K^2 值的不同, 再分成若干个更小的子组, 每个子组中的元素具有相同的 K^1 和 K^2 值。
- 依此重复, 直到对关键字 K^d 完成排序为止。
- 最后, 把所有子组中的元素依次连接起来, 就得到一个有序的元素序列。

■ 最低位优先法

- 首先依据最低位关键字 K^d 对所有元素进行一趟排序;再依据次低位关键字 K^{d-1} 对上一趟排序的结果再排序,依次重复,直到依据关键字 K^1 最后一趟排序完成,就可以得到一个有序的序列。
 - 这种排序方法对每个关键字进行排序时,不需要再分组,而是整个元素组都参加排序。
- LSD和MSD方法也可把单个关键字 K_i 看作是一个子关键字组: $(K_i^1, K_i^2, \dots, K_i^d)$ 进行排序。

链式基数排序

- 基数排序是典型的LSD排序方法，利用“分配”和“收集”对单关键字进行排序。在这种方法中，把单关键字 K_i 看成是一个 d 元组：
$$(K_i^1, K_i^2, \dots, K_i^d)$$
- 其中的每一个分量 K_i^j ($1 \leq j \leq d$) 也可看成是一个关键字。
- 分量 K_i^j 有 $radix$ 种取值，称 $radix$ 为基数。
- 例如，关键字984可以看成是一个3元组(9, 8, 4)，每一位有 0, 1, ..., 9 等10种取值，基数 $radix = 10$ 。字符串关键字 ‘size’ 可以看成是一个4元组(s,i,z,e)，每一位有 ‘a’, ‘b’, ..., ‘z’ 等26种取值， $radix = 26$ 。

链式基数排序过程

- 首先，针对 d 元组中的每一位分量，把元素序列中的所有元素，按 K_i^j 的取值，先“分配”到 $radix$ 个队列中去。
- 然后再按各队列的顺序，依次把元素从队列中“收集”起来，这样所有元素按取值 K_i^j 排序完成。
- 如果对于所有元素的关键字 $\{Key_1, Key_2, \dots, Key_n\}$ ，依次对各位的分量，让 $j=d, d-1, \dots, 1$ ，分别用“分配”、“收集”的运算逐趟进行排序，在最后一趟“分配”、“收集”完成后，所有元素就按其关键字的值从小到大排好序了。

- 各队列采用链式队列结构，分配到同一队列的关键字用链接指针链接起来。
- 每一队列设置两个队列指针：
 - `int font [radix]` 指示队头
 - `int rar [radix]` 指向队尾
- 基数排序以静态链表作为它们的存储结构。
- 【示例】 设待排序的表有10个记录，其关键字分别为（ 614,738,921,485,637,101,215,530,790,306 ） 。
说明采用基数排序进行排序的过程。

【示例】 设待排序的表有10个记录，其关键字分别为
(614, 738, 921, 485, 637, 101, 215, 530, 790, 306)。说明
采用基数排序进行排序的过程。

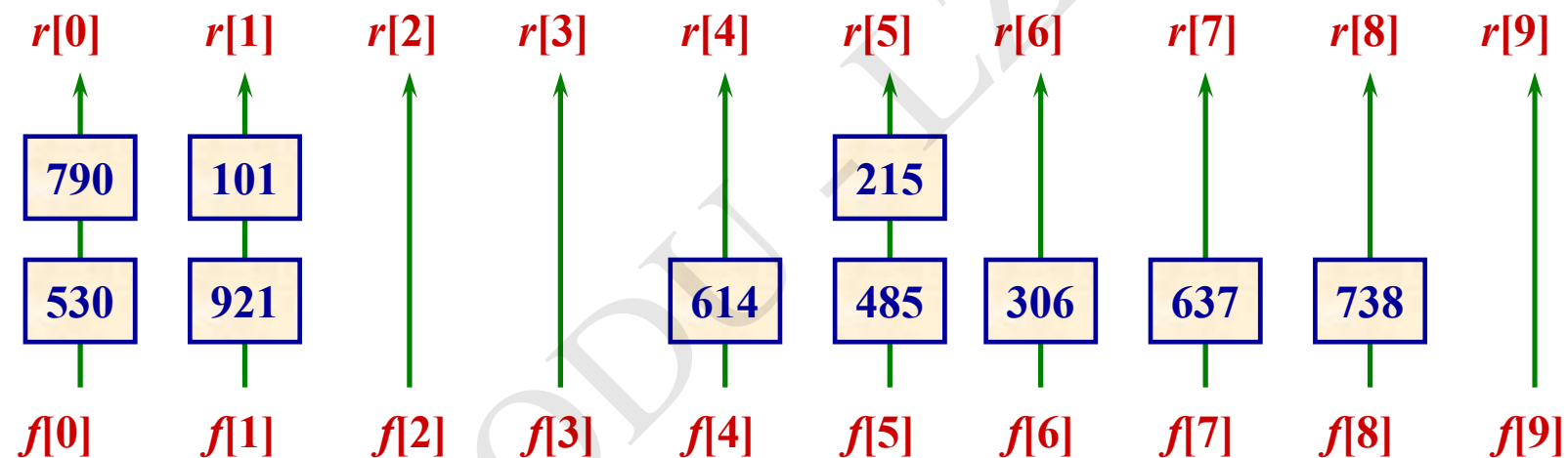
解：

建立10个队列， f 为队头， r 为队尾。

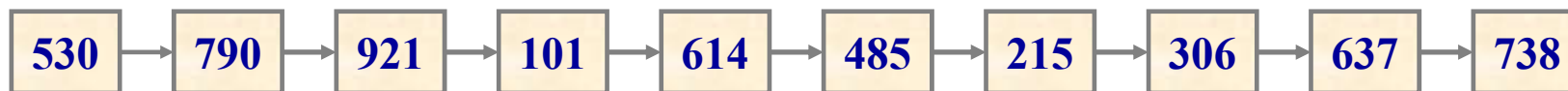
基数排序的“分配”与“收集”过程 第一趟



第一趟分配 (按最低位 $i = 3$)



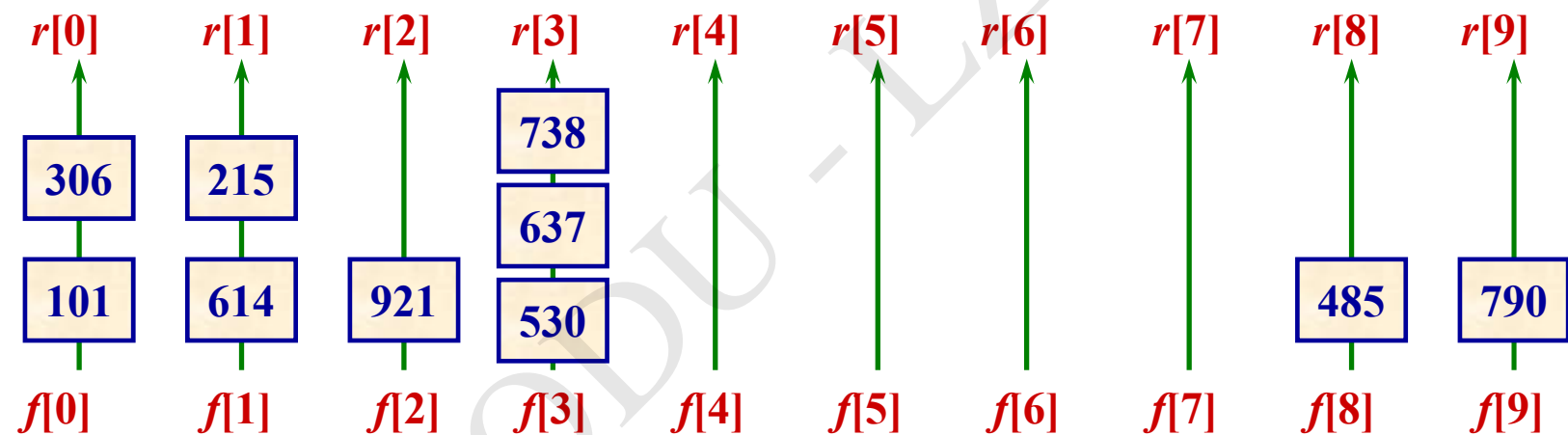
第一趟收集



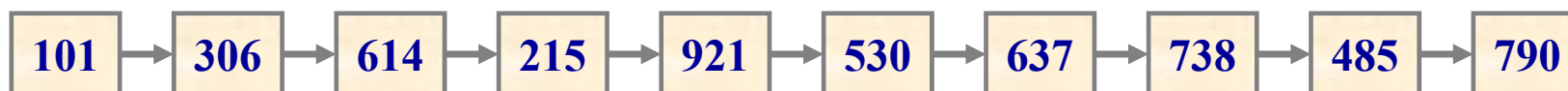
基数排序的“分配”与“收集”过程 第二趟



第二趟分配 (按次低位 $i = 2$)



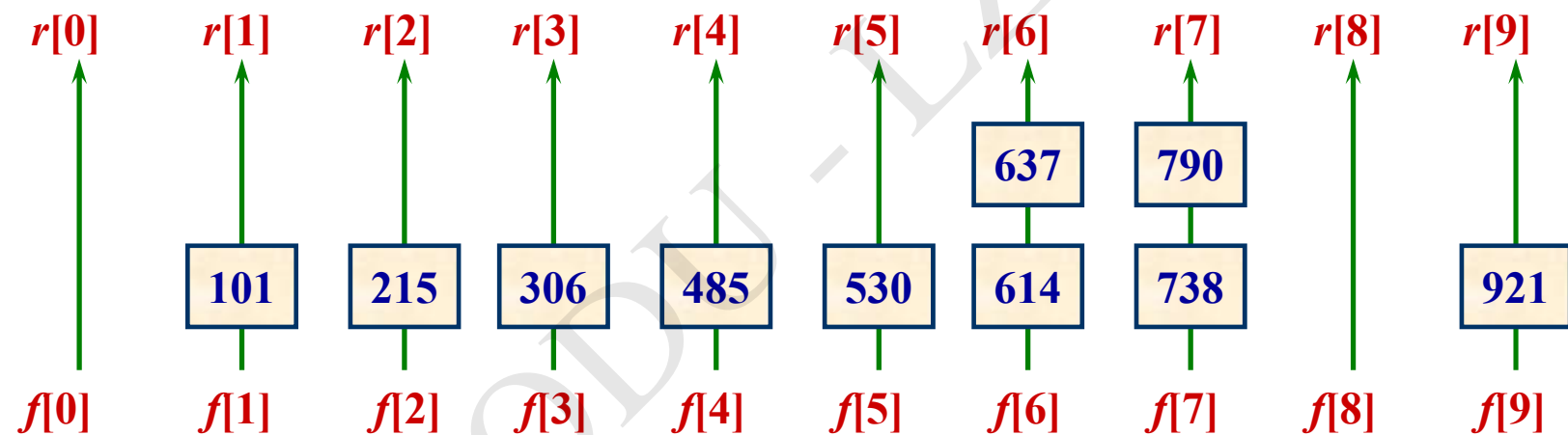
第二趟收集



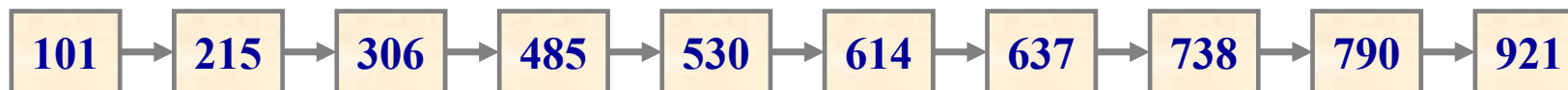
基数排序的“分配”与“收集”过程 第三趟



第三趟分配 (按最高位 $i = 1$)



第三趟收集



- 若每个关键字有 d 位，需重复执行 d 趟“分配”与“收集”。每趟对 n 个元素进行“分配”，对 rd 个队列进行“收集”。总时间复杂度为： $O(d(n+rd))$ 。

归纳起来，基数排序算法的性能如表所示。

时间复杂度			空间复杂度	稳定性
最好情况	最坏情况	平均情况		
$O(d(n+rd))$	$O(d(n+rd))$	$O(d(n+rd))$	$O(rd)$	稳定



— END —