

第十讲 排序（下）

浙江大学 陈 越

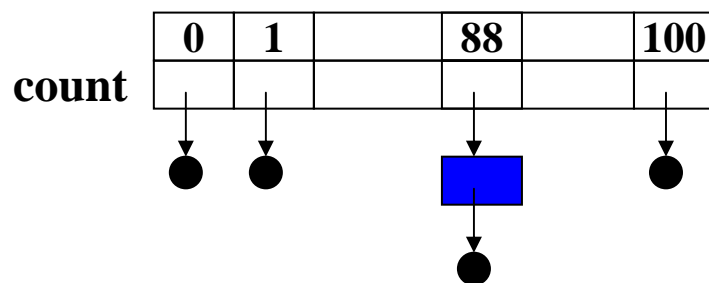
10.3 基数排序

基数排序是稳定的算法

桶排序



假设我们有 N 个学生，他们的成绩是0到100之间的整数（于是有 $M = 101$ 个不同的成绩值）。如何在线性时间内将学生按成绩排序？



如果 $M \gg N$
该怎么办？

```
void Bucket_Sort(ElementType A[], int N)
{
    count[] 初始化;
    while (读入1个学生成绩grade)
        将该生插入count[grade]链表;
    for ( i=0; i<M; i++ ) {
        if ( count[i] )
            输出整个count[i]链表;
    }
}
```

$$T(N, M) = O(M + N)$$

基数排序



假设我们有 $N = 10$ 个整数，每个整数的值在0到999之间（于是有 $M = 1000$ 个不同的值）。还有可能在线性时间内排序吗？

输入序列: 64, 8, 216, 512, 27, 729, 0, 1, 343, 125
用“次位优先” (Least Significant Digit)

设元素个数为N，整数进制为B，LSD的趟数为P，则最坏时间复杂度是：

$$T = O(P(N+B))$$

Bucket	0	1	2	3	4	5	6	7	8	9
Pass 1	0	1	512	343	64	125	216	27	8	729
Pass 2	0	512	125		343		64			
	1	216	27							
	8		729							
Pass 3	0	125	216	343		512		729		
	1									
	8									
	27									
	64									

多关键字的排序



一副扑克牌是按2种关键字排序的

K^0 [花色]

♣ < ♦ < ♥ < ♠

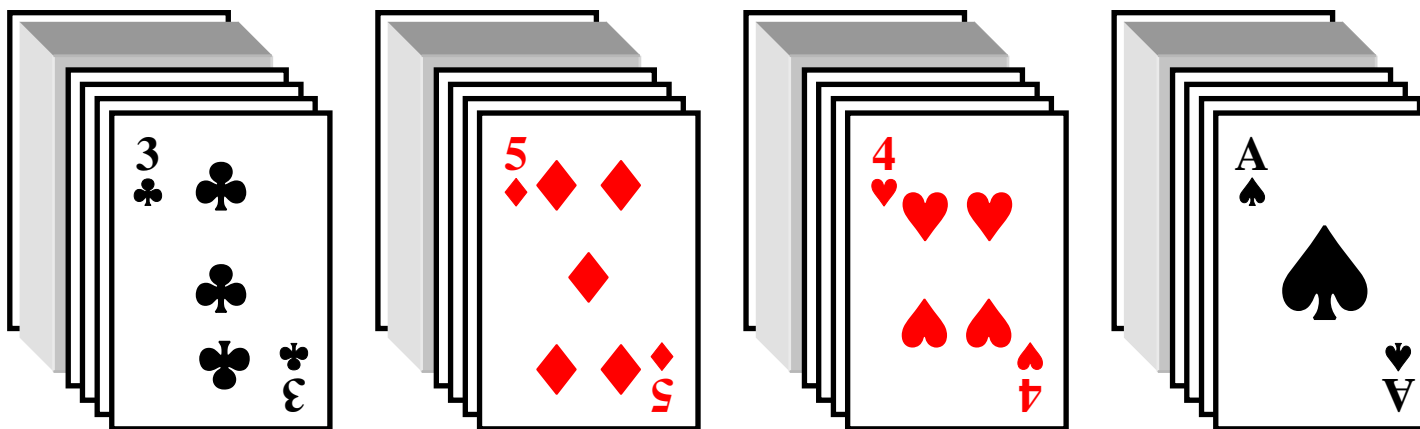
K^1 [面值]

2 < 3 < 4 < 5 < 6 < 7 < 8 < 9 < 10 < J < Q < K < A

有序结果:

2♣ ... A♣ 2♦ ... A♦ 2♥ ... A♥ 2♠ ... A♠

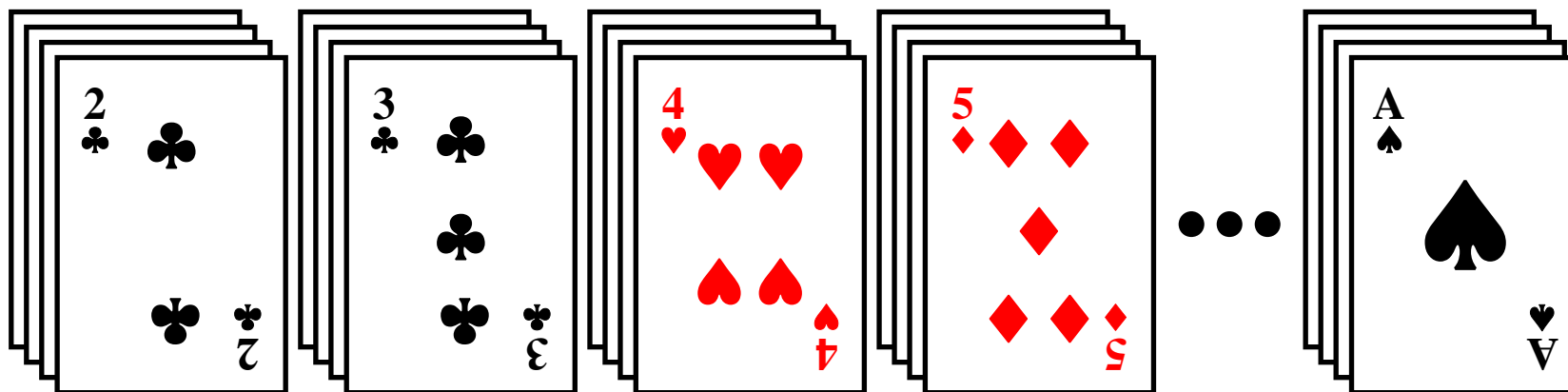
☞ 用“主位优先” (**Most Significant Digit**) 排序: 为花色建4个桶



在每个桶内分别排序, 最后合并结果。

多关键字的排序

- 用“次位优先” (**Least Significant Digit**) 排序：为面值建13个桶



- 将结果合并，然后再为花色建4个桶
- 问题： **LSD** 任何时候都比 **MSD** 快吗？