

4.3 生成组合对象的算法

4.3.1、生成排列

排列问题指的是对于给定的多个元素求其中各种可能的序列。为了简单起见，这里仅仅考虑1到 n 之间的整数的排列问题。

下面介绍三种生成方法：

- (1) 插入法
- (2) Johnson-Trotter 法
- (3) 字典顺序法

插入法生成排列

- 举例：求 $n=3$ 的排列

方法：在 $n=1$ 的排列中插入2
在 $n=2$ 的排列中插入3

过程：

在 1 中从右到左插入2得到 12, 21

在 12 中从右到左插入3得到 123, 132, 312

在 21 中从右到左插入3得到 213, 231, 321

于是得 {123, 132, 312, 213, 231, 321}

Johnson-Trotter 算法

- 有的算法并不需要知道规模 $n-1$ 的排列就可以直接得到规模 n 的排列结果。
- Johnson-Trotter算法就是其中一种。利用这一算法求得的排列序列还是相邻序列变化最小的一个序列集合，也就是说下一个序列与上一个序列仅仅交换了两个元素的位置。

Johnson-Trotter 算法

- 在排列的每一分量上画一个箭头。
- **可移动元素**：如果分量 k 的箭头指向一个相邻的较小元素，则该分量在排列中是可移动的。
- **步骤一**：找到当前最大的可移动整数 k ，移动此元素，掉转所有大于 k 的整数方向。
- 重复步骤一，直到没有可移动的元素为止。
- 例 $n=3$,从123开始：

$\overleftarrow{1}\overleftarrow{2}\overleftarrow{3}$ $\overleftarrow{1}\overleftarrow{3}\overleftarrow{2}$ $\overleftarrow{3}\overleftarrow{1}\overleftarrow{2}$ $\overrightarrow{3}\overleftarrow{2}\overleftarrow{1}$ $\overleftarrow{2}\overrightarrow{3}\overleftarrow{1}$ $\overleftarrow{2}\overleftarrow{1}\overrightarrow{3}$

Johnson-Trotter 算法伪代码

算法 Johnson-Trotter

Initialize the first permutation with $\overleftarrow{1} \overleftarrow{2} \overleftarrow{3} \dots \overleftarrow{N}$

while there exists a mobile integer **do**

$k \leftarrow$ the largest mobile integer

swap (k , the adjacent integer k is looking at)

reverse the direction of all integers larger than k

字典顺序生成排列

- 尽管Johnson-Trotter算法非常高效，但是似乎不是那么直观，不太符合人们的思维习惯。
- 比较自然的算法称为“字典排序算法”（lexicographic order），它是根据单词在字典中的排列顺序得到的算法。

字典生成顺序举例

- 例: $n=3$

在{1, 2, 3}中按字典顺序选择:

123

132

213

231

312

321