

【1】 令A是一个长度为n的正整数序列。试设计一个时间和空间复杂度分别为 $O(n)$ 和 $O(1)$ 的算法，判断A中是否存在这样的元素x，x在序列中出现次数超过 $n/3$ 。若存在这样的x，则将其输出。

【2】 在长度为n的列表A中，元素为互不相同的整型数。若存在这样的数x，它大于它左侧所有数，小于右侧所有数，则称x为A中的一个中间数。例如：若A为：

[3, 1, 6, 4, 5, 7, 9, 8, 10, 14, 12]

则A中有中间数7和10。试设计一个线性时间复杂度的算法，找出给定数组A中的所有中间数。

【3】 给定一个单链表L: $A_0 \rightarrow A_1 \rightarrow \dots \rightarrow A_{n-1}$, 将它重排为: $A_0 \rightarrow A_{n-1} \rightarrow A_1 \rightarrow A_{n-2} \rightarrow A_2 \rightarrow A_{n-3} \rightarrow \dots$ 。要求原地 (in-place) 操作且不改变结点中的内容。例如: 给定 $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$, 重排为 $1 \rightarrow 4 \rightarrow 2 \rightarrow 3$

【4】 现有将栈S, S中元素为整数。试设计算法, 使S中元素自栈顶到栈底元素有序。要求不用任何辅助数据结构, 仅通过对S自身的操作完成S中元素的排序。

【5】 设计算法，在三个有序序列A、B、C中找距离最近的三元组(a, b, c)，即对 $a \in A$ 、 $b \in B$ 、 $c \in C$ ，找出满足 $\min(|a-b|+|a-c|+|b-c|)$ 的三元组(a, b, c)。要求算法的时间复杂度为 $O(|A|+|B|+|C|)$

【6】 设计一个线性时间复杂度的算法，找出无序序列S中具有最大和的子序列。S中元素可正、可负。例如，若序列为：

[1, -2, 3, 10, -4, 7, 2, -5]

则具有最大和的子序列为：[3, 10, -4, 7, 2]，其和为18。

【7】 给定一个 $n \times n$ ($n > 0$) 的矩阵，试编写算法，找到此矩阵的一个其和为最大的子矩阵。例如：

$$\begin{bmatrix} 0 & -2 & -7 & 0 \\ 9 & 2 & -6 & 2 \\ -4 & 1 & -4 & 1 \\ -1 & 8 & 0 & -2 \end{bmatrix}$$

其中左下角的子矩阵：

$$\begin{bmatrix} 9 & 2 \\ -4 & 1 \\ -1 & 8 \end{bmatrix}$$

的和最大，为 $9+2+(-4)+1+(-1)+8=15$ 。
要求算法的时间复杂度能够达到 $O(n^3)$ 。

【8】 现有一整数序列S，用列表表示。试编写一个线性时间复杂度的算法，找出S中最长的对称子序列。
例如，若S为：

[1, 2, 3, 4, 3, 1]

则S中最长对称子序列为[3, 4, 3]，长度3；若S为：

[1, 2, 3, 4, 4, 3, 1]

则最长对称子序列为[3, 4, 4, 3]，长度4。



Copyright All Rights Reserved
版权所有：中国 · 南京 · 东南大学