

排序

- 下列选项中，不可能是快速排序第 2 趟排序结果的是 ()
A. 2, 3, 5, 4, 6, 7, 9 B. 2, 7, 5, 6, 4, 3, 9
C. 3, 2, 5, 4, 7, 6, 9 D. 4, 2, 3, 5, 7, 6, 9
- 对数据序列{8, 9, 10, 4, 5, 6, 20, 1, 2}采用冒泡排序, 要求从后向前进行, 最终按从小到大顺序排列, 需要进行的趟数至少是 ()
A. 3 B. 4 C. 5 D. 8
- 若数据元素序列{11, 12, 13, 7, 8, 9, 23, 4, 5}是采用下列排序方法之一得到的第二趟排序后的结果, 则该排序算法只能是 ()
A. 冒泡排序 B. 插入排序 C. 选择排序 D. 二路归并排序
- 在内部排序过程中, 对尚未确定最终位置的所有元素进行一遍处理称为一趟排序。下列排序方法中, 每一趟排序结束都至少能够确定一个元素最终位置的方法是 ()
I. 简单选择排序 II. 希尔排序 III. 快速排序 IV. 堆排序 V. 二路归并排序
A. 仅 I、III、IV B. 仅 I、III、V C. 仅 II、III、IV D. 仅 III、IV、V
- 对同一序列分别进行折半插入排序和直接插入排序, 两者之间可能的不同之处是 ()
A. 排序的总趟数 B. 元素的移动次数
C. 使用辅助空间的数量 D. 元素之间的比较次数
- 对初始数据序列(8, 3, 9, 11, 2, 1, 4, 7, 5, 10, 6)进行希尔排序。若第一趟排序结果为(1, 3, 7, 5, 2, 6, 4, 9, 11, 10, 8), 第二趟排序结果为(1, 2, 6, 4, 3, 7, 5, 8, 11, 10, 9), 则两趟排序采用的增量(间隔)依次是 ()
A. 3, 1 B. 3, 2 C. 5, 2 D. 5, 3
- 对给定的关键字序列 110, 119, 007, 911, 114, 120, 122 进行基数排序, 则第 2 趟分配收集后得到的关键字序列是 ()
A. 007, 110, 119, 114, 911, 120, 122 B. 007, 110, 119, 114, 911, 122, 120
C. 007, 110, 911, 114, 119, 120, 122 D. 110, 120, 911, 122, 114, 007, 119
- 设数组 $S[] = \{93, 946, 372, 9, 146, 151, 301, 485, 236, 327, 43, 892\}$, 采用最低位优先(LSD)基数排序将 S 排列成升序序列。第 1 趟分配、收集后, 元素 372 之前、之后紧邻的元素分别是 ()
A. 43, 892 B. 236, 301 C. 301, 892 D. 485, 301
- 对数据进行排序时, 若采用直接插入排序而不采用快速排序, 则可能的原因是 ()
I. 大部分元素已有序 II. 待排序元素数量很少
III. 要求空间复杂度为 $O(1)$ IV. 要求排序算法是稳定的
A. 仅 I、II B. 仅 III、IV C. 仅 I、II、IV D. I、II、III、IV
- 已知由 $n(n \geq 2)$ 个正整数构成的集合 A, 将其划分为两个不相交的子集 A1 和 A2, 元素个数分别是 n_1 和 n_2 , A1 和 A2 中元素之和分别为 S1 和 S2。设计一个尽可能高效的划分算法, 满足 $|n_1 - n_2|$ 最小且 $|S_1 - S_2|$ 最大。请给出算法的基本设计思想, 并说明该算法的时间和空间复杂度 (补充要求: 算法复杂度不得高于 $O(n)$)