

算法设计思想作业题目——第二周

必做题

1. 利用至少 3 种方法对存在磁盘上的 2^{31} 个 IPv4 地址进行排序；可用内存只有 1G；至少一种算法使用伪代码表示，其他可以描述排序思想。
2. 如果一对元素 $(A[i], A[j])$ 是倒序的，即 $i < j$ 但是 $A[i] > A[j]$ ，则称它们为一个倒置。设计一个时间复杂度为 $O(n \log n)$ 的算法计算数组中的倒置数量。
3. 写出使用快速排序对序列 E, X, A, M, P, L, E 按照字母表排序的过程。
4. 用非递归的形式实现合并排序和快速排序，写出伪代码。
5. 用主定理求下列递推式的时间复杂度：(1) $T(n) = 4T(n/2) + n$ ；(2) $T(n) = 4T(n/2) + n^3$
6. 设计针对链表的插入排序算法。
7. 设计一个能够找出 n 个元素的数组中距离最近的两个元素间的距离，例如 x 和 y 间的距离为 $|x - y|$ ；对比蛮力法和预排序法之间的效率差异。
8. 假设列表的可能值属于集合 $\{a, b, c, d\}$ ，用分布计数算法将下面的列表按照字母顺序排序：b, c, d, c, b, a, a, c。
9. 假设有 n 个直径各不相同的螺钉和 n 个相应的螺母。我们一次只能比较一对螺钉和螺母来判断螺母和螺钉的关系。然而我们不能拿两个螺母或者两个螺钉比较。我们的问题是要找到每一对匹配的螺钉和螺母。为该问题设计一个算法，它的平均效率要符合 $\Theta(n \log n)$ 。
10. 查找 Shell 排序的定义，完成一个 Shell 排序的伪代码。

选做题

1. 设计一个算法用于找到两个长度都为 n 的排好序的数组中所有元素的上中位数，要求时间复杂度是 $O(\log n)$ ，额外空间复杂度为 $O(1)$
2. 设计一个算法用于找到两个排好序的数组中第 k 大的数，长度分别为 m 和 n ，要求时间复杂度为 $O(\log(\min\{m, n\}))$ ，额外空间复杂度为 $O(1)$
3. 设计一个算法判断长度为 n 的单向链表是否为回文结构，即如果链表长度为偶数，则后半部分节点值为前半部分的倒叙；如果是奇数，则除了中位节点外其他节点满足偶数长度的性质。要求时间复杂度为 $O(n)$ ，额外空间复杂度为 $O(1)$
4. 假设单向链表中每一个节点都是 $0 \sim 9$ 之间的整数，那么链表可以代表一个整数，设计一个算法用于生成两个单向链表对应整数和的单向链表的算法。
5. 设计符合以下要求的算法：(1) 判断一个单向链表是否有环，如果有则返回第一个进入环的节点；(2) 判断两个无环单向链表是否相交，如果相交返回第一个相交的节点；(3) 如何判

断两个有环单向链表是否相交，如果相交返回第一个相交节。

6. 设计一个算法删除单向链表中重复出现的节点（借鉴排序算法的思路）。

7. 设计一种最快的方式删除单链表中当前指针指向的节点（不知道 head 节点在哪里）。

8. 假设这有一个各种字母组成的字符串 A，和另外一个字符串 B，字符串里 B 的字母数相对少一些。什么方法能最快的查出所有小字符串 B 里的字母在大字符串 A 里都有？

比如，如果是下面两个字符串：

String 1: ABCDEFGHLMNOPQRS String 2: DCGSRQPO

答案是 true，所有在 string2 里的字母 string1 也都有。

如果是下面两个字符串：

String 1: ABCDEFGHLMNOPQRS String 2: DCGSRQPZ

答案是 false，因为第二个字符串里的 Z 字母不在第一个字符串里。

9. 一个未排序整数数组，有正负数，重新排列使负数排在正数前面，并且要求不改变原来的正负数之间相对顺序。比如： input: 1, 7, -5, 9, -12, 15 ， ans: -5, -12, 1, 7, 9, 15。设计尽可能多的方案，并体现效率优化的过程。