

Homework 2

1. 下面的排序算法中哪些是稳定的：插入排序、归并排序、堆排序、快速排序和计数排序？给出一个能使任何排序算法都稳定的方法。你所给出的方法带来的额外时间和空间开销是多少？
2. 假设所有元素都是互异的，说明在最坏情况下。如何使快速排序的运行时间为 $O(n \log n)$ 。
3. 给定一个整数数组，其中不同的整数所包含的数字的位数可能不同。但该数组中，所有整数中包含的总数字位数为 n 。设计算法使其可以在 $O(n)$ 时间内对该数组进行排序。
4. SELECT 算法最坏情况下的比较次数 $T(n) = \Theta(n)$ ，但是其中的常数项使非常大的。请对其进行优化，使其满足：
 - 在最坏情况下的比较次数为 $\Theta(n)$ 。
 - 当 i 是小于 $n/2$ 的常数时，最坏情况下只需要进行 $n + O(\log n)$ 次比较。
5. OnlineJudge Problem H2-1 抽奖: <https://202.38.86.171:1443/problem/H2-1>
6. OnlineJudge Problem H2-2 数据排序: <https://202.38.86.171:1443/problem/H2-2>

1. 稳定：插入、归并、计数 排序

不稳定：快速、堆 排序

任意排序都稳定的方法：

在每个元素上加入一个 index 属性，若元素 key 相同则比较 index
只需增加 $O(n)$ 的空间 即可，时间开销不用增加

2. 为了使快排时间为 $O(n \lg n)$ ，需要划分是常数比例，
我们可以用一个 $O(n)$ 的线性选择算法 每次找到中
位数，用中位数进行划分，这样就能保证最坏情况
快排时间复杂度也为 $O(n \lg n)$

3.

① 求数字位数的方法可以 不断将数字除以 10 直到该数等于 0，
相当于遍历每一位，耗时 $O(n)$

② 对位数进行计数排序，耗时 $O(n)$

③ 对相同位数的数进行基数排序，耗时 $O(n)$

故算法总时间复杂度为 $O(n)$

4. ① 对 $\lfloor n/2 \rfloor$ 对不相交元素两两比较，并将较小的元素构成集合 S
若 n 为奇数， S 包含单独的一个元素

② 对 S 中元素求出第 i 大的数 a

③ 找到 S 中小于等于 S 的 i 个数及 ① 中这些数比较的数共 $2i$ 个

④ 求 $2i$ 个数中第 i 大的数

算法证明: S 中大于 a 的数必然比第 i 个数大, 故其比较的元素更不可能, 故只用对小于 a 的 2^i 个数进行 select

最坏比较次数: ①: $\lfloor n/2 \rfloor$
(设比较次数为 $U(n)$) ②: $U(\lfloor \frac{n}{2} \rfloor)$

③④: $T(2i)$

i 为常数时 $U(n) = \lfloor n/2 \rfloor + U(\lfloor \frac{n}{2} \rfloor) + \Theta(1)$

使用代入法, 要证: $U(n) \leq n + d \lg n$

$$\begin{aligned} U(n) &\leq U(\lfloor \frac{n}{2} \rfloor) + \lfloor n/2 \rfloor + c \\ &\leq \lfloor \frac{n}{2} \rfloor + d \lg \lfloor \frac{n}{2} \rfloor + \lfloor \frac{n}{2} \rfloor + c \\ &\leq n + d \lg(\frac{n}{2}) + c \\ &= n + d(\lg n - \lg 2) + c \\ &= n + d \lg n - d \lg 2 + c \end{aligned}$$

只需 $d \geq \frac{c}{\lg 2}$ 即可成立

故 $U(n) = n + O(\lg n)$