

计算机科学导论

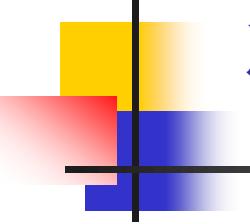
张家琳
中国科学院计算技术研究所

zhangjialin@ict.ac.cn

2021-4-23



算法思维

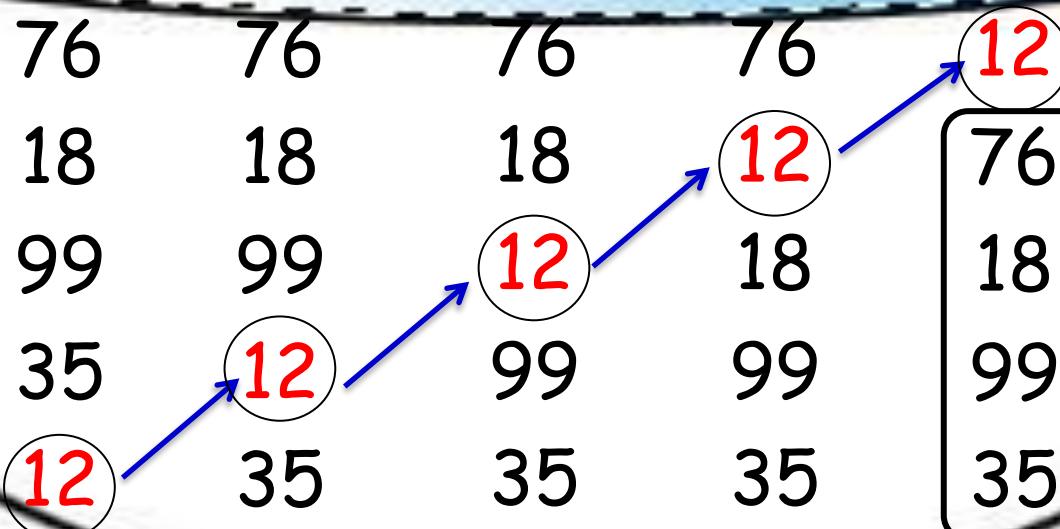


引入：排序

- 任务：给定n个正整数，把他们从小到大排起来
- 请思考在打斗地主、升级、桥牌.....的时候，你是怎么整理牌的？
 - 一张一张摸牌，每次插入已经理好的牌里面（插入排序法）
 - 把最大的牌放到最左边（选择排序法）
 - 先按花色分，再各个花色整理

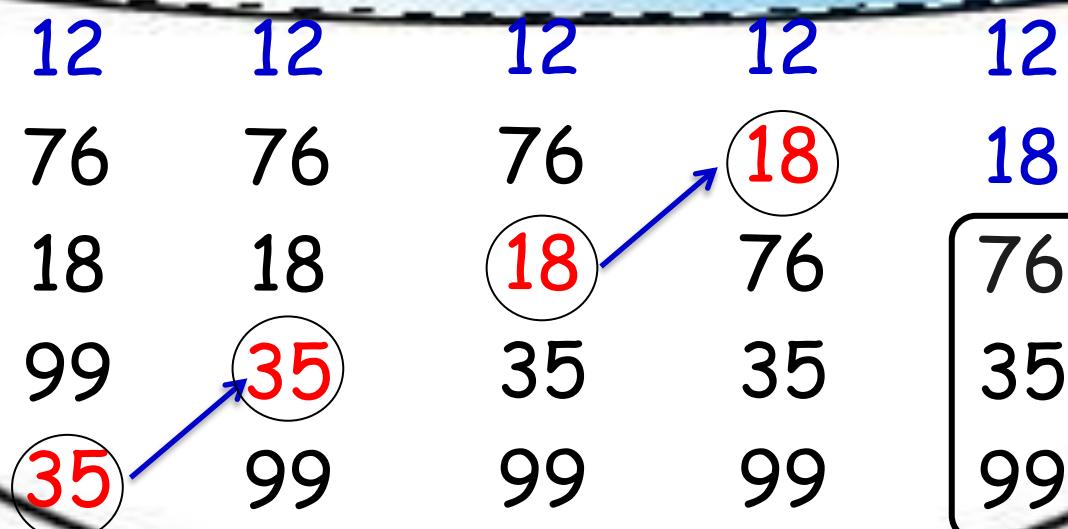
01

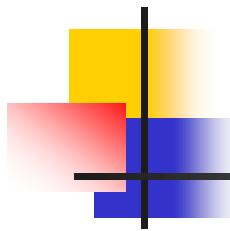
冒泡排序



继续排

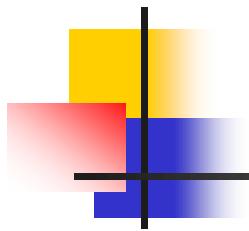
冒泡排序





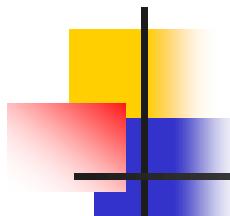
算法

- 有穷性。
- 确切性。
- 输入。
- 输出。
- 可行性。

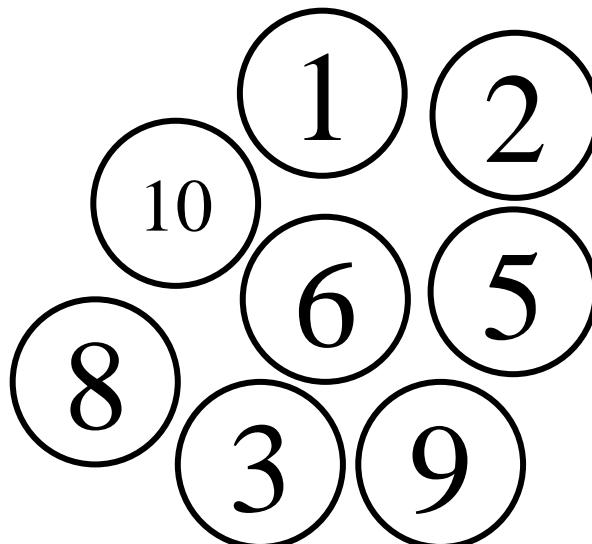


冒泡排序

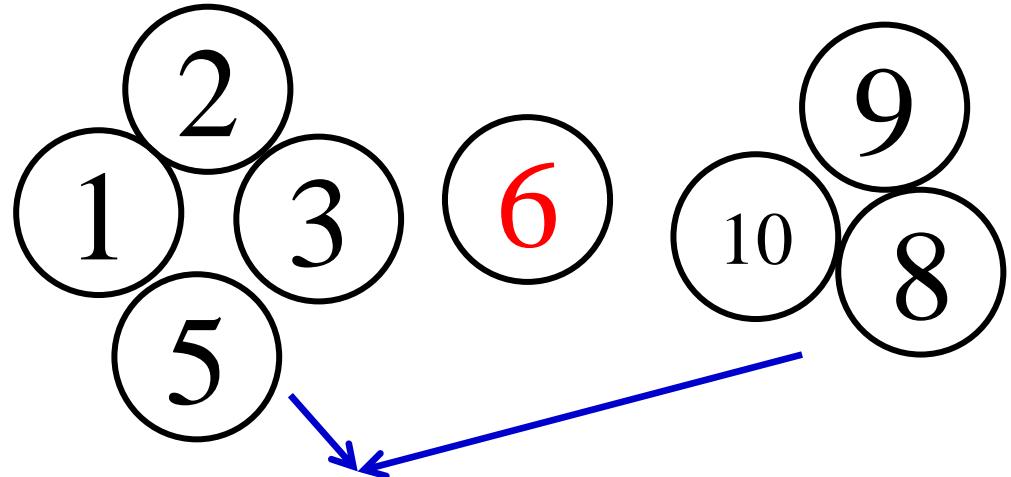
- 需要 $n-1+n-2+\dots+1=n*(n-1)/2$ 次比较操作
- 最坏情况下需要 $n*(n-1)/2$ 次交换操作
 - $n, n-1, \dots, 2, 1$
- 能不能更快？



快速排序



- Step1:** 随机选其中一个数字
Step2: 其他数字按照和基准数的大小关系分成两部分



- Step3:** 分别递归

快速排序

基准数6

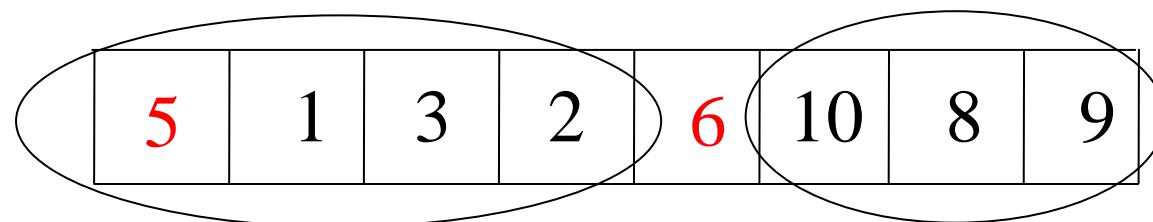
6	1	8	2	5	10	3	9
---	---	---	---	---	----	---	---

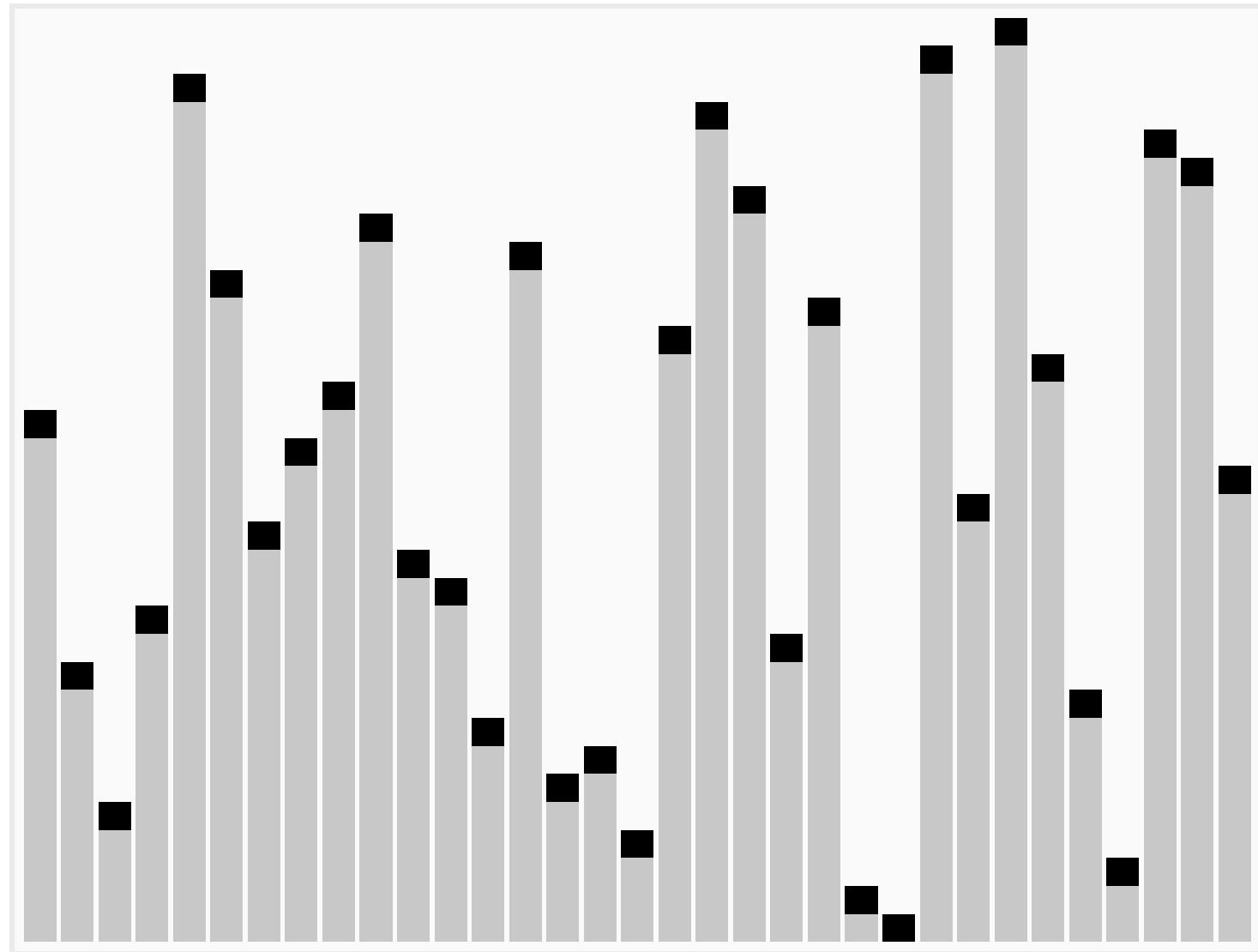
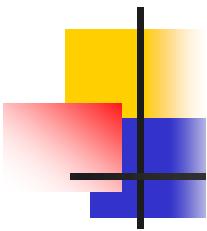


6	1	8	2	5	10	3	9
---	---	---	---	---	----	---	---

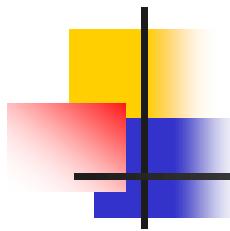


6	1	3	2	5	10	8	9
---	---	---	---	---	----	---	---



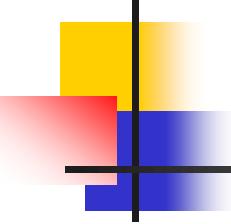


<https://en.wikipedia.org/wiki/Quicksort>



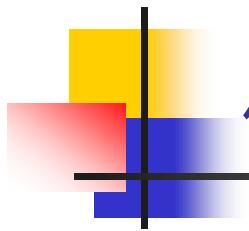
快速排序

- 需要多少次比较?
 - 最坏情况: $n*(n-1)/2$
 - 平均情况: ?



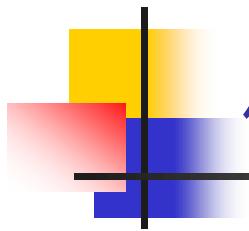
班级快速排序实验

- 用人体计算机实现快速排序算法
- 实验课
- 理论课：
 - 班级排序现场操作
 - 地点：操场OR大草坪OR礼堂
 - 各个班级派代表汇报总结



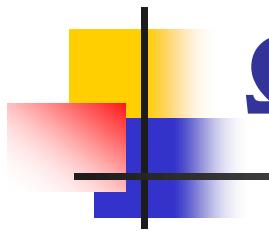
小o，大O记号

- 冒泡排序
 - 运行时间 $O(n^2)$
- 快速排序
 - 平均运行时间 $O(n \log n)$
 - 是 $o(n^2)$



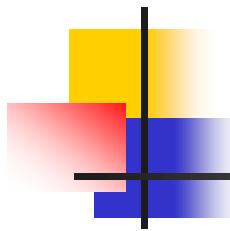
小o, 大O记号

- $f(n) = o(g(n))$: $\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = 0$
 - $n^{1.58} = o(n^2)$
 - $n^{1000} = o(2^n)$
 - $(\log n)^{200} = o(n)$
- $f(n) = O(g(n))$: \exists 常数 $c > 0, f(n) \leq cg(n)$ 对充分大的 n 成立
 - $n^{1.58} = O(n^2) \quad 10n^{1.58} = O(n^{1.58})$
 - $10^{1000} n = O(n)$



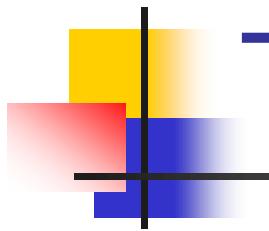
$\Omega(\cdot), \Theta(\cdot)$ 记号

- $f(n) = \Omega(g(n))$: \exists 常数 $c > 0$, $f(n) \geq cg(n)$ 对充分大的n成立
 - $n^2 = \Omega(n^{1.58})$ $10n^{1.58} = \Omega(n^{1.58})$
- $f(n) = \Theta(g(n))$:
 - $f(n) = O(g(n))$ 并且 $f(n) = \Omega(g(n))$
 - $10n^2 - 20n + 45 = \Theta(n^2)$
- 思考: $2^{\Theta(n)}$ 和 $\Theta(2^n)$ 一样吗?



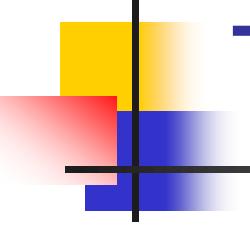
排序问题

- 冒泡排序
- 快速排序
- 还有更多的排序算法
 - 选择排序、插入排序、归并排序、堆排序、二叉搜索树、基数排序.....
 - 外部排序



TimSort介绍-设计思想

- $12,10,7,5,7,10,14,25,36,3,5,7,11,15,22$
- 分成若干单调的部分
- $12,10,7,5,7,10,14,25,36,3,5,7,11,15,22$
- 合并这些部分
 - 怎么合并两个已经排好序的数串？
 - 合并顺序会影响效率
 - TimSort: 给出了一种巧妙的合并顺序
 - 最坏情况 $O(n + n \log \rho)$



TimSort介绍

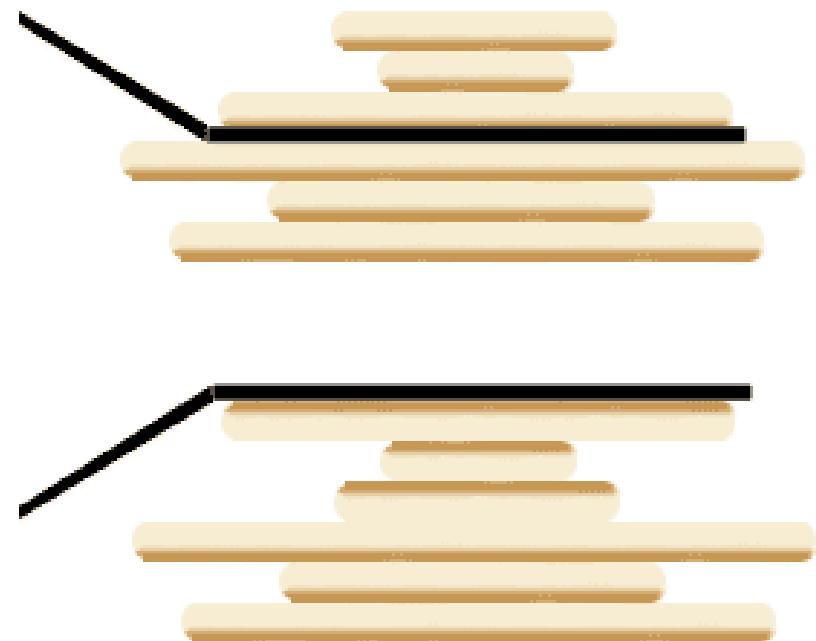
- 快速排序 QuickSort
 - 由Tony Hoare 1959年提出
 - 广泛应用在如C、Java的标准库里面
- TimSort
 - 1993年提出，2002年由Tim Peters实现
 - 应用在Python 2.3, Java SE 7, Android平台.....
 - 最坏情况 $O(n + n \log \rho)$, 2018年证明

思考题



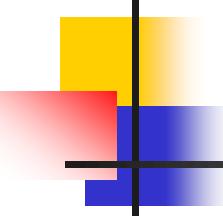
■ 翻煎饼问题(Pancake Sorting)

一个厨师做了一叠大小不同的煎饼，他要不断从上面拿起几个煎饼翻到下面。假设有 n 个煎饼，厨师需要翻动多少次，才能把煎饼按从小到大排好？



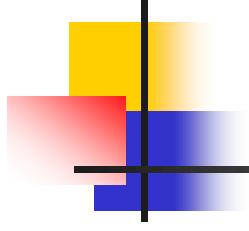
翻煎饼问题

- 2, 1, 4, 5, 3
- 5, 4, 1, 2, 3
- 3, 2, 1, 4, 5
- 1, 2, 3, 4, 5



算法思维

- 排序算法
 - 冒泡排序
 - 快速排序
- 小O, 大O记号
- 思考题:
 - 翻煎饼问题
 - $2^{\Theta(n)}$ VS $\Theta(2^n)$



谢谢！