



福昕高级PDF编辑器

高效 · 安全 · 专业

立即下载

点击购买



OFFICE格式互转



OCR文字识别



文本图像编辑



加密和签署



交互式动态表单



互联PDF文档



中国科学技术大学 计算机科学与技术系
University of Science and Technology of China
DEPARTMENT OF COMPUTER SCIENCE AND TECHNOLOGY

算法设计与分析

Design and Analysis of Algorithms

主讲人 徐云

Fall 2018, USTC



Part 1 Foundation

Part 2 Sorting and Order Statistics

chap 6 Heapsort

chap 7 Quicksort

chap 8 Sorting in Linear Time

chap 9 Medians and Order Statistics

Part 3 Data Structure

Part 4 Advanced Design and Analysis Techniques

Part 5 Advanced Data Structures

Part 6 Graph Algorithms

Part 7 Selected Topics

Part 8 Supplement

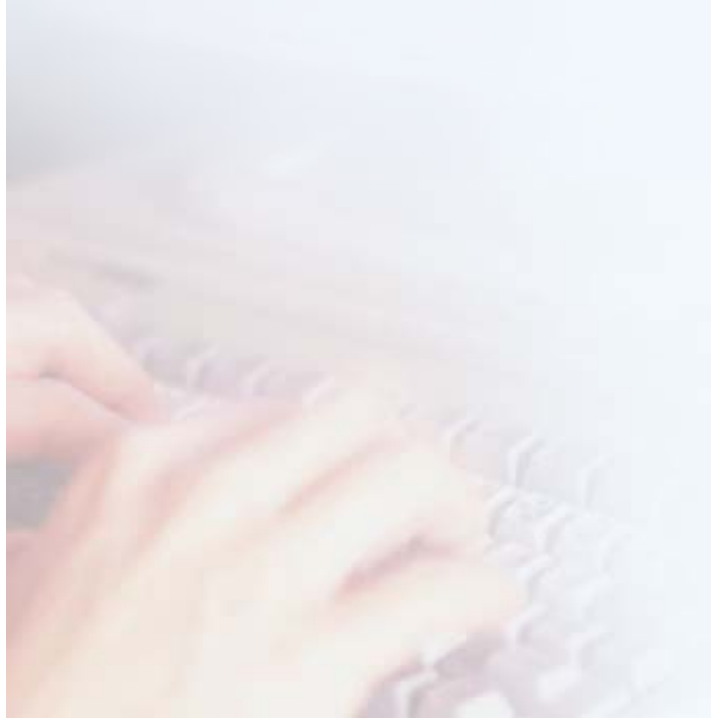


第9章 中值和顺序统计

9.1 最小和最大值

9.2 期望时间为线性的选择

9.3 最坏时间为线性的选择



9.1 最小和最大值

- 最小/最大值：最坏情形 $W(n)=n-1$ 次比较，时间为 $\theta(n)$
- 同时求最大、最小值
 - 一种方法：独立分别求，比较次数为 $n-1+n-2=\underline{2n-3}$
 - 另一种方法：

成对输入 x, y ，每对比较 3 次

 - ① 比较 x, y ;
 - ② 将 $\min(x, y)$ 与当前最小值比较;
 - ③ 将 $\max(x, y)$ 与当前最大值比较;

总比较次数约为 $\underline{3 \lfloor n/2 \rfloor}$ 。// 第一对元素比较一次，最后一组元素若为一个，至多比较二次



第9章 中值和顺序统计

9.1 最小和最大值

9.2 期望时间为线性的选择

9.3 最坏时间为线性的选择



9.2 期望时间为线性的选择

- 基本思想
- RandomizedSelect 算法
- 时间分析

基本思想

- 基于分治法的思想

- 利用快排序的随机划分法，进行问题的划分

- 具体步骤：

- ① 划分 $A[p..r] \Rightarrow A[p..q-1] \leq A[q] < A[q+1..r]$;

- // $A[q]$ 为划分元

- ② $k \leftarrow q - p + 1$; // 即 $A[q]$ 是第 k 个最小元

- ③ if ($i=k$) then // $k =$ 左区间长度+1

- return $A[q]$;

- if ($i < k$) then 在左区间中继续找第 i 个元素;

- if ($i > k$) then 在右区间中继续找第 $i-k$ 个元素;

- 临界条件：当区间长度为1时，直接返回该元素

RandomizedSelect 算法

RandomizedSelect(A, p, r, i)

{ //选择 i^{th} 元素

if $p=r$ then return $A[p]$; //临界问题处理

$q \leftarrow \text{RandomizedPartition}(A, p, r)$;

//进行划分, 并返回划分元的下标

$k \leftarrow q - p + 1$; // $A[q]$ 是第 k 个小的元素

// $A[q]$ 是 i^{th} 元素

if $i=k$ then

return $A[q]$;

else if $i < k$ then // i^{th} 元素落在左区间

return RandomizedSelect($A, p, q-1, i$);

else // i^{th} 元素落在右区间

return RandomizedSelect($A, q+1, r, \underline{i-k}$);

}

时间分析

- 最好：每次划分为相等的左右区间

$$T(n) = T(n/2) + n \Rightarrow T(n) = \Theta(n)$$

- 最坏：每次划分为不均等的左右区间

$$T(n) = T(n-1) + n \Rightarrow T(n) = \Theta(n^2)$$

- 平均(期望)：分析略。

$$T(n) = \Theta(n)$$

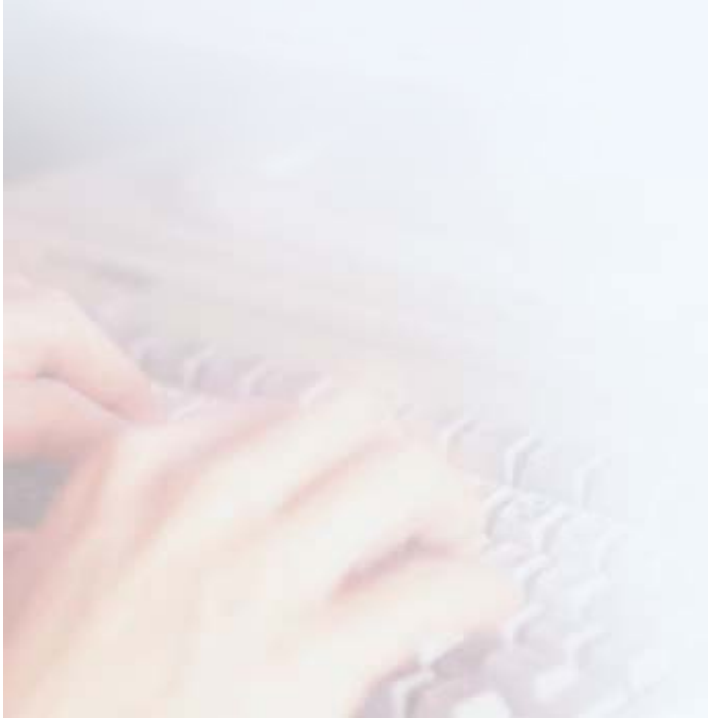


第9章 中值和顺序统计

9.1 最小和最大值

9.2 期望时间为线性的选择

9.3 最坏时间为线性的选择



9.3 最坏时间为线性的选择

- 算法步骤
- 时间分析

算法步骤

While $n > 1$ do

$O(n)$ step 1. 将 n 个元素分成5个1组，共 $\lceil n/5 \rceil$ 组。其中最后1组有 $n \bmod 5$ 个元素。

$O(n)$ step 2. 用插入排序对每组排序，取其中值。若最后1组有偶数个元素，取较小的中值。

$T(\lceil n/5 \rceil)$ step 3. 递归地使用本算法找 $\lceil n/5 \rceil$ 个中值的中值 x 。

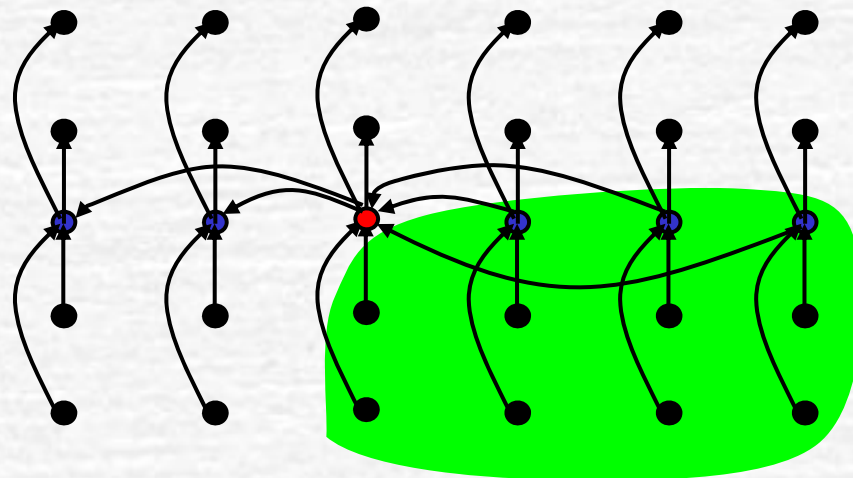
$O(n)$ step 4. 用 x 作为划分元对 A 数组进行划分，并设 x 是第 k 个最小元。

$T(7n/10+6)$ step 5. if $i=k$ then return x ;
else if $i < k$ then 找左区间的第 i 个最小元;
else 找右区间的第 $i-k$ 个最小元;

时间分析 (1)

- n 个元素中至少有多少个元素 $> x$?

每组按列形式，以每组中值升序次序从左到右排列如下：



图中：箭头指向的元素小于箭尾元素

可以知道，大于 x 的元素至少有 $3(\lceil n/5 \rceil / 2 - 2) \geq 3n/10 - 6$

同理，小于 x 的元素至少有 $3n/10 - 6$

- 由上 \Rightarrow 左区间和右区间的最大长度 $\leq 7n/10 + 6$

时间分析 (2)

- 运行时间递归式的建立

step 1, 2: $O(n)$;

step 3: $T(\lceil n/5 \rceil)$;

step 4: $O(n)$;

step 5: 至多 $T(7n/10+6)$

$$\Rightarrow T(n) \leq \begin{cases} \theta(1) & \text{if } n \leq 140 \\ T(\lceil n/5 \rceil) + T(\frac{7}{10}n + 6) + \theta(n) & \text{if } n > 140 \end{cases}$$

时间分析 (3)

- 运行时间递归式的求解

用替代法证: $T(n) \leq cn$

$$T(n) \leq c \lceil n/5 \rceil + c(7n/10 + 6) + an \quad //a \text{ 为常数}$$

$$\leq c(n/5 + 1) + c(7n/10 + 6) + an$$

$$= cn/5 + c + 7cn/10 + 6c + an$$

$$= 9cn/10 + 7c + an$$

$$= cn + (-cn/10 + 7c + an)$$

$$\leq cn \quad // \text{if } -cn/10 + 7c + an \leq 0$$

要使 $-cn/10 + 7c + an \leq 0$, 只要 $c \geq \underline{10an/(n-70)}$

\therefore 假定 $n > 140$, \therefore 有 $n/(n-70) < 2$

\therefore 取 $c \geq 20a \Rightarrow -cn/10 + 7c + an \leq 0$

故 $T(n) = O(n)$





End of Chap9

