Median of Medians on Selection - Sort [für Auswahlproblem]

Algorithmus Auswahl(A, i)

$\boxed{1}$ Wähle Pivot - Element $x \in A$

$\boxed{2}$ $A_{<x} \leftarrow \{a \in A \mid a < x\}$, $A_{>x} \leftarrow \{a \in A \mid a > x\}$

$\boxed{3}$ if $|A_{<x}| = i-1$ then return x

$\boxed{4}$ else if $|A_{<x}| \geq i$ then return Auswahl$(A_{<x}, i)$

$\boxed{5}$ else return Auswahl$(A_{>x}, i-1-|A_{<x}|)$

Optimierung passiert im Schritt $\boxed{1}$

1) — randomisierter Linearzeit - Algorithmus

2) — deterministischer Linearzeit - Algorithmus

Hier wird 2) im Schwerpunkt diskutiert.

Ersetze Schritt $\boxed{1}$ durch folgende Routine

i) partitioniere A in $\lfloor |A|/5 \rfloor$ Teilmengen mit 5 Elementen und eine Teilmenge mit den restlichen ($n \bmod 5$) Elementen - Laufzeit $O(|A|)$

ii) Bestimme für jede dieser Teilmengen deren Median und bilde A_M dieser Mediane - Laufzeit $O(|A|)$

iii) Berechne mittels rekursivem Aufruf von 'Auswahl' den Median x von A_M und verwende x als Pivot - Element.

Jetzt machen wir Laufzeit - Analyse

课件里直接给出

- $|A_M| = \left\lceil \frac{n}{5} \right\rceil$
- $|A_{>x}| \geq 3 \cdot \left(\left\lceil \frac{1}{2} \left\lceil \frac{n}{5} \right\rceil \right\rceil - 2 \right) \geq \frac{3n}{10} - 6 \quad \Rightarrow |A_{<x}| \leq \left\lfloor \frac{7n}{10} + 6 \right\rfloor$
- analog : $|A_{<x}| \geq \frac{3n}{10} - 6 \quad \Rightarrow |A_{>x}| \leq \cancel{\frac{7n}{10}} \left\lfloor \frac{7n}{10} + 6 \right\rfloor$

Folglich gilt für die monotone Laufzeitfunktion des rekursiven Algorithmus

$$f(n) \leq f\left(\left\lceil \frac{n}{5} \right\rceil\right) + f\left(\left\lfloor \frac{7n}{10} + 6 \right\rfloor\right) + cn \text{ , für ein } c > 0 \text{ und alle } n \in \mathbb{N}_{>0}$$

详细解释见背页

1) 为什么 $|A_M| = \lceil \frac{n}{5} \rceil$

因为按照前面对 (i) 的描述，我们把集合A分成了大小为5的小组。每组取一个中位数。

设 $|A| = n$，可以分为 $\lfloor \frac{n}{5} \rfloor$ 个小组，其有5个元素，考虑到 n 不一定能整除5，于是可能存在最后一个小组，它有 $n \bmod 5$ 个元素。②

所以我们一共有 $\lfloor \frac{n}{5} \rfloor + 1$，即 $\lceil \frac{n}{5} \rceil$ 个小组。

A_M 是一个集合，它有 $\lfloor \frac{n}{5} \rfloor + 1 = \lceil \frac{n}{5} \rceil$ 个元素。

2) x是什么，它有什么影响？

x 来自于 (ii)。它是 A_M 集合中的中位数。

因为 x 是中位数，所以在 A_M 中，至少有一半的数大于x，至少有一半的数小于x。①

x 是在 A_M 中用递归的方式找到的，它会被用作全局 Pivot。(iii)

3) $|A_{>x}| \geq 3(\lceil \frac{1}{2} \lceil \frac{n}{5} \rceil \rceil - 2)$ 是怎么推导出来的

→ 在 A_M 中至少有 $\lceil |A_M|/2 \rceil$ 个数 $\geq x$

e.g. a_1 $\boxed{a_2\ a_3\ a_4}$

$a_1\ a_2$ $\boxed{a_3\ a_4}$

$a_1\ a_2$ $\boxed{a_3\ a_4\ a_5}$

再由 A_M 到观察具体的 5-元素一组

$a \in A_M$, mit $a \geq x$

$a_1\ a_2\ a_3\ a_4\ a_5$ $\Rightarrow a_5 > a_4 > a_3$，所以每组至少有3个元素 大于/等于 x

→ ~~考虑到最后一个小组可能不足5个元素~~

考虑到最后一个小组可能不足5个元素，于是 $\lceil \frac{|A_M|}{2} \rceil - 1$

考虑到中位数可能恰好等于 x 的那一组，那么只有2个元素比 x 大，于是 $\lceil \frac{|A_M|}{2} \rceil - 1$

比如 $\{1, 2, x, x+1, x+2\}$ 这里只有 $x+1, x+2$ 比x大

综上 $3 \cdot (\lceil \frac{|A_M|}{2} \rceil - 1 - 1) \leq A_{\geq x}$，$|A_M| = \lceil \frac{n}{5} \rceil$

于是得到安全的下界

排除掉极端情况后，剩下的小组就可以100%保证"每个贡献至少3个元素 $\geq x$"

在上文，我们已得 $|A_{\geq x}| \geq 3\cdot\left(\left\lceil\frac{|A_M|}{2}\right\rceil - 2\right) = 3\cdot\left(\left\lceil\frac{n}{5}\right\rceil\cdot\frac{1}{2} - 2\right) \geq \frac{3n}{10} - 6$

同理，在 A_M 里至少有一半的中位数 $\leq x$

每个这样的5元素小组里至少有3个元素 $\leq$ 该组中位数 $\leq x$

所以 $|A_{<x}| \geq \frac{3n}{10} - 6$

这说明 pivot 至少去掉了 30% 的元素，

$\Rightarrow$ 剩下的子问题大小，最多是

$$n - \left(\frac{3n}{10} - 6\right) = \frac{7}{10}n + 6$$

$\Rightarrow |A_{>x}| \geq \frac{3n}{10} - 6 \quad \Rightarrow |A_{<x}| \leq \left\lfloor\frac{7n}{10} + 6\right\rfloor$

$|A_{<x}| \geq \frac{3n}{10} - 6 \quad \Rightarrow |A_{>x}| \leq \left\lfloor\frac{7n}{10} + 6\right\rfloor$

由此，运行时间满足

$$f(n) \leq f\left(\left\lceil\frac{n}{5}\right\rceil\right) + f\left(\left\lfloor\frac{7n}{10} + 6\right\rfloor\right) + c\cdot n, \text{ für ein } c > 0 \text{ und } n \in \mathbb{N}_{>0}$$

其中 $f\left(\left\lceil\frac{n}{5}\right\rceil\right)$ ：递归在 A_M 上中位数

$f\left(\left\lfloor\frac{7n}{10} + 6\right\rfloor\right)$：递归最多在 70% 的剩余元素上继续查找

$c\cdot n$ ： partion cost

最后个 Satz：Wählt man im Algorithmus 'Auswahl' das Pivot-Element mit der oben beschriebenen rekursiven Routine, dann ist die Laufzeitfunktion in $O(n)$.

Beweis 见课件最后一页.

K8. Median-of-Medians 思想 für Lösung des Auswahlproblems in erwartet linear Zeit

Beobachtung: Wählt man Pivot-Element x in Linearzeit so, dass

$$\max\{|A_{<x}|, |A_{>x}|\} \leq \lfloor d \cdot |A| \rfloor$$

für eine konstante $d < 1$, so hat Auswahl (A, i) Laufzeit $O(n)$

Beweis: Für Laufzeitfunktion $f: N \to N$ gilt dann

$$f(n) \leq f(\lfloor d \cdot n \rfloor) + \tilde{c} \cdot n$$

$d \cdot n$: 子问题的大小

$f(\lfloor d \cdot n \rfloor)$: 递归调用的代价，即进入子问题之后的运行问题

$\tilde{c} \cdot n$: partion cost, 划分代价.

为什么 划分代价是 $\tilde{c} \cdot n$?

因为当我们选择一个 Pivot x 之后，需要把集合拆成三部分

$A<x$, $A=x$, $A>x$

我们需要扫描整个数组，对每个元素做一次比较，判断它属于左边还是右边

即每个元素最多和Pivot 比较一次

所以 代价是 $O(n)$

对于 $f(n)$ 的 Laufzeit，有两种证法：

方法一：

$$f(n) \leq f(dn) + cn$$

$$\leq (f(d^2 n) + c \cdot (dn)) + cn \quad * \text{代入 } n = dn$$

$$\leq (f(d^3 n) + c \cdot (d^2 n)) + c \cdot (dn) + cn$$

$$\dots$$

$$\leq f(d^k n) + cn \cdot (d^0 + d^1 + d^2 + \dots d^{k-1})$$

weil $d < 1 \Rightarrow d^0 + d^1 + \dots d^{k-1} = \sum_{j=0}^{\infty} d^j = \frac{1}{1-d}$

$\Rightarrow f(n) \leq cn \cdot \frac{1}{1-d} + f(d^k n)$

weil $d^k \to 0$, $k \to \infty$

$\Rightarrow f(n) \leq cn \cdot \frac{1}{1-d} + t$, $t \in R$

$\Rightarrow f(n) \in O(n)$

方法 2:

use verallgemeinerter Aufteilungs - Beschleunigungs- Satz

Es sei $f: \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ monoton wachsend, $a \in \mathbb{R}_{>1}$, $b, c \in \mathbb{R}_{>0}$, so dass $f(1) \leq \frac{c}{a}$, und die folgende Ungleichung gilt:

$$f(a \cdot n) \leq b \cdot f(n) + c \cdot n \text{ , für alle } n \geq 1$$

Dann gilt:

$$f(n) = \begin{cases} O(n) & , a > b \\ O(n \log n) & , a = b \\ O(n^{\log_a b}) & , a < b \end{cases}$$

对于这个问题:

$f: N \to N$, dann gilt $f(n) \leq f(\lfloor d \cdot n \rfloor) + \hat{c} n$

我们把公式放一起比较

$$f(n) \leq f(\lfloor d \cdot n \rfloor) + \hat{c} n$$

$$f(a \cdot n) \leq b \cdot f(n) + cn$$

$\Rightarrow$ $b = 1$ $\lfloor d \cdot n \rfloor \cdot a = an$

$$\Rightarrow a = \left\lceil \frac{1}{d} \right\rceil$$

denn $d < 1$ $\Rightarrow a > 1$, $\Rightarrow a > b$

$\Rightarrow f(n) \in O(n)$. ▪

$\boxed{\text{Satz:}}$ Wählt man in Algorithmus Auswahl (A, i) das Pivot-Element $x \in A$ zufällig und gleichverteilt, so ist erwartete Laufzeit $O(n)$

证明见课件