思考题 5——有限空间选择和排序

孟妍廷　2015202009

2017 年 10 月 24 日

首先对题目的几个理解:
1. 只读数组: 只能获取值而不能被赋值的数组, 因此一定需要额外空间才能进行选择和排序
2. k 为 1 到 n 的值, 但是由于对称性我们可以假设 $k \leq \lceil \frac{n}{2} \rceil$. 因为当 k 超过 n 的一半的时候相当于求第 n-k 小的数, 而 n-k 小于 n 的一半, 逻辑仍然相同.
1. 读取一遍, 选择第 k 大的数需要的额外空间至少为 $\Omega(n) > \Omega(k)$, 只有当 k 为中位数或非常接近中位数时, 额外空间为 $\Theta(n^{\frac{1}{2}}) < O(\frac{n}{2}) = O(k)$, 以任意高的可能性找到 k.
通过阅读文献, 证明如下:
不失一般性, 假设额外空间大小为 s. 算法将第一个 s 读入内存, 并决定在读取第 s+1 个时丢弃哪一个.” 敌人策略” 确保第 s+1 个元素进入时不影响被替代的元素之外的所有元素的顺序.
重复” 敌人策略”, 当 $x = \lfloor \frac{n}{2s-1} \rfloor$ 时, 第 $sx + 1$ 个元素正准备被读入, 此时内存中起码有一个位置被替换出去的元素组成一个集合 X, 其中至少有 x 个元素未被比较过且未被排序. 可以证明剩余的 n-sx 个元素的中位数可以被设计为 X 的中位数. 因此每次丢弃 s 个元素产生的未被比较的集合大小最多为 $\lceil \frac{n}{s} \rceil$. 因此辅助定理 3 可证。

Lemma 3. *For any S-location algorithm on N input elements there is an ordering of the input tape so that after the first pass there is a set X of inputs with the following properties:*
(i) *no element of X remains in storage,*
(ii) *no orderings between elements of X are known,*
(iii) *the median of the original set is the median of X,*
(iv) *X contains at least $\lfloor N/(2S-1) \rfloor$ elements.*

由辅助定理 3 可以证明定理 4:

Theorem 4. *Any P-pass algorithm to determine the median (or Kth highest for $\frac{1}{2}N \geq K = \Omega(N)$) of N elements requires at least $\Omega(N^{1/P})$ storage locations.*

此时 p 为 1, 因此平均情况需要的额外空间至少为 $\Omega(n)$, 只有在第 k 大的元素正好位于 k 处时才能达到 $\Omega(k)$
而当 k 为中位数时, 由于为选择合适的额外存储空间 s, 要求当前位于 s 中的 s-1 个元素是连续的并尽可能接近当前的中值, 因此有定理 5:

Theorem 5. *For any $\varepsilon > 0$, $P \geq 1$ there is a P-pass median-finding algorithm with probability of failure at most ε which uses only $O(N^{1/2P})$ storage.*

此时 p 为 1, 得证。

2. 至少需要 $\Omega(n^{\frac{1}{2}})$, 至多需要 $O(n^{\frac{1}{2}} logn)$ 的额外空间。
由 1. 中的定理 4 可知当 p=2 时, 额外空间至少为 $\Omega(n^{\frac{1}{2}})$.

又由辅助定理 1 可得

Lemma 1. *If at most n elements lie between the filters at the beginning of a pass, then for the following pass this number is* $O(n(\log n)^2/S)$.

可以推出定理 2

Theorem 2. *A P-pass algorithm which selects the Kth highest of N elements requires storage at most* $O(N^{1/P}(\log N)^{2-2/P})$.

此时 p=2, 得证。
设计算法如下:
$sort(A,k)$
$len\ =\ \lceil sqrt(A.length)\rceil$
$B[len]\ =\ 0$　初始化
$for\ i=0\ to\ len-1$
　$B[i]=A[i]$　放入A的前len个元素
$while(i<A.length)$　　　　　$for\ j=i\ to\ i+len-1$
　　用A中后len个元素中比$B[0...len-1]$中元素小的替换它们
　　用排序算法对B排序
　$i++$
由此可以得到第 len 大的数, 若 len>=k 已经得到结果, 如果 len<k, 则以第 len 大的数为 B[0]
重复以上步骤, 直至找到第 k 大的数, 该算法的空间复杂度为 $\Omega(n^{\frac{1}{2}})$