

算法设计与分析实验报告

实验名称：第 K 小元素(分治法)

一、问题陈述，相关背景、应用及研究现状的综述分析

1.问题陈述：

给定一个线性序列集，要求求出其中指定的第 K 小的数的值和位置，如给定 n 个元素和一个整数 i， $1 \leq i \leq n$ ，输出这 n 个元素中第 i 小元素的值及其位置；

2.相关背景：

(1) 在某些特殊情况下，很容易设计出解选择问题的线性时间算法。如：当要选择最大元素或最小元素时，显然可以在 $O(n)$ 时间完成。（一趟比较即可）

(2) 一般的选择问题，特别是中位数的选择问题似乎比最小(大)元素要难。但实际上，从渐近阶的意义上，它们是一样的。也可以在 $O(n)$ 时间完成。

二、模型拟制、算法设计和正确性证明

1. 算法设计：

改编随机快速排序，不用把整个数组全部排序，而是选择的排序（更快）

先从数组中随机获取一个元素作为划分基准，然后对数组进行划分，分成左右两个部分，左边元素都比基准要小，右边元素都比基准要大，然后根据两部分元素个数选择子问题继续进行求解。

三、时间和空间复杂性分析

时间复杂度：

(1) 在最坏情况下，算法 randomizedSelect 需要 $O(n^2)$ 计算时间

比如，要找最小的元素，但是每次进行 Partition 函数划分时得到的位置总是很大（靠近 n ）（即总是在最大元素出划分）

(2) 但可以证明，算法 randomizedSelect 可以在 **$O(n)$ 平均时间**内找出 n 个输入元素中的第 k 小元素。

空间复杂度：

没有使用额外空间

$$S(n) = O(1)$$

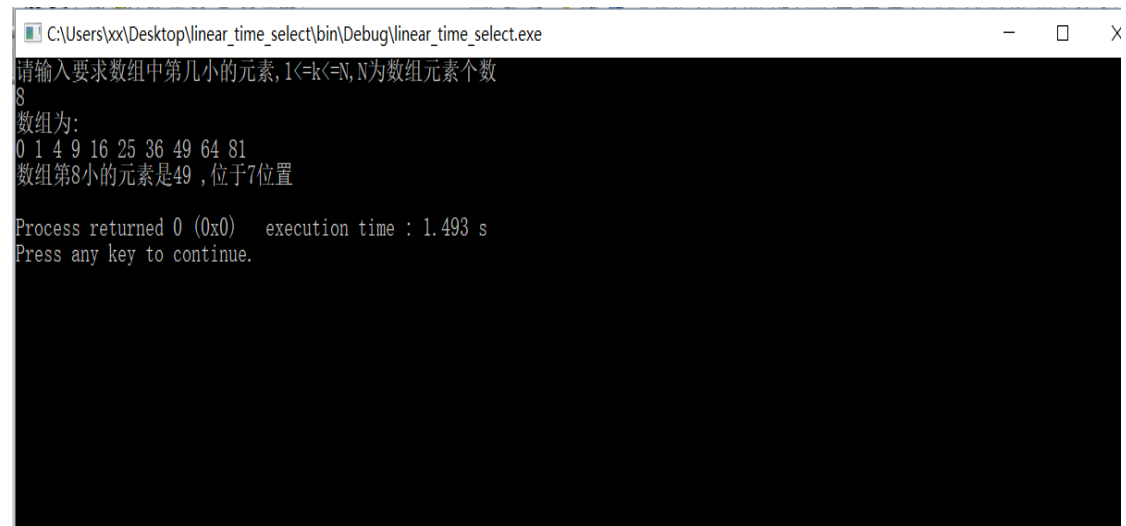
四、程序实现和实验测试过程

源程序见 linear_time_select.cpp

测试过程：

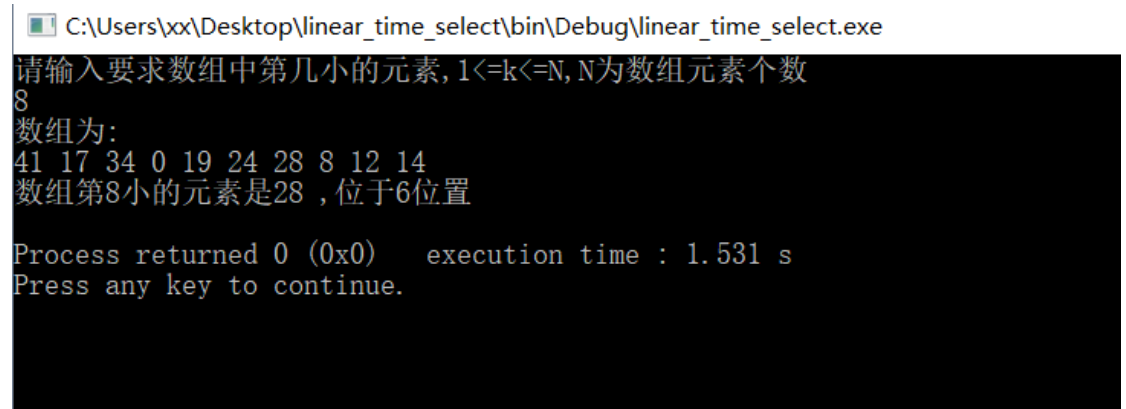
方法 1 要求数组元素互不相同。

方法 2 为课本方法



```
C:\Users\xx\Desktop\linear_time_select\bin\Debug\linear_time_select.exe
请输入要求数组中第几小的元素, 1<=k<=N, N为数组元素个数
8
数组为:
0 1 4 9 16 25 36 49 64 81
数组第8小的元素是49 , 位于7位置

Process returned 0 (0x0)   execution time : 1.493 s
Press any key to continue.
```



```
C:\Users\xx\Desktop\linear_time_select\bin\Debug\linear_time_select.exe
请输入要求数组中第几小的元素, 1<=k<=N, N为数组元素个数
8
数组为:
41 17 34 0 19 24 28 8 12 14
数组第8小的元素是28 , 位于6位置

Process returned 0 (0x0)   execution time : 1.531 s
Press any key to continue.
```

五、总结

直接排序求解为时间复杂度为 $O(n \log n)$

线性时间选择平均时间复杂度为 $O(n)$