

BFPRT 算法 $O(n)$

TOP-K问题：在一大堆数中求其前k大或前k小的问题

一、 算法步骤：

step1: 将n个元素每5个一组，分成 $n/5$ (上界)组，最后的一个组的元素个数为 $n\%5$ ，有效的组数为 $n/5$ 。

step2: 取出每一组的中位数，最后一个组的不用计算中位数，任意排序方法，这里的数据比较少只有5个，

可以用简单的冒泡排序或是插入排序。

step3 : 将各组的中位数与数组开头的数据在组的顺序依次交换，这样各个组的中位数都排在了数据的左边。

递归的调用中位数选择算法查找上一步中所有组的中位数的中位数，设为x，偶数个中位数的情况下设定为选取中间小的一个。

step4: 按照x划分, 大于或者等于x的在右边, 小于x的在左边，关于step4数据的划分，中位数放在左边或是右边会有些影响。

后面的代码调试将会看到。

step5: step4中划分后数据后返回一个下表i，i左边的元素均是小于x，i右边的元素包括i都是大于或是等于x的。

若 $i==k$ ，返回x；

若 $i < k$ ，在小于x的元素中递归查找第i小的元素；

若 $i > k$ ，在大于等于x的元素中递归查找第 $i-k$ 小的元素。

二、 例子示范思路：

=====

待查找的数组：

4 1 2 56 24 5 6 97 8 0 4 8 6 2 3 6 1 9 3 4 6 2

找出第 8 小的数

被分割的数组：

4 1 2 56 24

该数组的中位数：4

被分割的数组：

5 6 97 8 0

该数组的中位数：6

被分割的数组：

4 8 6 2 3

该数组的中位数：4

被分割的数组：

6 1 9 3 4

该数组的中位数：4

中位数的集合：

4 6 4 4

经过选择排序后的中位数数组：

4 4 4 6

定义的 x 为： 4

=====

待查找的数组：

4 1 2 0 4 2 3 1 3 4 2

找出第 8 小的数 //数组长度比8大

被分割的数组：

4 1 2 0 4

该数组的中位数：2

被分割的数组：

2 3 1 3 4

该数组的中位数：3

中位数的集合：

2 3

经过选择排序后的中位数数组：

2 3

定义的 x 为： 2 //前6小的数据已得出(0 1 1 2 2 2)

=====

待查找的数组：

4 4 3 3 4

找出第 2 小的数 3 //(8-6=2 查出第二2小的数据)

第 8 小的数为： 3
(0 1 1 2 2 2 3 3 4 4 4 5 6 6 6 6 8 8 9 24 56 97)