Finding the Kth largest item in a list of n items

SAIKIRAN PITLA

Indiana State University

Terre Haute

November 28 2011

Content

Introduction

- This problem deals with finding the Kth largest element from an unordered list which consists of 'n' elements.

Introduction

- This problem deals with finding the Kth largest element from an unordered list which consists of 'n' elements.
- The easy approach of solving this problem is first sort the unordered list and then return the kth largest element.

Introduction

- This problem deals with finding the Kth largest element from an unordered list which consists of 'n' elements.
- The easy approach of solving this problem is first sort the unordered list and then return the kth largest element.
- This problem can be solved by two algorithms:

Introduction

- This problem deals with finding the Kth largest element from an unordered list which consists of 'n' elements.
- The easy approach of solving this problem is first sort the unordered list and then return the kth largest element.
- This problem can be solved by two algorithms:
  - Selection algorithm.
  - Median of medians algorithm.

Introduction

- This problem deals with finding the Kth largest element from an unordered list which consists of 'n' elements.
- The easy approach of solving this problem is first sort the unordered list and then return the kth largest element.
- This problem can be solved by two algorithms:
    - Selection algorithm.
    - Median of medians algorithm.
- Median of medians algorithm is better than selection algorithm due to its worst case linear time performance.

History

The Median of Medians Algorithm was proposed by 5 great computer scientists they are **Manuel Blum, Robert Floyd, Vaughan Pratt, Ron Rivest** and **Robert Tarjan** in the year **1973**.

Comparison

- In median of medians algorithm, we divide the list by 5 and then we sort the divided list, where as in selection algorithm we directly sort the unordered list with out dividing.

Comparison

- In median of medians algorithm, we divide the list by 5 and then we sort the divided list, where as in selection algorithm we directly sort the unordered list with out dividing.
- Median of medians algorithm has a better performance when compared to selection algorithm.

Algorithm

1. Divide the list in to $n/5$ lists of 5 elements each.
2. Find the median in each sublist of 5 elements.
3. Recursively find the median of all the medians, call it m.
4. Partition the list in to unique elements larger than 'm'(call this sublist L1) and those no longer than 'm' (call this sublists $L2$).
5. If $K <= |L1|$, return selection $(L1, K)$.
6. If $K - 1 = |L1|$, return 'm'.
7. If $K > |L1| + 1$, return selection$(L2, K - |L1| - 1)$.

Example

Find the 8th largest element i.e k= 8.

Unordered list

2 3 5 4 1 12 11 13
16 7 8 6 10 9 17 15
19 20 18 23 21 22
25 24 14

Dividing by 5

2 3 5 4 1
12 11 12 16 7
8 6 10 9 17
15 19 20 18 23
21 22 25 24 14

Example

Finding medians

2	3	5	4	1
12	11	13	16	7
8	6	10	9	17
15	19	20	18	23
21	22	25	24	14

Median of medians

5 13 (10) 20 25

Example

2 3 5 4 1 7 8 6 9

|L2|= 9

10

m

12 11 13 16 17 15 19 20
18 23 21 22 25 24 14

|L1|= 15

Example

12 11 13 16 17
15 19 20 18 23
21 22 25 24 14

⇒ 13 20 25

12 11 13 16 17 15 19
18 14

|L2|=9

20

23 21 22 25 24

|L1|=5

Example

12 11 13 16 17
15 19 18 14

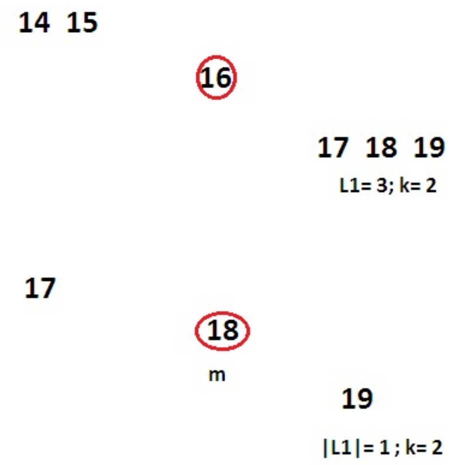
12 11

13

14 15 16 17 18 19

$|L1| = 6; K = 2$

Example



Why 5?

- Dividing the list by 5 assures a worst-case split of $70 - 30$.
- Atleast half of the medians are greater than the median-of-medians, hence atleast half of the $n/5$ blocks have atleast 3 elements and this gives a $3n/10$ split, which means the other partition is $7n/10$ in worst case.

  That gives $T(n) = T(n/5) + T(7n/10) + O(n)$.

Performance

- The best total running time of finding Kth largest item in a list of N items is $O(nlogn)$.
- Where running time of sorting N items is $O(nlogn)$ and running time of returning the Kth largest item is $O(1)$.
- The worst case running time is $O(n)$.

Applications

- **Order Statistics**: Selection include for finding the smallest elements, largest elements and median elements.
- **Computer chess program**: Identifying the most promising candidates in computer chess program.
- **Salary Distribution**: Selection is used in salary distribution.
- **Filtering Outlying** elements such as noisy data.

References

- T.Cormen, C.Leiserson, R. Rivest, and C. Stein Introduction to Algorithms. MIT Press, 2001.
- Donald Knuth The Art of Computer Programming.
- K.C. Kiwiel. On Floyd and Rivest.s SELECT Algorithm, Theoritical Computer Sci.
- Steven S.Skiena The Algorithm Design Manual.