Contents

Finding the k-th ranked element

- Finding the k-th ranked element in an unsorted array is called the selection problem
- Simplest approach is to sort the elements and then picking the k-th element
- Takes $\Theta(n \lg n)$ time – can we do better?
- We could leverage the partition procedure of quicksort

Section outline

Using partition of quickSort – quickSelect

Algorithm to find the k-th ranked of n elements

1. Choose a pivot, use partition to place it correctly at p (say)
2. If the pivot is in the right place (at $p = k$), then done

Using partition of quickSort – quickSelect

Algorithm to find the k-th ranked of n elements

1. Choose a pivot, use partition to place it correctly at p (say)
2. If the pivot is in the right place (at $p = k$), then done
3. If $k > p$, find $(k - p)$-th ranked in the $(n - k)$ larger elements

Using partition of quickSort – quickSelect

Algorithm to find the k-th ranked of n elements

1. Choose a pivot, use partition to place it correctly at p (say)
2. If the pivot is in the right place (at $p = k$), then done
3. If $k > p$, find $(k - p)$-th ranked in the $(n - k)$ larger elements
4. Find the k-th ranked among the $p - 1$ smaller elements

Using partition of quickSort – quickSelect

Algorithm to find the k-th ranked of n elements

1. Choose a pivot, use partition to place it correctly at p (say)
2. If the pivot is in the right place (at $p = k$), then done
3. If $k > p$, find $(k - p)$-th ranked in the $(n - k)$ larger elements
4. Find the k-th ranked among the $p - 1$ smaller elements

Example (5^{th} of 17,12,6,23,19,8,5,10)

6, 8, 5, 10 17, 12, 23, 19 – 1^{st} of
17, 12, 19, 23 – 1^{st} of
12, 17 – done

Analysis of quickSelect

Worst case analysis

- Worst case running time: $O(n^2)$
- Happens when all pivots at an end in each round
- $\mathcal{E}_l$ – event of pivot at any end when working with l elements
- $\Pr[\mathcal{E}_l] = \frac{2}{l}$, if $l > 1$; $\Pr[\mathcal{E}_1] = 1$
- Probabilility for worst case (assuming events are independent):

$$\prod_{l=2}^{n} \Pr[\mathcal{E}_l] = \prod_{l=2}^{n} \frac{2}{l} = \frac{2^{n-1}}{n!} \approx \frac{2^{n-1}}{\sqrt{2\pi n}\left(\frac{n}{e}\right)^n} = \frac{1}{\sqrt{8\pi n}\left(\frac{n}{2e}\right)^n}$$

– rapidly decreases as n increases

Analysis of quickSelect

Worst case analysis

- Worst case running time: $O(n^2)$
- Happens when all pivots at an end in each round
- $\mathcal{E}_l$ – event of pivot at any end when working with l elements
- $\Pr[\mathcal{E}_l] = \frac{2}{l}$, if $l > 1$; $\Pr[\mathcal{E}_1] = 1$
- Probabilility for worst case (assuming events are independent):
$$\prod_{l=2}^{n} \Pr[\mathcal{E}_l] = \prod_{l=2}^{n} \frac{2}{l} = \frac{2^{n-1}}{n!} \approx \frac{2^{n-1}}{\sqrt{2\pi n}\left(\frac{n}{e}\right)^n} = \frac{1}{\sqrt{8\pi n}\left(\frac{n}{2e}\right)^n}$$
– rapidly decreases as n increases
- Worst case is so unlikely, can we show that average case performance is better?

Analysis of quickselect (contd.)

Optimistic analysis

- Possibility of the worst case situation is quite low – $\left(\frac{2}{l}\right)$
- Suppose that pivot has value is in the middle 50% of all elements
- Then, in the worst case also, 25% of the elements can always be discarded

Analysis of quickselect (contd.)

Optimistic analysis

- Possibility of the worst case situation is quite low – $\left(\frac{2}{l}\right)$
- Suppose that pivot has value is in the middle 50% of all elements
- Then, in the worst case also, 25% of the elements can always be discarded
- $T(n) \leq cn + T\left(\frac{3n}{4}\right)$; $T(0) \leq c$
- $T(n) \leq cn \sum_{k=0}^{\left\lceil \log_{\frac{4}{3}} n \right\rceil} \left(\frac{3}{4}\right)^k = 4cn\left(1 - \frac{3}{4}^{\left\lceil \log_{\frac{4}{3}} n \right\rceil + 1}\right) \in O(n)$

Analysis of quickselect (contd.)

Pragmatic analysis

- Some rounds discard less than 25% elements
- Group rounds into phases so that 25% discarded after each phase
- Random variable X_k rounds per phase, maximum $\left\lceil \log_{\frac{4}{3}} n \right\rceil$ phases

Analysis of quickselect (contd.)

Pragmatic analysis

- Some rounds discard less than 25% elements
- Group rounds into phases so that 25% discarded after each phase
- Random variable X_k rounds per phase, maximum $\left\lceil \log_{\frac{4}{3}} n \right\rceil$ phases
- $E[T(n)] \leq E\left[cn \sum_{k=0}^{\left\lceil \log_{\frac{4}{3}} n \right\rceil} X_k \left(\frac{3}{4}\right)^k \right] = cn \sum_{k=0}^{\left\lceil \log_{\frac{4}{3}} n \right\rceil} E[X_k] \left(\frac{3}{4}\right)^k$

Analysis of quickselect (contd.)

Pragmatic analysis

- Some rounds discard less than 25% elements
- Group rounds into phases so that 25% discarded after each phase
- Random variable X_k rounds per phase, maximum $\left\lceil \log_{\frac{4}{3}} n \right\rceil$ phases
- $E[T(n)] \leq E\left[cn \sum_{k=0}^{\left\lceil \log_{\frac{4}{3}} n \right\rceil} X_k \left(\frac{3}{4}\right)^k \right] = cn \sum_{k=0}^{\left\lceil \log_{\frac{4}{3}} n \right\rceil} E[X_k] \left(\frac{3}{4}\right)^k$
- $E[X_k] = 2$ (see next), so $E[T(n)] \leq 8cn \in O(n)$

Computation of $\mathrm{E}[X_k]$

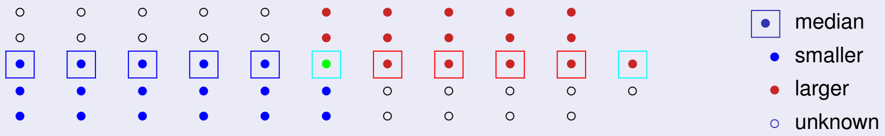
- By definition: $\mathrm{E}[X_k] = \sum_{i=0}^{\infty} i \Pr[X_k = i]$
- Event $[X_k = i]$: $(i-1)$ pivots have value outside middle 50%, but i-th pivot is within middle 50%
- Assuming uniform and independent distribution of the pivot position, $\Pr[X_k = i] = \frac{1}{2^i}$
- $\mathrm{E}[X_k] = \sum_{i=0}^{\infty} i \Pr[X_k = i] \leq \sum_{i=0}^{\infty} \frac{i}{2^i} = 2$

NB $\sum_{i=1}^{\infty} ir^{i-1} = \frac{1}{(1-r)^2}, |r| < 1$

Section outline

Pivot via median of medians

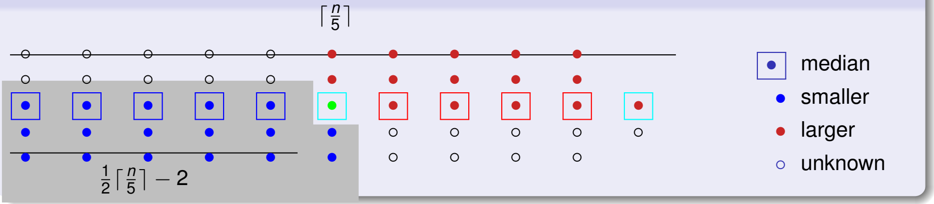
Illustration of median of medians



- Pivot deterministically chosen:
  1. Divide into groups of 5
  2. For each group, find that median of the group
  3. Determine median of the medians and use as pivot
- Proceed as in quickselect – larger or smaller elements definitely discarded while recursing

Pivot via median of medians

Illustration of median of medians



- Pivot deterministically chosen:
  1. Divide into groups of 5
  2. For each group, find that median of the group
  3. Determine median of the medians and use as pivot
- Proceed as in quickselect – larger or smaller elements definitely discarded while recursing

- $3\left(\left\lfloor \frac{1}{2}\left\lceil \frac{n}{5} \right\rceil - 2 \right\rfloor\right) + 2 > \frac{3n}{10} - 6$ elements are definitely discarded
- At most $n - \frac{3n}{10} + 6 = \frac{7n}{10} + 6$ retained
- $T(n) \leq O(n) + T\left(\left\lceil \frac{n}{5} \right\rceil\right) + T\left(\frac{7n}{10} + 6\right)$
- $T(n) \leq O(1), n < 140$, (why?)
- Key observation: $\frac{1}{5} + \frac{7}{10} < 1$
- $T(n) \in O(n)$ – but constants are large

Example of selection by median of medians I

Finding the 8-th ranked among 34 elements

14	32	23	5	10	60	29	57	2	52	44	27	21	11
24	43	12	17	48	1	58	6	30	63	34	8	55	39
37	25	3	64	19	41								

Finding medians of groups of 5 or smaller

5	10	14	23	32
11	21	24	27	44
6	30	34	58	63
3	19	41	64	

2	29	52	57	60
1	12	17	43	48
8	25	37	39	55

Finding median of medians...

Example of selection by median of medians II

Finding the 4-th ranked among 7 elements

14	52	24	17	34	37	19

Finding medians of groups of 5 or smaller

14	17	24	34	52

19	37

Finding median of medians...

19	24

Position of median of medians (19) is 3 among 7 elements, required rank is 4... lower than required rank, find 1-th ranked among 4 elements, by way of problem decompostion...

24	34	37	52

Example of selection by median of medians III

Position of median of medians (24) is 15 among 34 elements, required rank is 8... higher than required rank, find the 8-th ranked among 14 elements, by way of problem decompostion...

Finding the 8-th ranked among 14 elements

5	10	14	23	19	2	3	8	6	17	11	21	12	1

Finding medians of groups of 5 or smaller

5	10	14	19	23
1	11	12	21	

2	3	6	8	17

Finding median of medians...

6	11	14

Example of selection by median of medians IV

Position of median of medians (11) is 8 among 14 elements, required rank is 8... same as required rank, done!

8-th ranked among the 34 integers is 11

Practice problems and developments

- Work out the recurrence equation for selection using median of medians when:
    1. the group size is 3
    2. the group size is 7
- In each case workout whether or not the resulting algorithm will be $O(n)$ or not

---

- The median-of-medians algorithm is by Blum, Floyd, Pratt, Rivest, Tarjan (1973)
- An algorithm that makes $5n + o(n)$ comparisons is due Schonhage, Pippenger, Paterson (1976).
- Same people, same year, an algorithm that makes $3n + o(n)$ comparisons, Schonhage, Pippenger, Paterson (1976).
- Much later, an algorithm that makes $2.95n + o(n)$ comparisons, due to Dor and Zwick (1995).