CSE 421

Divide and Conquer / Median

Yin Tat Lee
Exercise for Midterm

Given a weighted directed acyclic graph with \( n \) vertices and \( m \) edges. Give an \( O(n + m) \) time algorithm to find the shortest path distance from vertex \( s \) to all other vertices. (Hints: Topological sort.)
Exercise for Midterm

Given a connected graph with $n$ vertices and $m$ edges with $m \geq n$. Give an $O(n)$ time algorithm to find a cycle.
Exercise for Midterm

Given an array of elements $a_1, a_2, \ldots, a_n$. Suppose that there are more than half of the elements are same. We call that element as the majority. Given an $O(n \log n)$ time algorithm to find the majority. (Hints: Divide and Conquer)
Median
Selecting k-th smallest

Problem: Given numbers \( x_1, \ldots, x_n \) and an integer \( 1 \leq k \leq n \) output the \( k \)-th smallest number
\[
\text{Sel}(\{x_1, \ldots, x_n\}, k)
\]

A simple algorithm: Sort the numbers in time \( O(n \log n) \) then return the \( k \)-th smallest in the array.

Can we do better?

Yes, in time \( O(n) \) if \( k = 1 \) or \( k = 2 \).

Can we do \( O(n) \) for all possible values of \( k \)?
Choose a number \( w \) from \( x_1, \ldots, x_n \)

Define
\[
\begin{align*}
S_<(w) &= \{ x_i : x_i < w \} \\
S_=(w) &= \{ x_i : x_i = w \} \\
S_>(w) &= \{ x_i : x_i > w \}
\end{align*}
\]

Solve the problem recursively as follows:
\[
\begin{align*}
&\text{If } k \leq |S_<(w)|, \text{ output } Sel(S_<(w), k) \\
&\text{Else if } k \leq |S_<(w)| + |S_=(w)|, \text{ output } w \\
&\text{Else output } Sel(S_>(w), k - |S_<(w)| - |S_=(w)|)
\end{align*}
\]

Ideally want \(|S_<(w)|, |S_>(w)| \leq n/2\). In this case ALG runs in \(O(n) + O\left(\frac{n}{2}\right) + O\left(\frac{n}{4}\right) + \cdots + O(1) = O(n)\).
How to choose $w$?

Suppose we choose $w$ uniformly at random similar to the pivot in quicksort. Then, $\mathbb{E}[|S_{<}(w)|] = \mathbb{E}[|S_{>}(w)|] = n/2$. Algorithm runs in $O(n)$ in expectation. Can we get $O(n)$ running time deterministically?

- Partition numbers into sets of size 3.
- Sort each set (takes $O(n)$).
- $w = \text{Sel}(\text{midpoints}, n/6)$
How to lower bound $|S_<(w)|, |S_>(w)|$?

Assume all numbers are distinct for simplicity.

- $|S_<(w)| \geq 2 \left( \frac{n}{6} \right) = \frac{n}{3}$
- $|S_>(w)| \geq 2 \left( \frac{n}{6} \right) = \frac{n}{3}$.

So, what is the running time?
If $k \leq |S_<(w)|$, output $Sel(S_<(w), k)$
Else if $k \leq |S_<(w)| + |S_{\leq}(w)|$, output $w$
Else output $Sel(S_>(w), k - |S_<(w)| - |S_{\leq}(w)|)$

Where $\frac{n}{3} \leq |S_<(w)|, |S_>(w)| \leq \frac{2n}{3}$

$$T(n) = T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + O(n) \Rightarrow T(n) = O(n \log n)$$
Partition into $n/5$ sets. Sort each set and set $w = Sel(midpoints, n/10)$

- $|S_{<}(w)| \geq 3 \left( \frac{n}{10} \right) = \frac{3n}{10}$
- $|S_{>}(w)| \geq 3 \left( \frac{n}{10} \right) = \frac{3n}{10}$

$$T(n) = T\left( \frac{n}{5} \right) + T\left( \frac{7n}{10} \right) + O(n) \Rightarrow T(n) = O(n)$$
An Improved Idea

Sel(S, k) {  
    n ← |S|  
    If (n < ??) return ??  
    Partition S into n/5 sets of size 5  
    Sort each set of size 5 and let M be the set of medians, so |M|=n/5  
    Let w=Sel(M,n/10)  
    For i=1 to n{  
        If x_i < w add x to S< (w)  
        If x_i > w add x to S> (w)  
        If x_i = w add x to S= (w)  
    }  
    If (k ≤ |S< (w)|)  
        return Sel(S< (w), k)  
    else if (k ≤ |S< (w)| + |S= (w)|)  
        return w;  
    else  
        return Sel(S> (w), k − |S< (w)| − |S= (w)|)  
}