CPS 130 Homework Solutions

Guangwei Yuan

Fall 1999

HOMEWORK 4

3.3: Given function \text{RANDOM}(a, b), which returns randomly an integer between \(a\) and \(b\), inclusive.

\[
\text{RANDOMIZED-\text{PERMUTATION}}(A, 1, n) \\
\text{for } i \leftarrow 2 \text{ to } n \\
\quad k \leftarrow \text{RANDOM}(1, i) \\
\quad \text{exchange } A[i] \leftrightarrow A[k]
\]

We can prove by induction that \text{RANDOMIZED-\text{PERMUTATION}} produces each of \(n!\) permutations with probability \(1/n!\).

4.1: Hint: first sort the list in \(O(n \log n)\) time.

4.2: Note that the \text{rank} of a number in a list is the number of elements less than or equal to it.

(a) Similar to binary search. \(O(\log n)\).

(b) Scan the list, and increase the rank (initialized to be 0) of \(x\) by 1 if the current GPA is less than or equal to it. \(O(n)\).

4.3 We establish a recurrence for the expected running time of randomized \text{SELECT} (see lecture notes for the algorithm).

\[
T(n) \leq \begin{cases} 
\frac{1}{n^2}[\frac{n}{2}T(\frac{3n}{4}) + \frac{n}{2}T(n-1)] + n & \text{if } n > 0 \\
0 & \text{if } n = 0
\end{cases}
\]

We can prove by induction that \(T(n) \leq cn\) for some constant \(c\). So \(T(n) = O(n)\).

4.4 We can improve quicksort by using \text{SELECT} to find the median as the pivot at each step. The recurrence of the running time \(T(n)\) of improved quicksort is therefore: \(T(n) = 2T(n/2) + O(n)\), which solves to \(O(n \log n)\).

4.5 The steps of the algorithm are given below:
1. Use SELECT to find the median $S[m]$ of array $S$ ($T = O(n)$);
2. Compute $D[i] = |S[i] - S[m]|$, for $i = 1, 2, \ldots, n$ ($T = O(n)$);
3. Use SELECT to find the $k$th smallest number $D[j]$ of array $D$ ($T = O(n)$);
4. Apply PARTITION on $S$, using $S[j]$ as the pivot ($T = O(n)$);
5. The first $k$ numbers of $S$, $S[1], S[2], \ldots, S[k]$, are what we need.

It is easy to get the above algorithm is $O(n)$-time.