Sorting III

Sorting lower bounds
  - Decision trees

Linear-time sorting
  - Counting sort

Readings: CLRS 8.1, 8.2, 8.3, 8.4

Comparison Sorting

Insertion sort, merge sort and heapsort are all comparison sorts.

Best worst-case running time we know is $O(n \log n)$

Can we do better?
Sort $\langle a_1, a_2, \ldots, a_n \rangle$

Each internal node labeled $i:j$, compare $a_i$ and $a_j$, go left if $a_i \leq a_j$, go right otherwise

**Example**

$\text{Sort } \langle a_1, a_2, a_3 \rangle = \langle 9, 4, 6 \rangle$

Each leaf contains a permutation, i.e., a total ordering
Decision Tree Model

(can model execution of any comparison sort)

One tree size for each input size n

Running time of algo: length of path taken

Worst-case running time: height of the tree

In order to sort, we need to generate a total ordering of the elements!

Theorem

Any decision tree that can sort n elements must have height \( \Omega(n \log n) \).

Proof: Tree must contain \( \geq n! \) leaves since there are \( n! \) possible permutations.

A height-\( h \) binary tree has \( \leq 2^h \) leaves. Thus, \( n! \leq 2^h \)

\[ \Rightarrow h \geq \log(n!) \quad \left( \geq \log\left(\frac{(n/e)^n}{\sqrt{2\pi n}}\right) \right) \quad \text{Stirling} \]

\[ \geq n \log n - n \log e \quad \geq \Omega(n \log n) \]
Sorting in Linear Time

Counting sort: no comparisons between elements

Input: \( A[i..n] \) where \( A[i] \in \{1, 2, \ldots k\} \)

Output: \( B[i..n] \) sorted

Auxiliary storage: \( C[i..k] \)

Intuition

Since elements are in the range \( \{1, 2, \ldots k\} \), imagine collecting all the \( j \)'s such that \( A[i] = 1 \), then the \( j \)'s such that \( A[i] = 2 \), etc.

Don't compare elements, so it is not a comparison sort!

\( A[i] \)'s index into appropriate positions
PSEUDO CODE & ANALYSIS

\[ \Theta(k) \{ \text{for } i \leftarrow 1 \text{ to } k \]
\[ \quad \text{do } c[i] = 0 \]
\[ \Theta(n) \{ \text{for } j \leftarrow 1 \text{ to } n \]
\[ \quad \text{do } c[A[j]] = c[A[j]] + 1 \]
\[ \Theta(k) \{ \text{for } i \leftarrow 2 \text{ to } k \]
\[ \quad \text{do } c[i] = c[i] + c[i-1] \]
\[ \Theta(n) \{ \text{for } j \leftarrow n \text{ downto } 1 \]
\[ \quad \text{do } b[c[A[j]]] = a[i] \]
\[ \quad c[A[j]] = c[A[j]] - 1 \]
\[ \Theta(n+k) \]

EXAMPLE

Note: records may be associated with the A[i]'s

\[ \begin{align*}
A & : & 1 & 2 & 3 & 4 & 5 \\
   & : & 4 & 1 & 3 & 4 & 3 \\
B & : & 1 & 3 & 3 & 4 & 4 \\
\end{align*} \]

\[ \begin{align*}
c[3] &= 3 \\
c[4] &= 5 \\
\end{align*} \]

and so on...