Odds + Ends

Sorting, Selecting, Searching

We can sort in $O(n \log n)$ time

Quicksort (average)

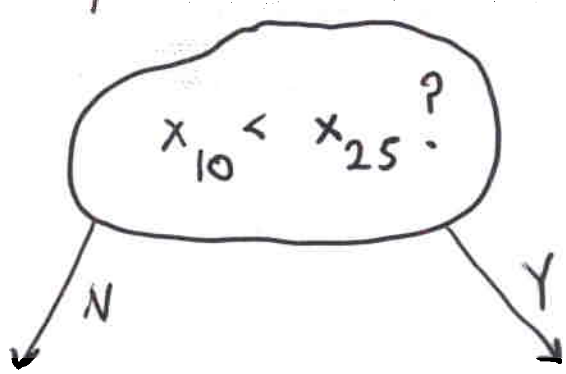
Merge sort (worst-case)

Is this bound tight?

Decision Tree Model

Input: a permutation of n numbers

At each node, ask any yes-no question about the permutation (comparisons a special case)



Each permutation reaches a different leaf

Depth = # questions $\le$ time

Too broad: no accounting for deciding what questions to ask: program need not be of fixed size.

Too narrow: only binary decisions

There are $n!$ permutations on n items.

$\lceil \lg(n!) \rceil \equiv \lceil \log_2(n!) \rceil$ is a lower bound on the worst-case depth.

Stirling's approximation for $n!$

$$n! \geq \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$$

$$\Rightarrow \lceil \log_2 n! \rceil \geq n \log_2 n - n/\ln 2$$

$\Rightarrow$ sorting takes $n \lg n - n/\ln 2$ comparisons (or binary decisions) (worst-case)

Information-Theoretic Lower Bound

Average case?

Lower bound: sum of depths of leaves / $n!$

sum of depths of leaves = external path length

minimized for N external nodes

at $(q+1)N - 2^q$ where $q = \lceil \lg N \rceil$

(complete binary tree: all leaves on at most 2 levels)

$$EPL = N \lg N + 1 + \theta - 2^{\theta}$$

where $\theta = \lg N + \theta$

$$1 + \theta - 2^{\theta} \geq 0$$

$\leq .0861$

For sorting, ave length is $\geq EPL(n!)/n!$

$\geq \lg n! \geq n \lg n - n/\ln 2$

Ave case analysis of quicksort

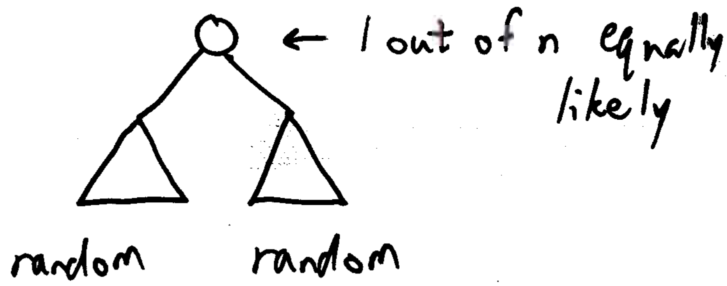
Compare each item to splitter
(n-1 comparisons)

Recursively sort smaller set, larger set

$$A_n = n-1 + \frac{1}{n}\sum_{k=0}^{n-1}(A_k + A_{n-k-1})$$

$$A_0 = A_1 = 0$$

$A_n/n + 1$ is also the ave depth
of an unbalanced binary tree
constructed by random insertions



Observe: items i and j are compared iff one of them is chosen as a splitter before anything in between. The chance of this is $2/(j-i+1)$.

Expected # comparisons:

$$\sum_{i=1}^{n-1} \sum_{j>i} 2/(j-i+1) = \sum_{i=1}^{n-1} \sum_{k=2}^{n-i+1} 2/k$$

$$= \sum_{l=2}^{n} \sum_{k=2}^{l} 2/k$$

$$\leq 2n\ln(n) - O(n) \doteq 1.3863\, n \lg n$$

$$H_l = \sum_{k=1}^{l} 1/k \doteq \ln l + 1$$

Worst-case analysis of merge sort

Divide into two halves

Sort halves

Merge

$$W(n) = W(\lceil n/2 \rceil) + W(\lfloor n/2 \rfloor) + n - 1 = \sum_{k=1}^{n} \lceil \lg k \rceil$$

$$W(n) \leq n \lceil \lg n \rceil - 2^{\lceil \lg n \rceil} + 1 = n \lg n - O(n)$$

Selection in $O(n)$ time

Find the k^{th} out of n

Quick select: like quick sort, but only
recur in the half containing the desired item

if small half contains $i \geq k$, look for k^{th}
in small half

if small half contains $k-1$, splitter is k^{th}

else look for $k-(i+1)^{st}$ in large half

$O(n)$ on average (analysis like quicksort)

Worst-case $O(n)$: double recursion

Divide into $\lceil n/5 \rceil$ sets of 5 (or less)

Find medians of 5

Find medians of medians

Use m of m as a splitter

Recur in appropriate half

$S(n) = O(n) + S(n/5) + S(7n/10)$ ← $3n/10$ discarded

linear since $1/5 + 7/10 = 9/10 < 1$

Optimum Search Trees

Lower bound on weighted (external)

path length in a binary tree

(entropy bound)

Positive weights p_i, $P = \sum_1 p_i$

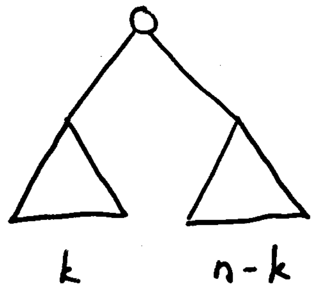
$$WEPL' \geq \sum_{i=1}^{n} p_i \lg(P/p_i)$$

WEPL' counts edges, not nodes

(+P to get node bound)

Proof by induction

$n \geq 2$



$$WEPL' \geq P + \sum_{i=1}^{k} p_i \lg(P'/p_i) + \sum_{i=k+1}^{n} p_i \lg((P-P')/p_i)$$

$$\geq \sum_{i=1}^{n} p_i \lg(P/p_i) + f(P')$$

for some k, where

$$f(P') = P + P' \lg P' + (P-P') \lg (P-P') - P \lg P$$

$f(P')$ is non-negative, taking its min val 0

at $P' = P/2$

This bound is achievable to within P

(non-alphabetic: Huffman codes)

or to within $2P$

(alphabetic: Knuth-(internal) or

Hu-Tucker (external))

What about internal path length?

$$WIPL \geq \sum_{i=1}^{n} p_i \lg(P/p_i) - \lg\left(\sum_{i=1}^{n} p_i \lg(P/p_i)\right) - O(1)$$

if $P = 1$