Heaps

Heap has root either < || > all children.

10,6,8,2,5,7,4,1, read in rows, 10-root,6/8 is next row, right to left

When replace node, always pick the bottom rightmost. Then heapify(look for c</>p)

Topo sort = inverse of pop

Hashing

map a file size n into a table size m use f(n)

Open Hashing: Each Cell is a header of linked list of all keys assigned to it

Closed Hashing: One key per cell.

In case of collision, either linear probing (find next available free cell) or Double Hashing

Key Comparisons = depth of keys

AVG number = Σ1/n(depth)

String Matching

String Matching: Horspool: create a shift table with contains values for how far to shift if a match is found. Always start from the right. EX for BARBER

B=2,A=4,R=3,B=2,E=1, ELSE 6

Beyer Moore: Same, but also create a table for the substring matches

Facts

Insertion Sort: Compare items sequentially

Topo sort is DFS, record PopPush

MergeSort O(nlogn) QuickSort O(n^2)

Tree Traversal: Preorder: Ro,L,R|Inorder:L,Ro,R|Post:-L,R,Ro|

AVL Tree cannot have a height difference > 1

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com