CS61B Week 11: Red-Black Trees, Tries, and Review

1. Starting with an empty red-black tree, draw the result of inserting each of the following values, in this order: [12, 4, 7, 15, 22, 6]

2. Draw the result of removing the nodes from the tree above, in this order: [15, 6, 4, 7, 12, 22]

3. Assume we have a trie where internal nodes are stored as character-indexed arrays, and nodes with only one children are optimized not to have an internal array. Give an upper bound on the total length of these arrays if:

   (a) every word has a maximum length of 3, and otherwise there is no limit on the number of words stored.

   (b) there are 1000 words stored, and each has a maximum length of 200 characters.

Sample Interview Question of the Week:
Given two binary trees, S and T, check if S is a subtree of T. A subtree of T is a tree consisting of a node in T and all of its descendants in T. For example, the subtree corresponding to the root of node of T is the entire tree T.

Write down important things to remember about each of the following topics:
Algorithmic Analysis (e.g. Ω, Θ, big-O, amortized time)

Collections (e.g. what each is good for, running times)

Trees (e.g. three traversals, level-order traversal, BSTs)

Backtracking Search, Minimax

Generic Programming (e.g. where to put them, wildcards, type bounds)

Heaps (e.g. time for each operation)

Range Queries

Hashing (e.g. hash functions, external chaining vs. open addressing)

Sorting (e.g. insertion sort, heapsort, mergesort, quicksort, radix sort)