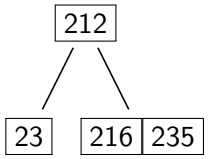
hw27: 2-3 tree practice

You may do this homework on paper instead of in LaTeX. Scan to PDF and submit, as usual. Make sure your scan is legible.

Read the notes and play with the visualization linked from the calendar if needed. But try not to rely solely on the visualization to complete this homework.

(Do enough extra practice such that you are confident you can do any 2-3 tree operation on paper if asked on an exam without aid of notes or visualization.)

Let T be the following 2-3 tree.



Ex 1. The 4 items stored in T are $\{23, 212, 216, 235\}$. Adding to an empty 2-3 tree a *permutation* of those 4 items may or may not yield the same tree T.

(a) Give 2 different permutations of those 4 items that would result in T.

(b) Give a permutation that results in a different 2-3 tree T'; draw T'.

Ex 2. Add to T three items 100, 105, and 241, in that order. Show each step: draw a tree each time the tree changed. Box the tree at the end of adding each item. For example, after adding 100, box that tree, and then show steps to add 105, box the result, and then continue. You will box a total of three trees, one after each item is fully added.

Ex 3. The **height** of a tree is the max number of edges from root to a leaf. For example, T has height 1; a tree with height h has $h + 1$ levels. Suppose we have a 2-3 tree of height h with $h \geq 1$.

(a) What is the min/max number n of items stored in such a tree? Explain. (Give exact lower/upper bounds for n in terms of h.)

(b) What is the max number of comparisons needed to search for an item in such a tree? Explain. (Give exact answer in terms of h.)

(c) What is the time complexity of a Search operation in a 2-3 tree with n items? Explain. (Combine the two parts above to get a Big-Oh bound in terms of n; there should be no h in your final answer.)

(d) Medidate for a moment to convince yourself that Add and Delete operations also obey the same Big-Oh bound.