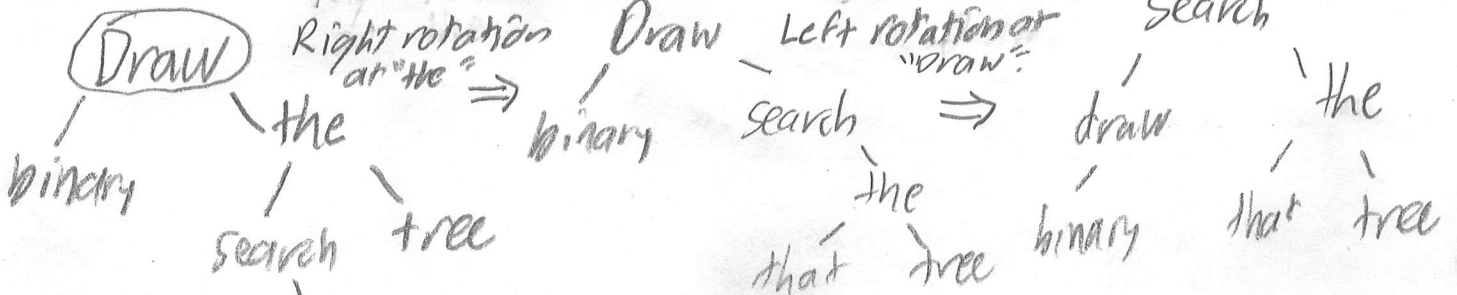
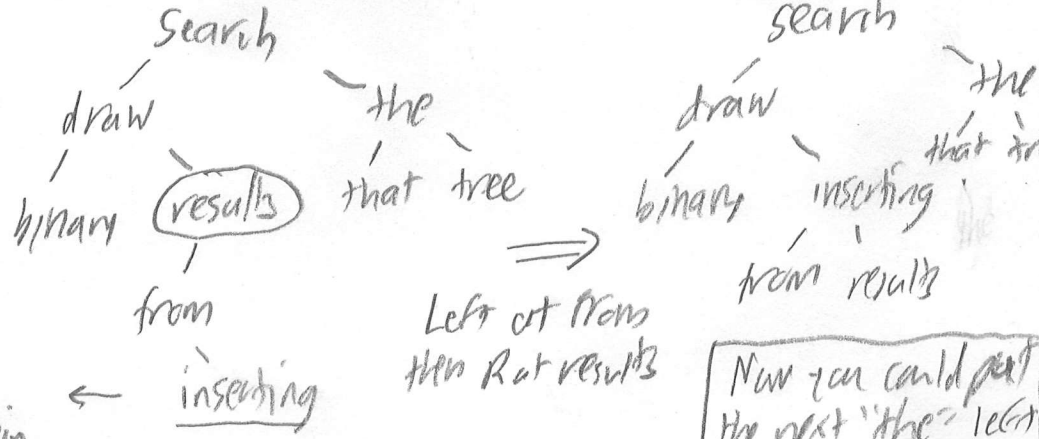
(1)



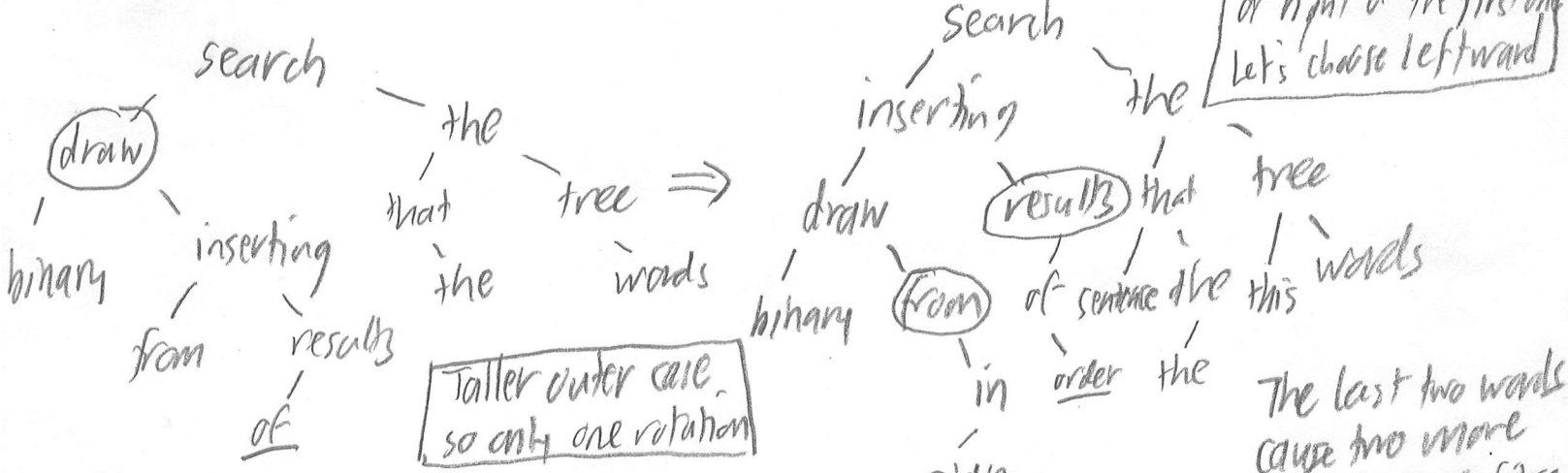
Causes imbalance at Draw. Taller inner grandchild case needs a double rotation.



Now imbalance at multiple nodes, but we fix the lowest one first. "Taller inner" case again ←

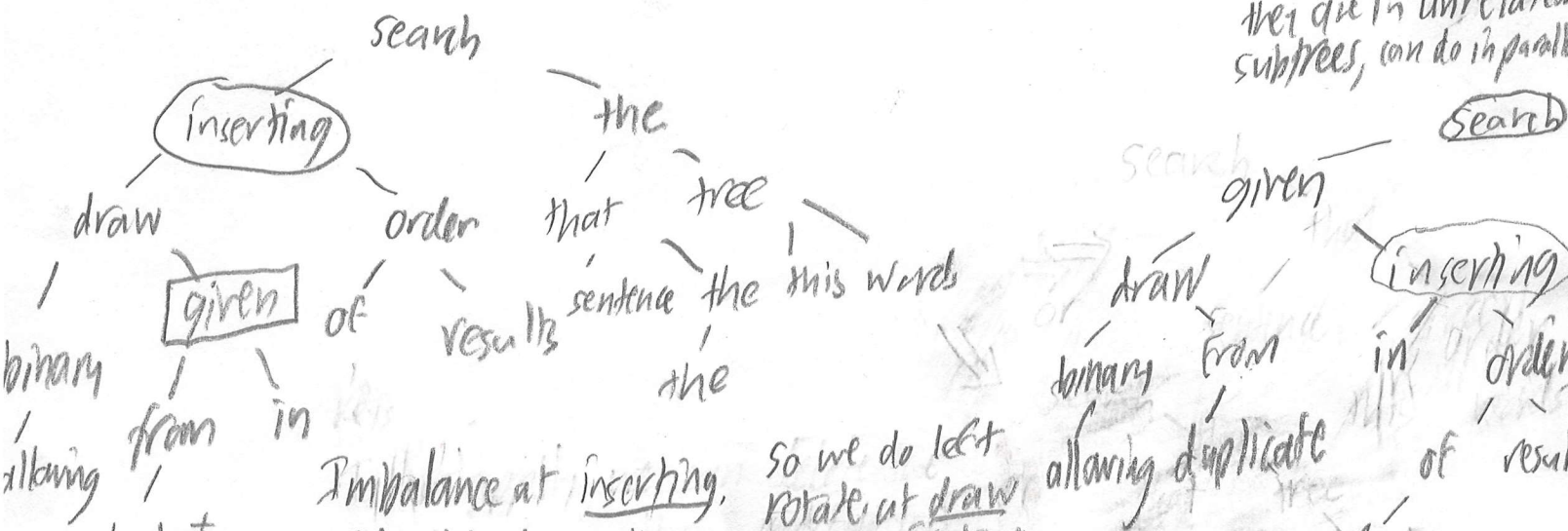
Left at from then Rat results ⇒

Now you could pick the next "the" left or right of the first one. Let's choose leftward



Taller outer case, so only one rotation

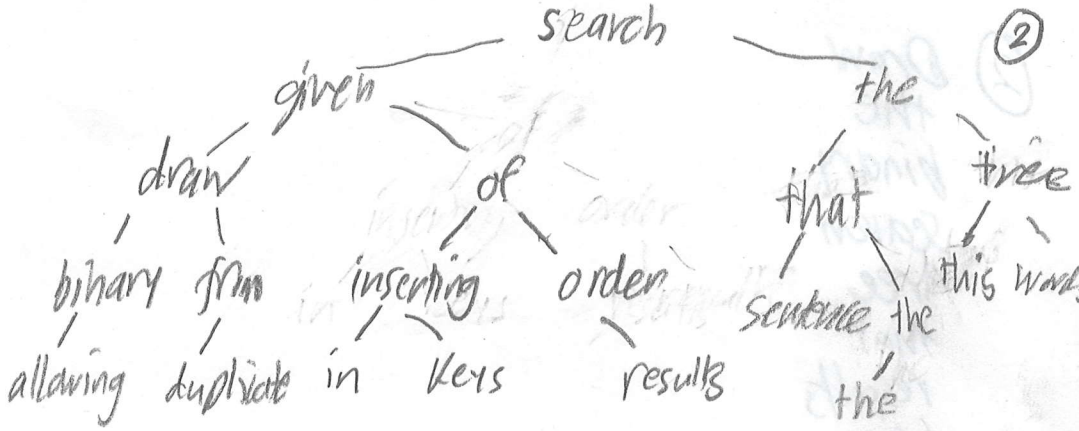
The last two words cause two more imbalances. Since they are in unrelated subtrees, can do in parallel



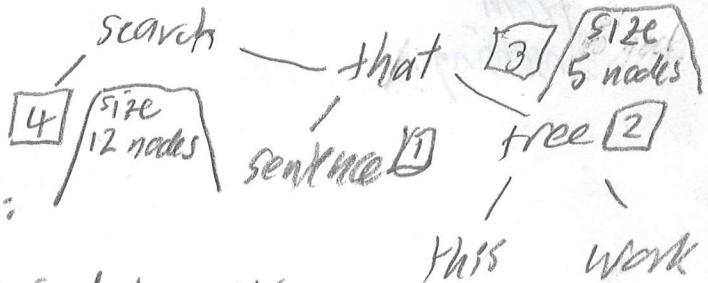
Imbalance at inserting, with given being the "taller inner" case again

So we do left rotate at draw before right at inserting

The insertion of "keys" causes an imbalance at inserting. Once again, we have a "taller inner" case so we have to rotate at "order" too. The resulting whole tree:



To remove the three occurrences of "the" we can simply delete the two lower ones. The last one can be replaced by its inorder predecessor "that" or the successor "this". Showing the former:



With the latter too — and with other forms obtained by putting new equal keys rightward rather than leftward (see AQKey.Scala) — the right subtree of the root is much smaller than the left subtree (5 vs. 12 nodes). But it does not cause a height imbalance. So that is the final tree.

(2)	Sum	%8
draw	24	0
the	13	5
binary	43	3
search	11	3
tree	10	2
that	21	5
results	39	7
from	28	4
inserting	21	5
the	13	5
words	23	7
of	21	5
This	28	4
sentence	8	0
in	23	7
the	13	5
order	23	7
given	19	3
allowing	21	5
duplicate	29	1
keys	44	4

It is curious that out of 19 different words
- none hashes to 6
- only one each goes to buckets 1 & 2!

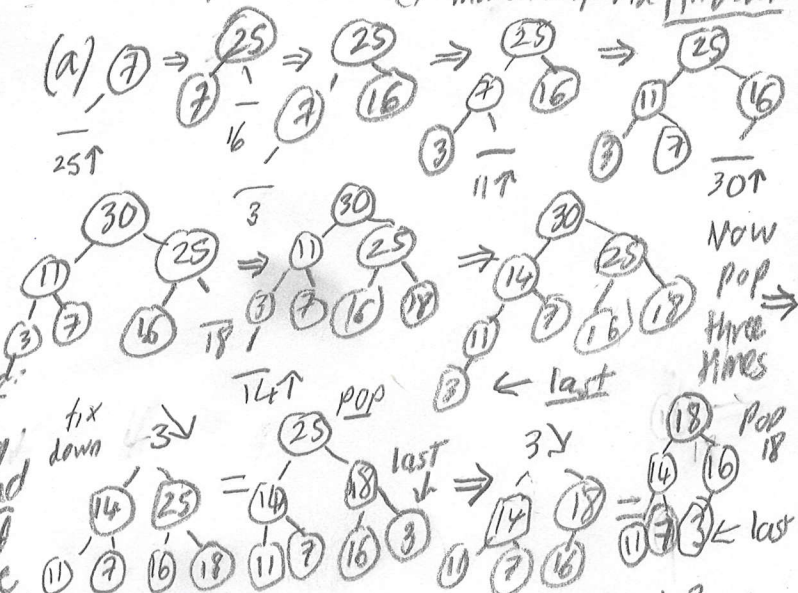
(I see this kind of "lumpy randomness" all the time in my chess data.)

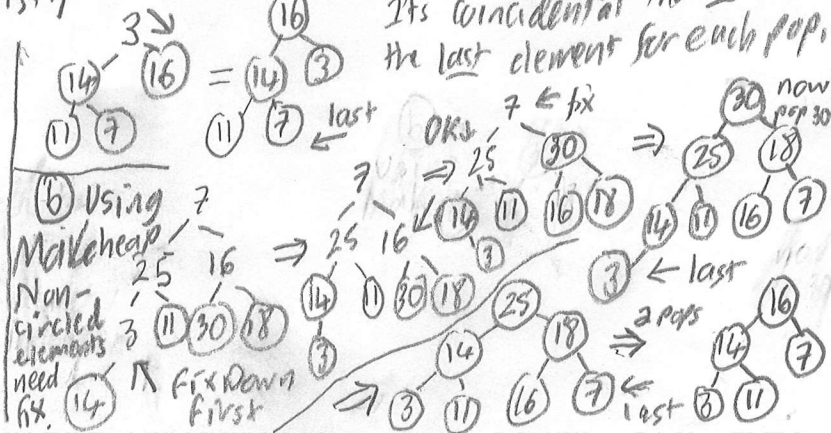
Using append rather than prepend:

0	draw-sentence
1	duplicate
2	tree
3	binary-search-given
4	from-this-keys
5	the-that-inserting-the-of-the-allowing
6	
7	results-words-in-order

((Using prepend would reverse each list.)

(3) Can heap either by (a) inserts and fixUp (b) makeHeap via fixDown





It's coincidental that 3 was the last element for each pop.