

INSERT 15

BEGIN WITH
EMPTY BST
(ONE EXTERNAL NODE)

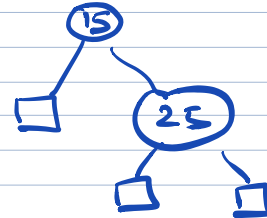
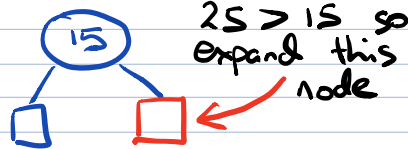
AFTER
INSERTING,
THIS IS THE
RESULT



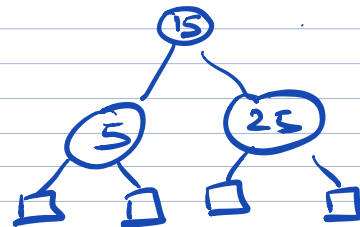
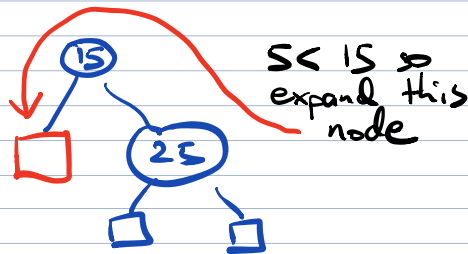
EXAMPLE OF INSERTING
& REMOVING TO/FROM
A BINARY SEARCH
TREE

FROM LECTURE WEEK 3

INSERT 25

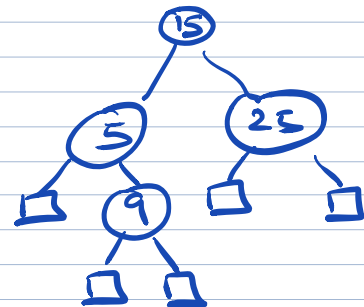
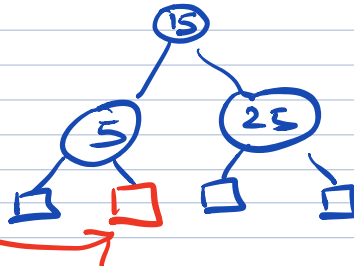


INSERT 5

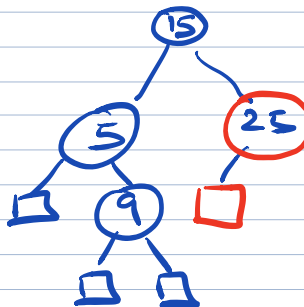


INSERT 9

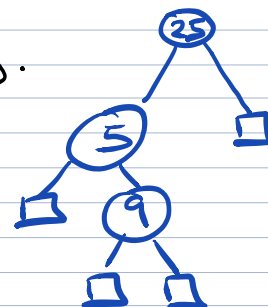
$9 < 15$ so go left
 $9 > 5$ so expand
this node



REMOVE 15



15 has 2 internal children, so it
cannot be removed directly.
Find the next biggest value
(leftmost child in the right
subtree) and copy its
value to replace 15.
Then remove this node
and its left child.

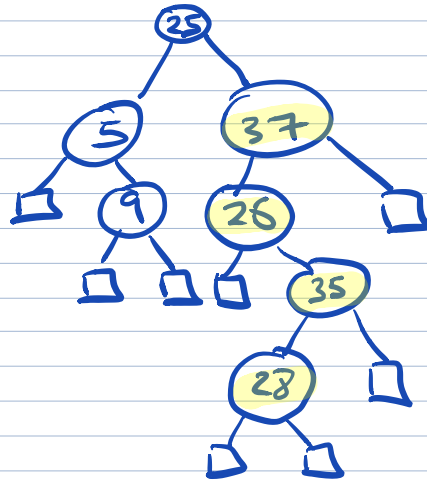


INSERT 37

INSERT 26

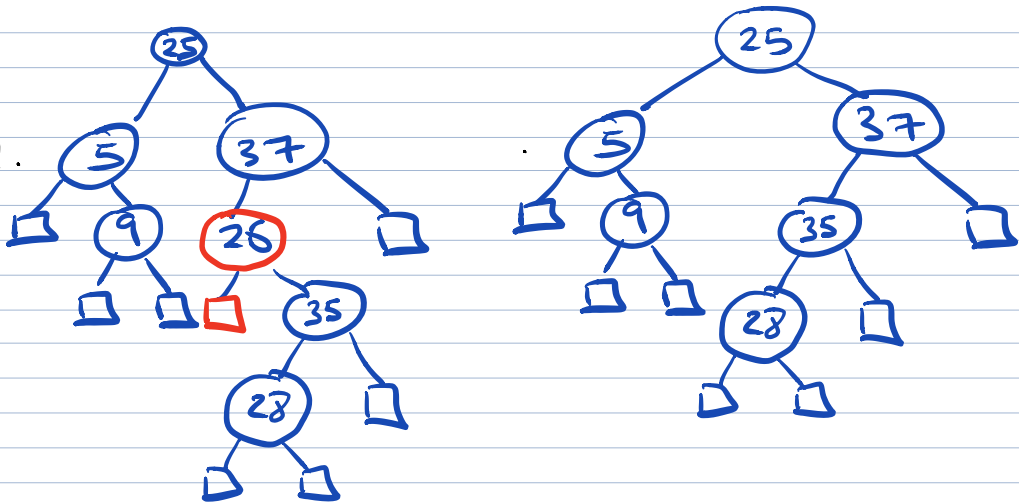
INSERT 35

INSERT 28

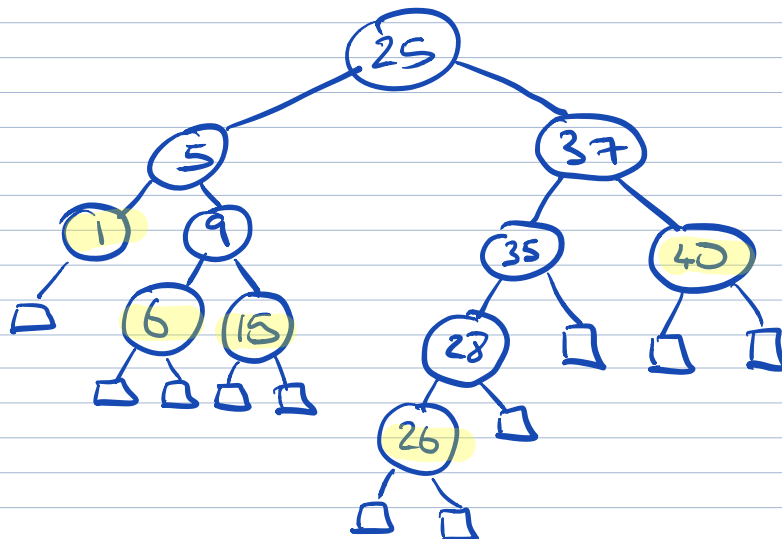


REMOVE 26

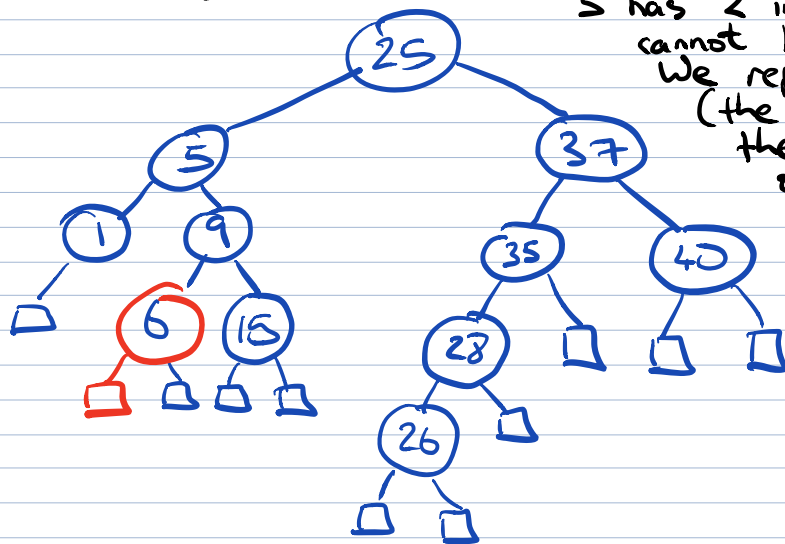
26 has an external child. Remove directly (no need to swap).



INSERT 40
INSERT 1
INSERT 15
INSERT 26
INSERT 6



REMOVE 5



5 has 2 internal children so cannot be removed directly. We replace 5 with 6 (the next biggest value), then remove the node of 6 and its left child.

FINAL TREE

