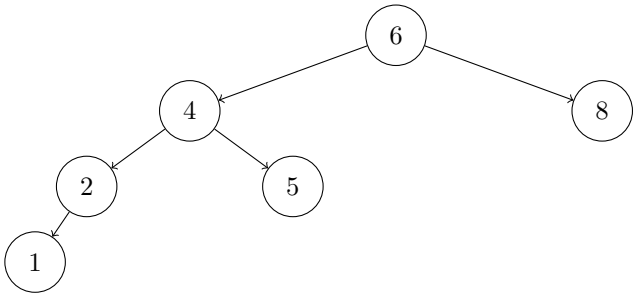
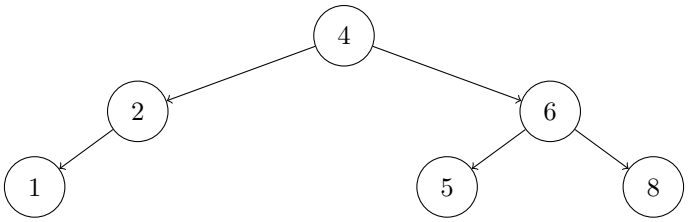
1 AVL Trees

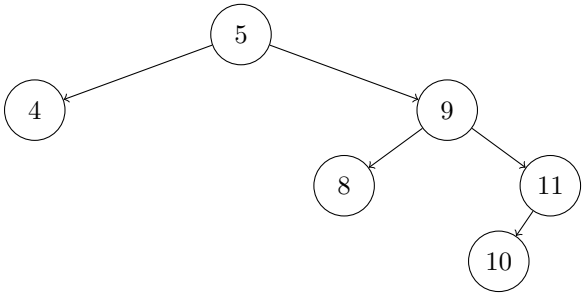
Problem 1. Perform a right rotation on the root of the following tree. Be sure to specify the X, Y, and Z subtrees used in the rotation.



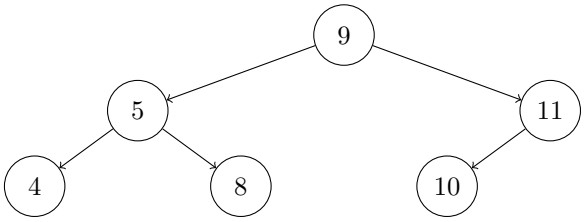
In the following rotated tree, x = 2, 1; y = 5; z = 8



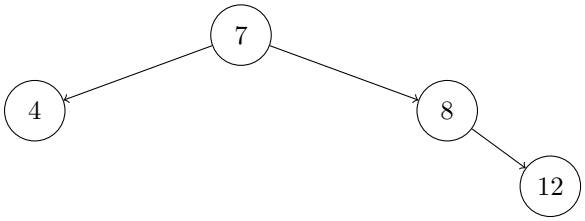
Problem 2. Perform a left rotation on the root of the following tree. Be sure to specify the X, Y, and Z subtrees used in the rotation.



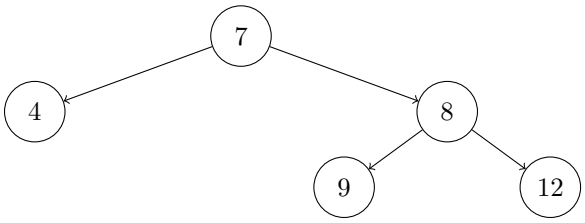
In the following rotated tree, x = 4; y = 8; z = 11, 10.



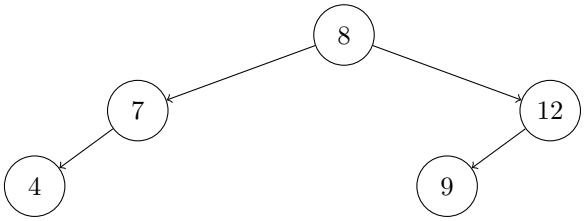
Problem 3. Using the appropriate AVL tree algorithm, insert the value 9 into the following tree. Show the tree before and after rebalancing.



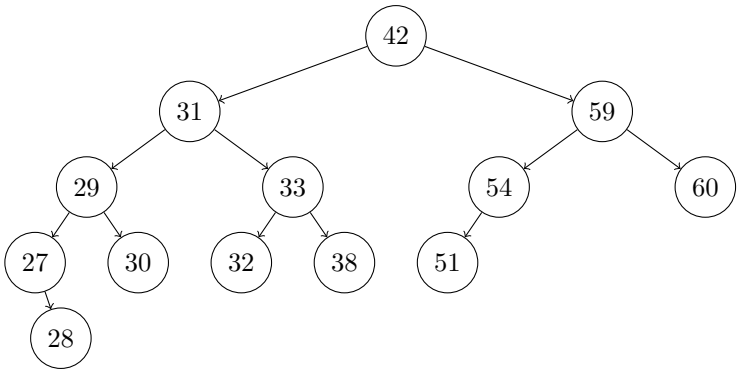
After inserting 9 into the tree above, the tree before rebalancing is as follows:



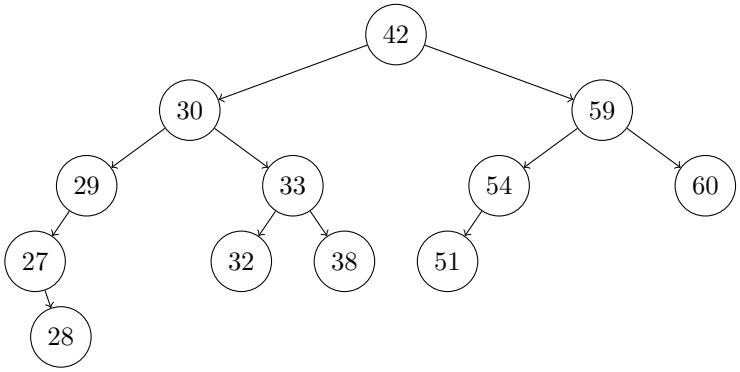
The re-balanced tree is as follows, where w = 4, y = 9, and x, z = null.



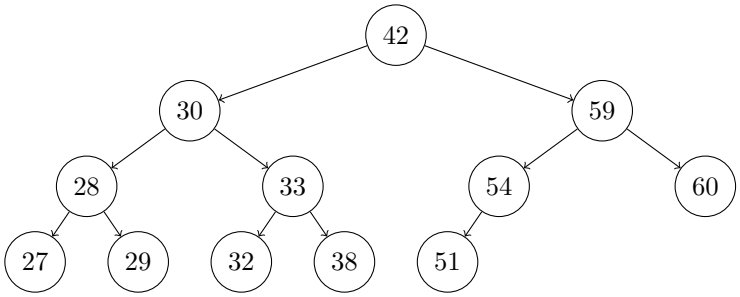
Problem 4. Using the appropriate AVL tree algorithm, remove the value 31 from the following tree. Show the tree before and after rebalancing.



After removing 31 into the tree above, the tree before rebalancing is as follows:

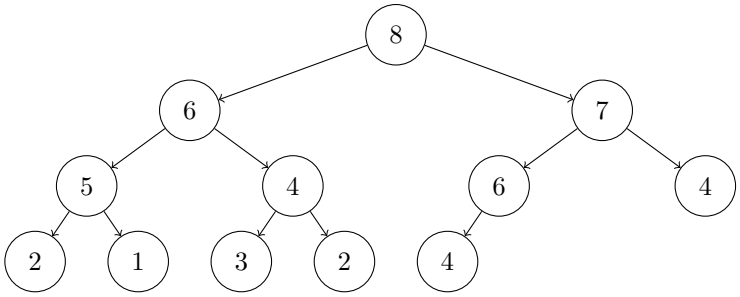


The re-balanced tree is as follows:

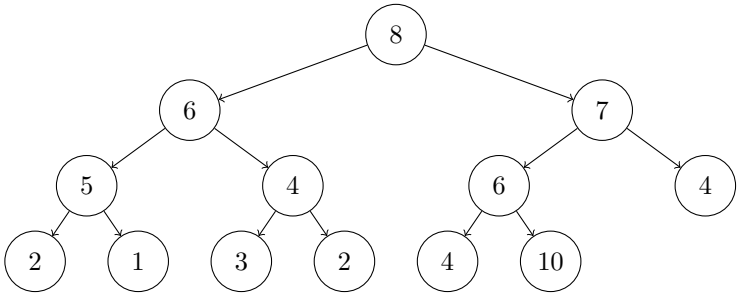


2 Heaps

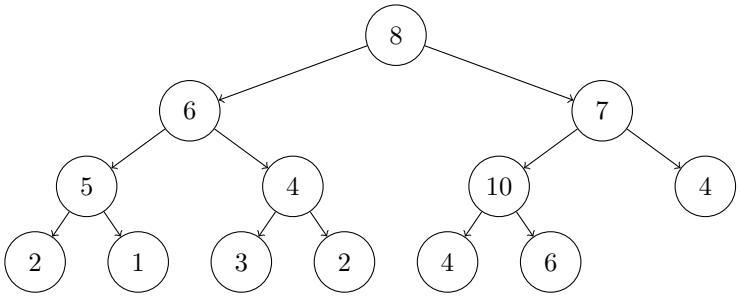
Problem 1. Show the addition of the element 10 to the max-heap below. First, show the addition of 10 to the tree; then, show each bubbling step.



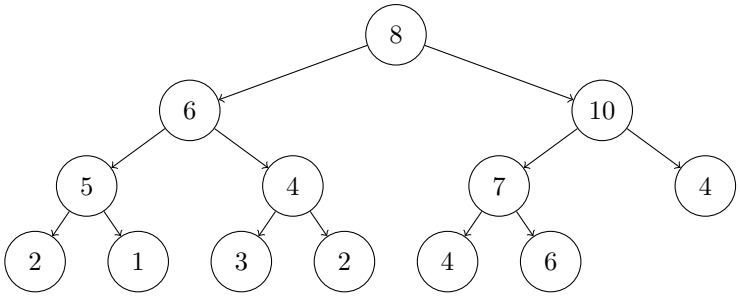
The addition of 10 to the max-heap:



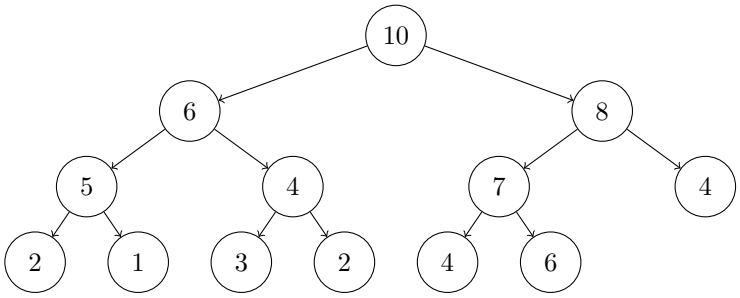
Bubbling up: (1)



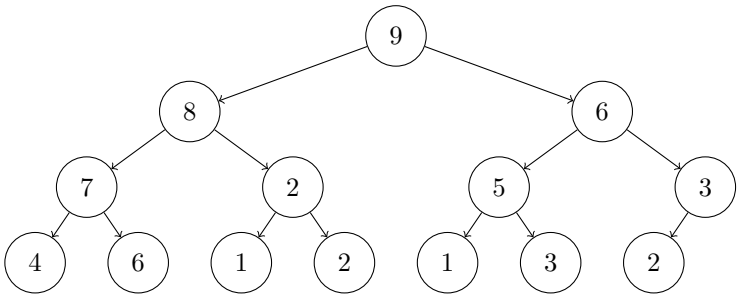
(2)



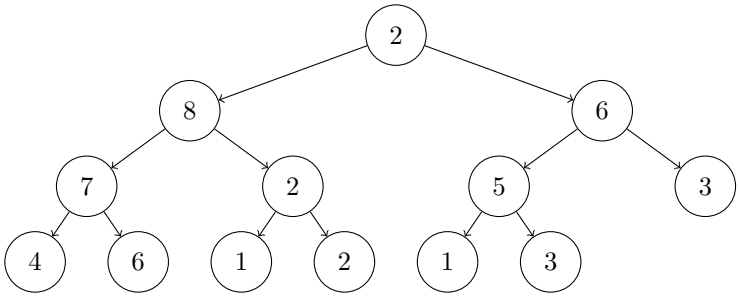
(3) Final Heap



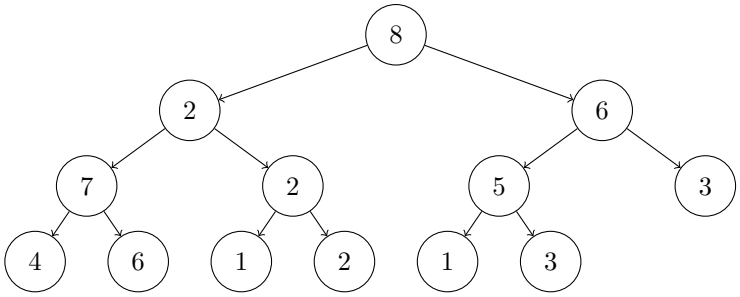
Problem 2. Show the removal of the top element of this max-heap. First, show the swap of the root node; then, show each bubbling step.



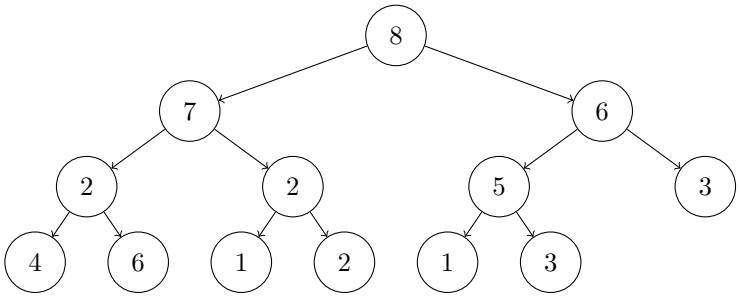
The removal of the top element, 9, from the max-heap:



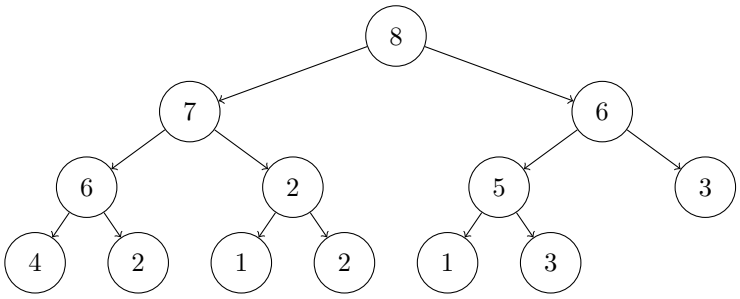
Bubbling down: (1)



(2)

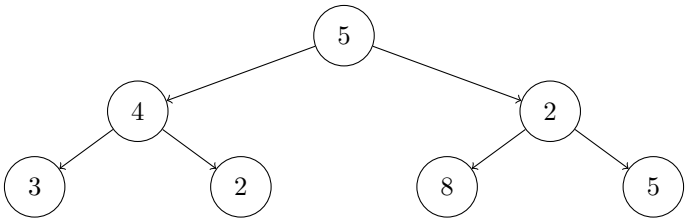


(3) Final Heap

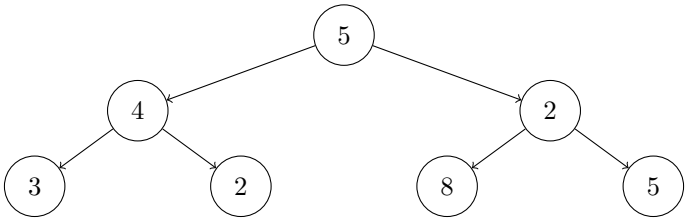


Problem 3. Consider the sequence of elements `[5,4,2,3,8,5]`. Using the representation discussed in class, show the tree to which this sequence corresponds. Then, show the *heapification* of this tree; that is, show how this tree is transformed into a heap. Demonstrate each bubbling step.

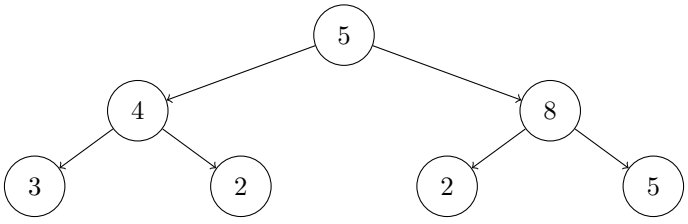
The following complete binary tree represents the list 5,4,2,3,2,8,5:



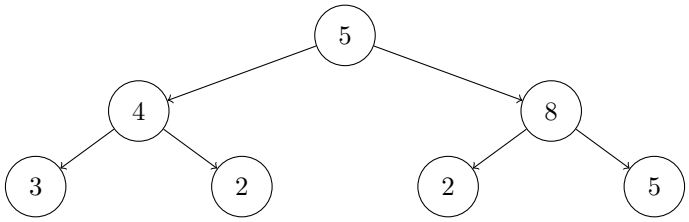
The bubble-down operation as the tree is heapified is shown as follows: (1) Starting with four leafs: 3, 2, 8, 5.



(2) BubbleDown(2); 2 is swapped with its maximal-valued child, 8



(3) BubbleDown(4); Since 4 is larger than both of its children, it remains in the same place.



(4) BubbleDown(5); Since 5 is smaller than its child, 8, 8 is swapped with 5.