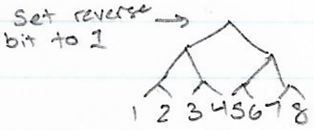
6) In order to say that we will have this tree represented in reverse order, we put a 1 bit onto the root level. This is simply one operation, making the complexity O(1). The reverse bits in a node would then be pushed down to each of its children and then swapping the children. Then, by clearing the parent, you could consider finding the kth value the same way as in q5.

For example, given a tree representing leaves with values 1-8

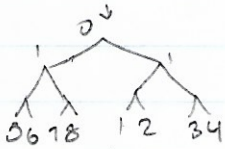


Assuming k = 5

↳This is finding the kth number in the reversed tree

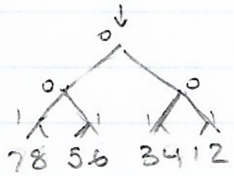
↓

Zero bit and Push down to children and swap



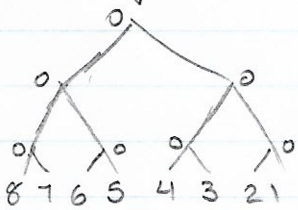
5 > 4 → Search path for k is to the right

counter = 5-4 → 1



1 < 2 → Path is to the left

↓



Path of k is to the left again

Making the element at k = 4 of the reverse

Because the algorithm to find the path to k is preserved, we can derive split and concat in the same runtime