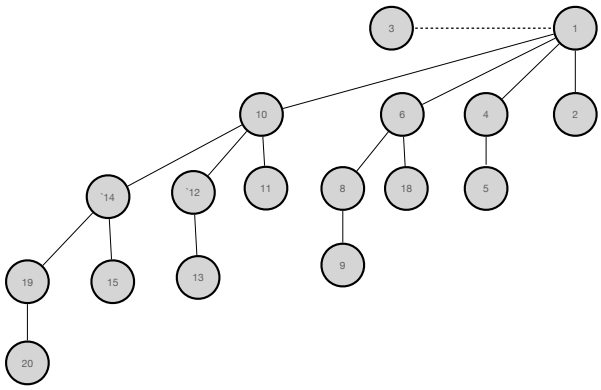
EC 504
Spring, 2021
Exam 1.6

Friday, March 19, 2021

Instructions

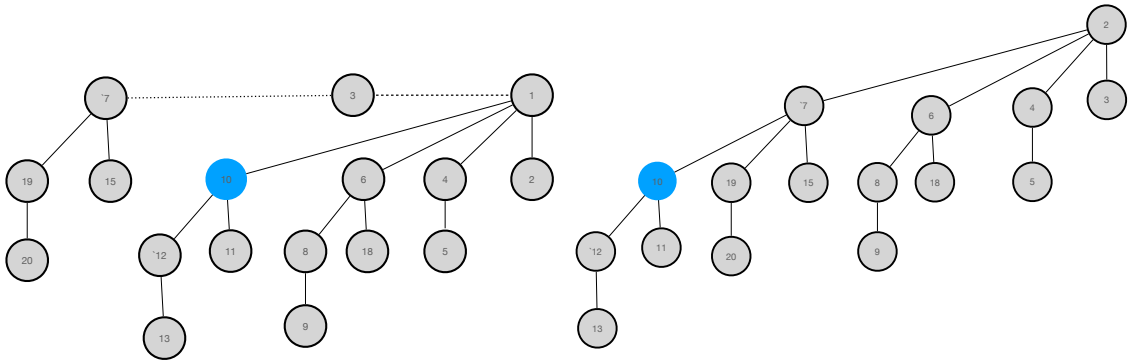
- Put your name on each page; and both your name and BU ID on the first page.
- This exam is open notes, but no consultation with anyone else is permitted.
- This is a take-home exam. You need to turn it into Gradescope by noon on Monday, March 22, 2021. Please don't wait until the last minute to do this, as there will be no late acceptances. Make sure you assign your work to each of the problems.

1. Consider the Fibonacci min-heap illustrated on the right. Show the Fibonacci heap that results after the following operation: Reduce key 14 to 7. Then, show the Fibonacci heap that results when you delete minimum in the resulting heap from the first part.

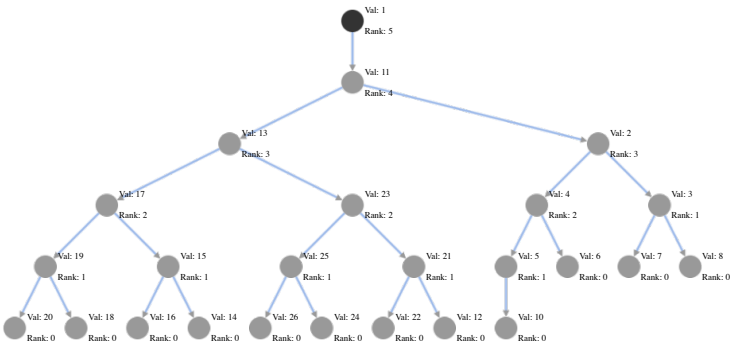


Solution:

The solutions are illustrated below. Decreasing the key from 14 to 7, and then deleting the minimum

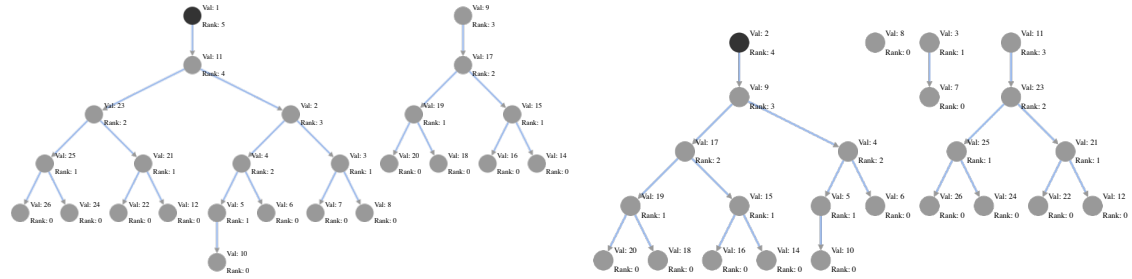


2. Consider the rank-pairing min-heap illustrated on the right. Show the rank-pairing heap, including ranks, that results after the following operation: Reduce key 13 to 9. Then, show the rank-pairing heap that results when you delete minimum in the resulting heap from the first part, using recursive merging, showing the resulting ranks.



Solution:

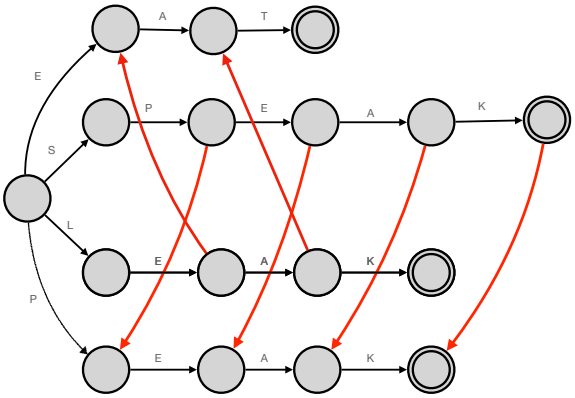
The solutions are illustrated below. Decreasing the key from 13 to 9, and then deleting the minimum.



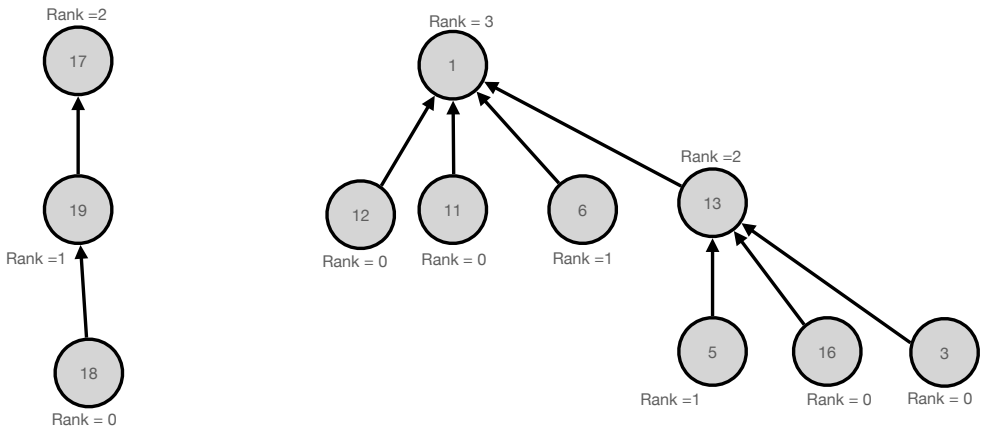
3. Draw the Aho-Corasick trie corresponding to the following search patterns: SPEAK, LEAK, EAT, PEAK. Show the suffix links, except for the suffix links that revert back to the root. Don't bother showing the output links.

Solution:

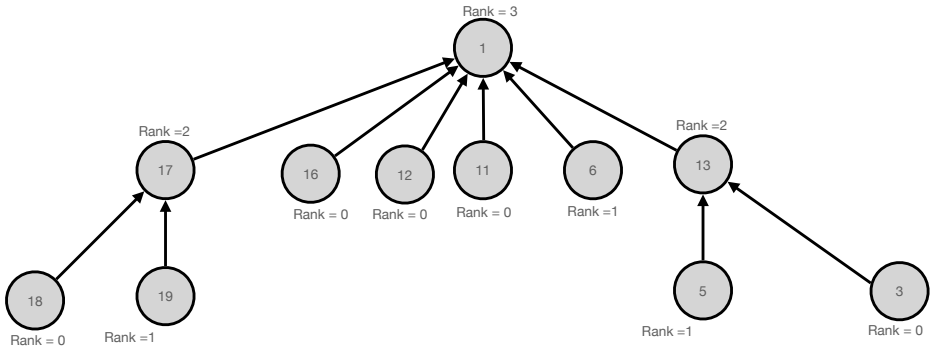
The trie is shown below:

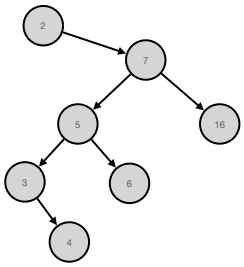


4. Consider the two trees, shown below, that are part of a disjoint set forest. Suppose we find a relation between keys 16 and 18. Show the disjoint set tree that results from the union operation using merge by rank and path compression, where, if two roots have the same rank, merge the larger value root under the smaller value root.



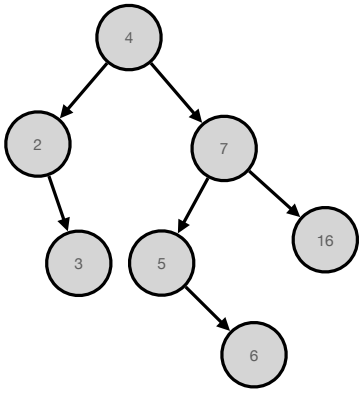
Solution: The merged trees are shown below, using merge-by-rank and path compression.

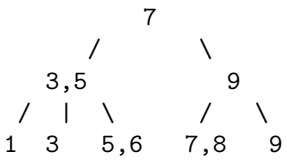




5. Consider the splay tree shown on the right. Show the splay tree that results when you find the key 4 (and splay it to the top using bottom up splaying.)

Solution:

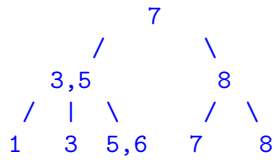




6. Consider the B+ tree of order 3, shown on the right. Show the two B+ trees that result when you delete 9 first, then when you delete 8 subsequently.

Solution:

Delete 9: borrow from sibling



```
Del 8 leads to underflow -- must fix.
            5
          /   \
         3     7
        / \   /  \
       1   3 5,6  7
```

7. For each of these recursions, please give the tightest upper bound for the recursion. You can write your answer as $T(n) \in O(g(n))$ for your best choice of function $g(n)$.

   (a) $T(n) = 8T(n/2) + n^2$
   Solution: $\log_2(8) = 3$, so by the Master theorem, $T(n) \in \Theta(n^3) \in O(n^3)$.

   (b) $T(n) = 3T(n/3) + n$
   Solution: This is the case where $\gamma = \log_3(3) = 1$, and $f(n) \in \Theta(n^\gamma)$. Thus,

   $$T(n) \in \Theta(n(\log n))$$

   (c) $T(n) = 3T(n/5) + \log^2(n)$.
   Solution:
   $\gamma = \log_5(3) < 1$. $f(n) = \log^2(n) \in O(n^{\gamma - epsilon})$ for some small $\epsilon > 0$. Hence, $T(n) \in O(n^{log_5(3)})$