COMP 3804 — Solutions Midterm Winter 2023

Q1: Straightforward induction.

Q2: The Master Theorem cannot be used, because of the logarithm. It is obvious that

$$T(n) = T(n/2) + n \log n \geq n \log n = \Omega(n \log n).$$

For the upper bound, unfolding gives

$$\begin{aligned} T(n) &= \sum_{i=0}^{\log n} \frac{n}{2^i} \log\left(\frac{n}{2^i}\right) \\ &\leq \sum_{i=0}^{\log n} \frac{n}{2^i} \log n \\ &= (n \log n) \sum_{i=0}^{\log n} \frac{1}{2^i} \\ &\leq (n \log n) \sum_{i=0}^{\infty} \frac{1}{2^i} \\ &= 2n \log n \\ &= O(n \log n). \end{aligned}$$

Q3: Note that $1/17 + 16/17 = 1$. Draw the recursion tree and look at the path from the root to the leftmost leaf (always go to the $(1/17)$-child). This path has $\Theta(\log n)$ nodes. All levels from the root to this leftmost leaf are full. Each of them contributes n to $T(n)$.

Q4: For a quick (and informal) argument: On average, the value of k is equal to $n/2$. This gives the recurrence

$$P(n) = P(n/2) + n^2.$$

Now use the Master Theorem.

This can be done in a formal way, as we did in class for randomized selection.

Q5: For $i = 1, 2, \ldots, 27$, let x_i be the $(i \cdot n/27)$-th smallest number in the sequence. If there is a number that occurs at least $n/27$ times, then it must be one of these.

Q6: Since both x and y can be computed in $O(n)$ time, we obtain the recurrence

$$T(n) = 3 \cdot T(n/3) + n.$$

Now use the Master Theorem.

Q7: Draw a heap with 15 nodes, number these nodes from 1 to 15. If you stare at your picture for one minute, you know the answer.

Q8: A max-heap is not a binary search tree.

Q9: In class, we have seen that the time to insert a number into a max-heap is equal to $O(1)$ plus the time for increasekey.

Q10: Use the Master Theorem.

Q11: There is a cycle with 7 vertices.

Q12: If a connected component has a vertices and b edges, then $b \geq a - 1$, because a connected component is connected. This component has a cycle if and only if $b \geq a$.

Q13: See separate file.

Q14: Take the graph with vertices

$$s, a_1, b_1, a_2, b_2, \ldots, a_{(n-2)/2}, b_{(n-2)/2}, t.$$

s has edges to a_1 and b_1. Each of $a_{(n-2)/2}$ and $b_{(n-2)/2}$ has an edge to t. Each a_i has edges to a_{i+1} and b_{i+1}. Each b_i has edges to a_{i+1} and b_{i+1}.

Q15: This was done in the tutorial.

Q16: If the DFS-forest consists of one tree, then there are no cross edges. Still, there can be a directed cycle (that contains a back edge).

Q17: Take the graph with vertices A, B, and C, and edges $\{A, B\}$ of weight 1, $\{B, C\}$ of weight 1, and $\{A, C\}$ of weight 3.