CECS 528, Learning Outcome Assessment 4, Feb 24th, Spring 2023, Dr. Ebert

NO NOTES, BOOKS, ELECTRONIC DEVICES, OR INTERPERSONAL COMMUNICATION ALLOWED. Submit solutions to at most 2 LO problems on separate sheets of paper.

Problems

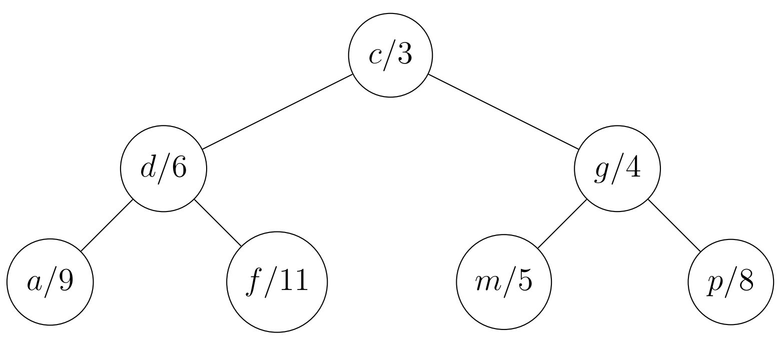
LO1. Which function has a faster big-O growth: $(\log n)^{\sqrt{\log n}}$ or $(\log \log n)^{\log^2 n}$? Defend your answer.

Solution. After applying the log-ratio test we must compare the growth of $\sqrt{\log n}\log(\log n)$ with that of $\log^2 n \log(\log(\log(n)))$. Letting $z = \log n$, this is equivalent to comparing $\sqrt{z}\log z$ with $z^2 \log(\log z)$. By Theorem 2.3.3 and 2.3.4 of the Big-O lecture, we see that $z^2 \log(\log z)$ grows faster. Therefore, $(\log \log n)^{\log^2 n}$ grows faster.

LO2. The tree below shows the state of the binary min-heap at the beginning of some round of Prim's algorithm, applied to some weighted graph G. If G has edges

$$(b, c, 3), (c, e, 7), (c, f, 3), (c, g, 6), (c, p, 2),$$

then draw a plausible state of the heap at the end of the round. (0 points)



LO3. Answer the following with regards to a correctness-proof outline for the Fractional Knapsack algorithm.

(a) Assume $x_1, x_2, \ldots, x_n$ is an ordering of the items in *decreasing* order of profit density (i.e. profit per unit weight). Let f_i denote the fraction of item i that the FK-algorithm adds to the knapsack, $i = 1, 2, \ldots, n$. In relation to the FKA algorithm, why is it the case that $f_1 \geq f_2 \geq \cdots \geq f_n$ is a non-increasing sequence of fractions.

Solution. $f_i \geq f_{i-1}$ since the algorithm always adds as much of an item as possible. Thus, the fraction sequence has the form

$$1, \ldots, 1, f, 0, \ldots, 0,$$

where $f \in [0, 1]$. In other words, all of an item will be added so long as there is enough remaining capacity. This is followed by at most one item for which only a fraction f of the item can be added, meaning the knapsack will result in being filled. Therefore, all subsequent fractions must equal 0.

(b) Let $f'_1, f'_2, \ldots, f'_n$ be a sequence of fractions that optimizes total profit, and assume that $f_i = f'_i$, for all $i < k$, but $f_k \neq f'_k$. Explain why, in this case, it must be true that $f'_k < f_k$. Hint: in relation to the FKA algorithm, what is the contradiction in case the opposite was true?

Solution. From the answer to part a, $f'_k > f_k$ means that the algorithm did not add as much of item x_k as it could have, a contradiction since the algorithm always adds as much of an item as is physically possible.

(c) From part b, the optimal solution uses $(f_k - f'_k)w_k$ less weight of item x_k. What is the most profitable way that the optimal solution can replace this lost weight? Show that the profit made in this case is no greater than the profit that would have been made had it used f_k units of x_k. Conclude that a second optimal solution exists for which $f_i = f'_i$ for all $i = 1, \ldots, k$.

Solution. Since items $x_1, \ldots, x_k$ have all been accounted for, the next most profitable item to add is x_{k+1}. But adding $(f_k - f'_k)w_k$ of this item yields a profit equal to

$$d_{k+1}(f_k - f'_k)w_k \leq d_k(f_k - f'_k)w_k,$$

since the profit density d_{k+1} does not exceed that of d_k. Therefore, by replacing $(f_k - f'_k)w_k$ units of the next most profitable item with $(f_k - f'_k)w_k$ of x_k, the resulting knapsack remains optimal and now agrees with the FKA knapsack up to item x_k. Continuing in this manner we evenutally arrive at and optimal knapsack that entirely agrees with the FKA knapsack.

LO4. Given $T(n) = aT(n/3) + n^4$, for some unknown integer $a \geq 1$, using appropriate big-O notation, provide tight lower and upper bounds for the growth of $T(n)$. Justify your answer.

Solution. By Case 3 of the Master Theorem, we know that $T(n) = \Omega(f(n)) = \Omega(n^4)$. In other words, $T(n)$ cannot grow as $o(n^4)$. By Case 2 of the Master Theorem, if $a = 81$, then $T(n) = \Theta(n^4 \log n)$. Finally, for $a > 81$, Case 1 of the Master Theorem implies $T(n) = \Theta(n^{\log_3 a})$ which grows unboundedly as $a \to \infty$. Therefore $T(n)$ has worst-case polynomial growth with the slowest possible growth being $\Theta(n^4)$.