CECS 528, Learning Outcome Assessment 12a, May 5th, Spring 2023, Dr. Ebert

Problems

LO1. Suppose function $f(n)$ defined as follows.

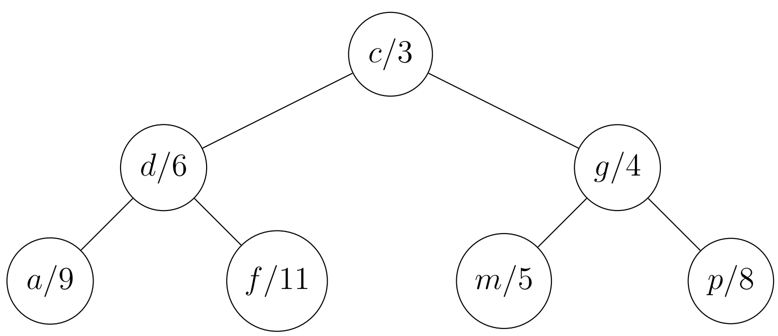
$$f(n) = \begin{cases} 4n^{2.3}\log^3 n & \text{if } n \bmod 3 = 0 \\ 2^{\log^{2.2} n} & \text{if } n \bmod 3 = 1 \\ 10n^{2.2}\log^{35} n & \text{if } n \bmod 3 = 2 \end{cases}$$

Provide a big-O upper bound and big-Ω lower bound for $f(n)$.

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

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

then draw a plausible state of the heap at the end of the round.



LO3. In the correctness proof of Prim's algorithm, suppose $T = e_1, \ldots, e_{n-1}$ are the edges selected by Prim's algorithm (in that order) and T_{opt} is an mst that uses edges $e_1, \ldots, e_{k-1}$, but for which $e_k \notin T_{\text{opt}}$. Explain how to identify an edge $e \in T_{\text{opt}}$ for which $T_{\text{opt}} + e_k - e$ is an mst that now possesses edges $e_1, \ldots, e_k$. Hint: consider tree T_{k-1} which is Prim's tree after having selected e_{k-1} and is thus a subtree of T_{opt}.

LO4. Given the recurrence $T(n) = 2T(n/2) + n \log n$, use the substitution method to prove $T(n) = \Omega(n \log^2 n)$. Remember to state the inductive assumption.

LO5. At the top level of recursion for the Median-of-Five Find Statistic algorithm, with

$$a = 84, 38, 54, 60, 10, 74, 22, 75, 57, 19, 27, 64, 35, 3, 80, 68, 72, 13, 87, 36, 6, 5$$

and $k = 10$ as inputs,

(a) What number will be chosen as the pivot for the partitioning step? Show work.

(b) At the next level (1) of recursion, which array will be examined: a_{left}? a_{right}? neither? Explain.

LO6. Given that $r = ae + bg$, $s = af + bh$, $t = ce + dg$, and $u = cf + dh$ are the four entries of AB, and Strassen's products are obtained from matrices

$$A_1 = a, B_1 = f - h, A_2 = a + b, B_2 = h, A_3 = c + d, B_3 = e, A_4 = d, B_4 = g - e,$$

$$A_5 = a + d, B_5 = e + h, A_6 = b - d, B_6 = g + h, A_7 = a - c, B_7 = e + f,$$

Compute $P_1, \ldots, P_7$ and use them to compute r, s, t, and u.

LO7. Answer/Solve the following questions/problems.

(a) The dynamic-programming algorithm that solves the `Matrix-Chain Multiplication` optimization problem defines a recurrence for the function $\mathrm{mc}(i, j)$. In words, what does $\mathrm{mc}(i, j)$ equal? Hint: do *not* write the recurrence (see Part b).

(b) Provide the dynamic-programming recurrence for $\mathrm{mc}(i, j)$.

(c) Apply the recurrence from Part b to the dimension sequence 3,2,6,2,4. Show the matrix of subproblem solutions and use it to provide an optimal parenthesization.

LO8. Answer the following.

(a) Provide the dynamic-programming recurrence for computing $\mathrm{mc}(u, v)$ the maximum-cost of any path from vertex u to vertex v in a directed acyclic graph (DAG) $G = (V, E, c)$, where $c(e)$ gives the cost of edge e, for each $e \in E$. Hint: **credit will not be awarded for using $d(u, v)$ instead of $\mathrm{mc}(u, v)$.**

(b) Draw the vertices of the following DAG G in a linear left-to-right manner so that the vertices are topologically sorted, meaning, if (u, v) is an edge of G, then u appears to the left of v. The vertices of G are a-h, while the weighted edges of G are

$$(a, b, 1), (a, e, 13), (a, f, 12), (b, c, 16), (b, g, 15), (c, d, 2), (c, g, 16), (c, h, 18), (d, h, 15), (e, b, 6)$$

$$(e, f, 19), (f, b, 19), (f, c, 7), (f, g, 4), (g, d, 15), (g, h, 12).$$

(c) Starting with $u = h$, and working backwards (from right to left in the topological sort), use the recurrence from part a to compute $\mathrm{mc}(u, h)$ for each $u \in \{a, b, \ldots, h\}$, where the ultimate goal is to compute $d(a, h)$.

LO9. Consider the `2SAT` instance

$$\mathcal{C} = \{(x_1, \overline{x}_2), (x_1, \overline{x}_3), (x_1, \overline{x}_4), (\overline{x}_1, x_3), (x_2, x_4), (\overline{x}_2, x_3)(\overline{x}_3, \overline{x}_4)\}.$$

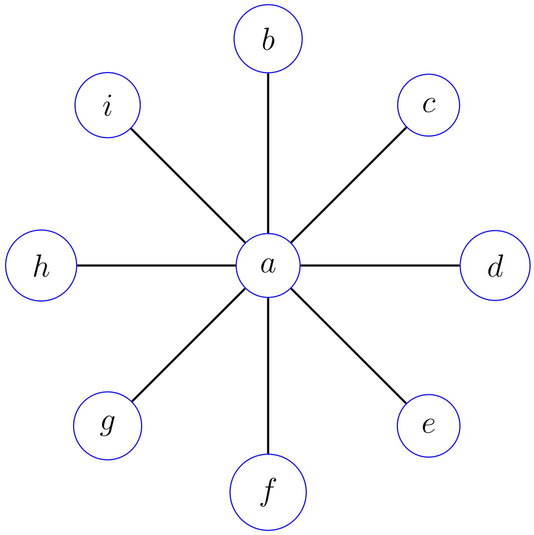
(a) Draw the implication graph $G_{\mathcal{C}}$.

(b) Find a literal l for which i) R_l is an inconsistent reachability set, ii) $R_{\overline{l}}$ is a consistent reachability set, and iii) $\alpha_{R_{\overline{l}}}$ satisfies *all* the clauses of $\mathcal{C}$. For full credit clearly state the literal l you have chosen and verify that each of the three properties are satisfied. Hint: for example, if you choose $l = \overline{x}_3$, then $\overline{l} = \overline{\overline{x}}_3 = x_3$.

LO10. Answer the following.

(a) Provide the definition of what it means to be a mapping reduction from decision problem A to decision problem B.

(b) For the mapping reduction f : `Clique` $\rightarrow$ `Half Clique` provided in Exercises 11 and 12, and for `Clique` instance $(G, k = 3)$, draw $f(G, k)$, where G is shown below.



(c) Verify that (G, k) and $f(G, k)$ are either both positive instances or both negative instances of their respective decision problems. Defend your answer.

LO11. Recall the mapping reduction $f(\mathcal{C}) = (G, k)$, where f maps an instance of `3SAT` to an instance of the `Clique` decision problem. Given `3SAT` instance

$$\mathcal{C} = \{(x_1, \overline{x}_2, \overline{x}_2), (x_1, x_2, x_2), (\overline{x}_1, x_2, x_2), (\overline{x}_1, \overline{x}_2, \overline{x}_2)\}$$

answer the following questions about $f(\mathcal{C})$. Hint: to answer these questions you do *not* need to draw G.

(a) How many vertices does G have? Justify your answer.

(b) How many edges does G have? Show work and justify your answer.

(c) Verify that $\mathcal{C}$ and $f(\mathcal{C}) = (G, k)$ are either both positive instances or both negative instances of their respective decision problems. Defend your answer.