CECS 528, Exam 2, Pink, Fall 2023, Dr. Ebert

NO NOTES, BOOKS, ELECTRONIC DEVICES, OR INTERPERSONAL COMMUNICATION ALLOWED. Submit **AT MOST SIX** solutions. Make sure your name and SID are on each answer sheet and please **USE BOTH SIDES** of each answer sheet to save paper.

Problems (25 Points Each)

1. Answer the following with regards to a correctness-proof outline for Dijkstra's algorithm. Note: correctly solving this problem counts for passing LO5.

   (a) In relation to Dijkstra's algorithm, provide a definition for what it means to be i) an i-neighboring path from source s to an external vertex v, and ii) the i-neighboring distance $d_i(s, v)$ from source s to external vertex v. Hint: at this point in the algorithm i nodes have been added to the DDT. (8 points)

   (b) Using the definitions from part a, describe the greedy choice that is made in each round of Dijkstra's algorithm. (7 points)

   (c) If vertex v is chosen by Dijkstra in Round $i+1$, use part b to prove that $d(s, v) = d_i(s, v)$. Hint: if i-neighboring path P from s to v has cost $d_i(s, v)$ and Q is any other path from s to v, explain why $\text{cost}(Q) \geq d_i(s, v)$. (10 points)

2. Do the following. Note: correctly solving this problem counts for passing LO7.

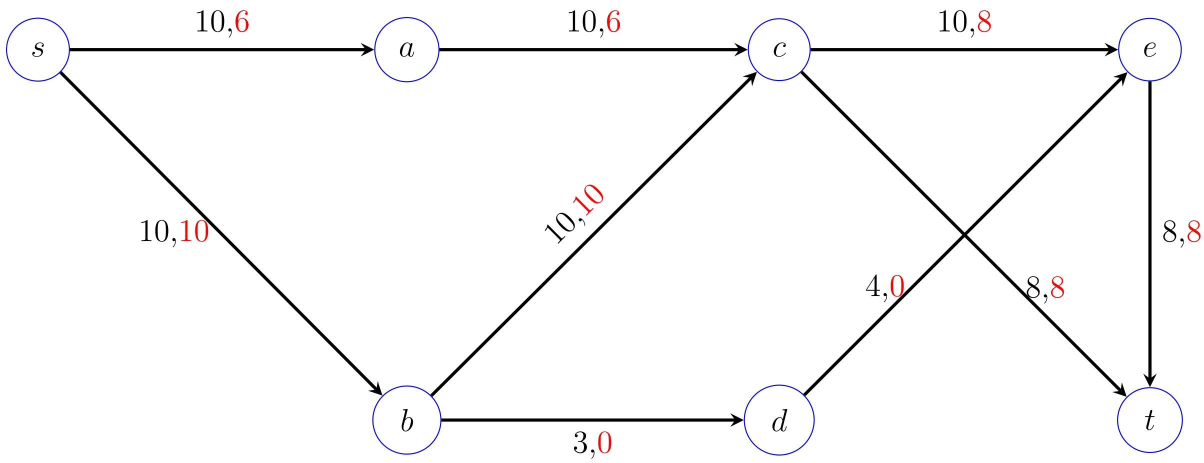
   (a) The dynamic-programming algorithm that solves the `Edit Distance` optimization problem defines a recurrence for the function $d(i, j)$. In words, what does $d(i, j)$ equal? Hint: do *not* write the recurrence (see Part b). (5 pts)

   (b) Provide the dynamic-programming recurrence for $d(i, j)$. (5 pts)

   (c) Apply the recurrence from Part b to the words $u =$ cbbcca and $v =$ bccaba. Show the matrix of subproblem solutions and use it to provide an optimal sequence of left-to-right edits. (15 pts)

3. A flow f (2nd value listed on each edge) has been placed in the network G below. Note: correctly solving this problem counts for passing LO8.

   (a) Draw the residual network G_f and use it to determine an augmenting path P. Highlight path P in the network so that it is clearly visible. (10 pts)



(b) In the original network, cross out any flow value that changed, and replace it with its updated value from $f_2 = \Delta(f, P)$. (7 pts)

(c) What one query can be made to a `Reachability` oracle to determine if f_2 is a maximum flow for G? Hint: three inputs are needed for the `reachable` query function. Cleary define each of them. (8 pts)

4. Recall that an instance of the `Unit Task Scheduling` optimization is a set of triples

$$\{(a_1, d_1, p_1), \ldots, (a_n, d_n, p_n)\},$$

where, for each $i = 1, \ldots, n$, a_i is a task identifier, d_i is the task's deadline, and p_i is the profit obtained if a_i is schedule at or before its deadline. We assume each task requires one unit of time to process. The problem is to construct a schedule such that i) at each time slot t there is *at most* one task a_i that is scheduled at that slot and $t \leq d_i$, and ii) the sum of the profits of all the tasks in the schedule is a maximum among all possible schedules. We may use the following greedy algorithm for finding an optimal schedule. For each time slot $t = 1, \ldots, m$, let L_t denote the sorted (by decreasing profit) list of tasks, each having deadline equal to t. Then for each $t = m, m-1, \ldots, 1$, if $L_t \neq \emptyset$, then remove task a from the front of L_t and schedule a at time t. If $t > 1$, then update L_{t-1} by merging it with the rest of L_t. Otherwise, stop: the schedule is complete. Answer the following. Note this problem counts for passing LO5.

(a) Let S be the schedule returned by the algorithm, and S_{opt} be an optimal schedule. Moreover, suppose S and S_{opt} agree at time slots $m, m-1, \ldots, k+1$, for some $k \geq 1$, but they disagree at time slot k in that S schedules task a at time k, but S_{opt} does not schedule a at any time slot. Why must S_{opt} schedule a task $a' \neq a$ at time k? (8 pts)

(b) Explain why it must be the case that $p \geq p'$, where p is a's profit and p' is a''s profit. Conclude that we may replace a' at time t with a to obtain an optimal schedule that agrees with S from time m down to time k. (17 pts)

5. Suppose n words $w_1, \ldots, w_n$ are to be typed in left-to-right order on a page for which each line can hold up to M characters, where $M > 0$ is a constant. Furthermore, each word w_i has length l_i characters and must appear on a single line. The goal is to space the words so as to avoid having lines with lots of whitespace. To accomplish this, each line incurs a penalty of W^2, where W is the number of whitespace characters that appear in the line. Note that it is not necessary to place whitespace between two consecutive words. For example, if $M = 10$ and

three words having respective lengths of 3, 2, and 3 are printed on a line, then the whitespace penalty for that line is $(10 - 3 - 2 - 3)^2 = 4$. Moreover, the goal is to minimize the total whitespace penalty

$$P = \sum_{j=1}^{m} W_j^2,$$

assuming that the words are typed on m lines, and W_j^2 is the penalty incurred by line j. Note: it is helpful to define $W_j^2 = \infty$ in case the words that are chosen to be typed on line j have a total number of characters that exceeds M.

(a) If $p(i)$ represents the minimum whitespace penalty when having to type words $w_1, \ldots, w_i$, then provide a dynamic-programming recurrence for $p(i)$. Hint: imagine taking a first step towards solving this problem by deciding on which words to type on the final line. (18 points)

(b) Apply your recurrence from part a to a sequence of seven words whose respective lengths are $3, 3, 4, 3, 5, 2, 2$, and each line has capacity $M = 10$. Provide the array of subproblem solutions, along with the optimal placement of the words. (7 points)

6. Eswar's logistics project calls for finding minimum spanning trees for weighted graphs whose edges sometimes have negative weights.

   (a) Based on how the correctness of Prim's algorithm is proved, explain why the algorithm correctly computes mst's for graphs that may have one or more negative edge weights. (12 pts)

   (b) Despite part a, the program Eswar has decided to use only accepts graphs with nonnegative edge weights. Explain how Eswar can work around this constraint and still use the algorithm. How should he modify his graph to conform to the non-negative-weight constraint? How must he adjust the answer returned by the program (in terms of the sum-of-weights cost of the mst) in order to obtain the correct answer for his original graph? Assume his graph has n vertices and m edges. (13 pts)

LO Makeup Problems (0 Points Each)

LO1. Solve the following problems.

(a) Compute the multiplicative inverse of 37 modulo 86.

(b) For the Strassen-Solovay primality test, verify that $a = 2$ an accomplice to $n = 5$ being a prime number. Show all work.

LO2. Solve the following problems.

(a) Use the Master Theorem to determine the growth of $T(n)$ if it satisfies the recurrence $T(n) = 16T(n/4) + n^{\log_3 9}$. Defend your answer.

(b) Use the substitution method to prove that, if $T(n)$ satisfies

$$T(n) = 2T(n/2) + n^{1.5},$$

Then $T(n) = \mathrm{O}(n^{1.5})$.

LO3. Solve the following problems.

(a) 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.

(b) Draw the recursion tree that results when applying Mergesort to the array

$$a = 5, 4, 12, 8, 7, 11, 13, 9, 10, 16,$$

Label each node with the sub-problem to be solved at that point of the recursion. Assume arrays of size 1 and 2 are base cases. Assume that odd-sized arrays are split so that the left subproblem has one more integer than the right. Next to each node, write the solution to its associated subproblem.

LO4. Answer/solve the following.

(a) The FFT algorithm owes its existence to what two properties that are possessed by the nth roots of unity when n is even?

(b) Compute $\mathrm{DFT}_4(5, -4, 3, -2)$ using the FFT algorithm. Show the solution to each of the seven subproblem instances and, for each one, clearly represent it using DFT notation and apply the formula for computing it. Show all work.

LO6. For the weighted graph with edges

$$(b, d, 5), (a, e, 4), (a, b, 1), (e, c, 3), (e, f, 6), (e, d, 2),$$

Show how the membership-tree forest (not the Kruskal forest!) changes when processing each edge in the Kruskal sorted order when performing Kruskal's algorithm. When merging two trees, use the convention that the root of the merged tree should be the one having *lower* alphabetical order. For example, if two trees, one with root a, the other with root b, are merged, then the merged tree should have root a.