CECS 528, Quiz 2, Spring 2026, Dr. Ebert

IMPORTANT: READ THE FOLLOWING DIRECTIONS.

- For each problem, write your solution using **ONE SHEET OF PAPER ONLY (BOTH FRONT AND BACK)**. Write **NAME** and **PROBLEM NUMBER** on each sheet.
- Write solutions to different problems on **SEPARATE SHEETS** of paper.
- For solving makeup problems there are two options: **SOLVE ONE ENTIRE LO PROBLEM** OR (exclusive) **SOLVE ONE PART EACH OF TWO DIFFERENT PROBLEMS**.

Unit 2 LO Problems

LO5. Do the following.

(a) The dynamic-programming algorithm that solves the `0-1 Knapsack` problem defines a function $p(i, c)$. Describe in one sentence the definition of $p(i, c)$ and give the ranges for i and c. Do *not* provide a recurrence (part b will ask for this).

(b) Provide the dynamic-programming recurrence for $p(i, c)$.

(c) Apply the recurrence from Part b to the items $1-6$ whose respective weights are $3, 3, 4, 2, 5, 2$ and respective profits are $20, 50, 30, 10, 20, 50$. Assume a knapsack capacity equal to $M = 10$. Provide the optimal set of items.

LO6. Solve the following.

(a) Consider the `2SAT` instance

$$\mathcal{C} = \{(x_1, \overline{x}_2), (\overline{x}_1, \overline{x}_3), (\overline{x}_1, \overline{x}_4), (\overline{x}_1, x_6), (x_2, x_4), (x_3, \overline{x}_6), (\overline{x}_3, x_5), (\overline{x}_5, x_6), (\overline{x}_5, \overline{x}_6)\}.$$

i. Draw the implication graph $G_{\mathcal{C}}$.

ii. Perform the `Improved 2SAT` algorithm by computing the necessary reachability sets. Use numerical order (in terms of the variable index) and positive literal before negative literal when choosing the reachability set to compute next. Draw the resulting reduced `2SAT` instance whenever a consistent reachability set is computed. Either provide a final satisfying assignment for $\mathcal{C}$ or indicate why $\mathcal{C}$ is unsatisfiable.

(b) Given `2SAT` instance $\mathcal{C}$, if the reachability set $R(x_1)$ is consistent and contains the literal x_4, then provide the reasoning for why the assignment $\alpha_{R_{x_1}}$ induced by $R(x_1)$ satisfies the clause $(x_3, \overline{x}_4)$.

Makeup Problems

LO1. Solve the following problems.

(a) Use the Master Theorem to determine the growth of $T(n)$ if it satisfies the recurrence $T(n) = 4T(n/3) + n^{\frac{3}{2}}$. Defend your answer.

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

$$T(n) = 4T(n/2) + \sqrt{n},$$

then $T(n) = \mathrm{O}(n^2)$.

LO2. Solve the following problems.

(a) Use Strassen's products $P_1 = a(f-h) = af - ah$, $P_2 = (a+b)h = ah + bh$, $P_3 = (c+d)e = ce + de$, $P_4 = d(g-e) = dg - de$, $P_5 = (a+d)(e+h) = ae + ah + de + dh$, $P_6 = (b-d)(g+h) = bg + bh - dg - dh$, $P_7 = (a-c)(e+f) = ae - ce - cf + af$. to compute the matrix product

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} e & f \\ g & h \end{pmatrix}$$

assuming that the product equals the matrix

$$\begin{pmatrix} r & s \\ t & u \end{pmatrix}.$$

(b) Consider Karatsuba's algorithm which we'll call `multiply` for multiplying two n-bit binary numbers x and y, where we assume that n is even. Let x_L and x_R be the leftmost $n/2$ and rightmost $n/2$ bits of x respectively. Define y_L and y_R similarly. Let P_1 be the result of calling `multiply` on inputs x_L and y_L, P_2 be the result of calling `multiply` on inputs x_R and y_R, and P_3 the result of calling `multiply` on inputs $x_L + x_R$ and $y_L + y_R$. Then return the value $P_1 \times 2^n + (P_3 - P_1 - P_2) \times 2^{\frac{n}{2}} + P_2$. Demonstrate the algorithm (only at the root level of the recursion tree!) with $x = 43$ and $y = 13$. Hint: convert x and y to two *binary* numbers, each having the same (even) number of bits. However, for the sake of readability and for easier computation, when evaluating P_1, P_2, and P_3 it is OK to express those answers in decimal.

LO3. Do the following.

(a) Consider the FFT algorithm when applied to a polynomial $A(x)$ having degree $2^n - 1$. Provide the equation that relates $A(x)$ to the two subproblem polynomials $A_e(x)$ and $A_o(x)$. What are the degrees of these two polynomials? Based on this equation, why is it essential that, for even n, the nth roots of unity come in additive-inverse pairs?

(b) If $p(x) = -2 + 3x + x^2 - 5x^3$, then compute $\mathrm{DFT}^{-1}(p)$ using the FFT algorithm. Show the entire recursion tree as was done in the lecture notes. (10 points)

LO4. Do the following.

(a) Recall the use of the disjoint-set data structure for the purpose of improving the running time of the `Unit Task Scheduling` algorithm. For the set of tasks

Task	a	b	c	d	e	f
Deadline Index	4	4	4	5	4	3
Profit	60	50	40	30	20	10

For *each* task, show the MTree forest after it has been inserted (or at least has attempted to be inserted in case the scheduling array is full). Note that the earliest possible deadline index is 1, meaning that the earliest slot in the schedule array has index 1. To receive credit, your solution should show six different snapshots of the M-Tree forest.

(b) Do the following.

i. State the greedy choice that is being made in each round of Prim's algorithm.

ii. The weighted edges of a graph $G = (V, E)$ are

$$E = \{(1,2,11),(1,3,19),(1,4,15),(1,5,13),(1,6,8),(2,3,14),(2,4,17),$$

$$(2,5,16),(2,6,22),(3,4,10),(3,5,12),(3,6,19),(4,5,15),(5,6,20)\}.$$

For each round of Prim's algorithm applied to G, indicate the selection for that round and provide a drawing of the final output of the algorithm. Hint: you do *not* need to use a heap data structure. Also, break any ties by choosing the vertex having least index. For example, if there is a tie beetween vertices 2 and 4, then choose vertex 2.