CECS 528, Learning Outcome Assessment 6, Yellow, Fall 2023, Dr. Ebert

NO NOTES, BOOKS, ELECTRONIC DEVICES, OR INTERPERSONAL COMMUNICATION ALLOWED. Submit each solution on a separate sheet of paper.

Problems

LO2. Solve the following problems.

(a) Use the Master Theorem to determine the growth of $T(n)$ if it satisfies the recurrence $T(n) = 4T(n/2) + n^{\log_4 16}$. Defend your answer.

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

$$T(n) = 4T(n/2) + 6n$$

then $T(n) = \Omega(n^2)$.

LO3. Solve the following problems.

(a) Recall the combine step of the `Minimum Distance Pair (MDP)` algorithm where, for each point P in the δ-strip, there is a $2\delta \times \delta$ rectangle whose bottom side contains P and is bisected by the vertical line that divides the points into left and right subsets. Explain why there can be at most 7 other points (from the problem instance) in this rectangle.

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

$$\begin{pmatrix} 4 & -1 \\ -3 & 6 \end{pmatrix} \begin{pmatrix} -3 & 4 \\ 2 & -5 \end{pmatrix}$$

Show all work. (13 pts)

LO4. Answer the following.

(a) Write each of the 3rd roots of unity in the form $a + bi$, where a and b are real numbers.

(b) Compute $\text{DFT}_4(4, -3, 2, -1)$ 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.

LO5. Answer the following with regards to a correctness-proof outline for the `Fuel Reloading` algorithm.

(a) Let $S = s_1 < s_2 < \cdots < s_m$ denote the stations selected by the algorithm. Assume S_{opt} is an optimal solution that contains stations $s_1, \ldots, s_{k-1}$, but does not contain station s_k. Explain why S_{opt} must have at least one station s for which $s > s_{k-1}$. Note: each station is identified with its location on the number line. Hint: what contradiction arises if there is no such station.

(b) Let s be the least (most left on the number line) station in S_{opt} for which $s > s_{k-1}$. Explain why we must have $s < s_k$. Note: for any two stations s_1 and s_2, we assume that either $s_1 < s_2$ or $s_2 < s_1$ is true.

(c) Explain why replacing s with s_k in S_{opt} represents a valid set of stations in the sense that the car will never run out of fuel. Hint: discuss the next station (or final destination if no next station exists) that comes after s in S_{opt}.

LO6. 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	3	3	4	3
Profit	60	50	40	30	20	10

For each task, show the M-Tree 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 0, meaning that the earliest slot in the schedule array has index 0. Also, assume that an insert attempt that takes place at index i results in the function call `root`(i). Finally, to receive credit, your solution should show six different snapshots of the M-Tree forest.

Solutions

LO2. Solve the following problems.

(a) Use the Master Theorem to determine the growth of $T(n)$ if it satisfies the recurrence $T(n) = 4T(n/2) + n^{\log_4 16}$. Defend your answer.

Solution. Since $\log_2 4 = \log_4 16 = 2$, it follows by Case 2 of the Master Theorem that $T(n) = \Theta(n^2 \log n)$.

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

$$T(n) = 4T(n/2) + 6n$$

then $T(n) = \Omega(n^2)$.

Solution. Inductive Assumption: $T(k) \geq Ck^2$ for all $k < n$. Show $T(n) \geq Cn^2$. We have

$$T(n) = 4T(n/2) + 6n \geq 4C(\frac{n}{2})^2 + 6n = Cn^2 + 6n \geq Cn^2 \iff 6n \geq 0$$

which is true for $n \geq 1$. Therefore, $T(n) = \Omega(n^2)$.

LO3. Solve the following problems.

(a) Recall the combine step of the `Minimum Distance Pair (MDP)` algorithm where, for each point P in the δ-strip, there is a $2\delta \times \delta$ rectangle whose bottom side contains P and is bisected by the vertical line that divides the points into left and right subsets. Explain why there can be at most 7 other points (from the problem instance) in this rectangle.

Solution. The $2\delta \times \delta$ rectangle consists of two $\delta \times \delta$ squares, one on each side of the problem-instance dividing line. Moreover, we know that any two points in the data set that are both on one side of the dividing line are at least δ away from each other. Therefore, in a $\delta \times \delta$ square there can be at most 4 points in the data set within that square. And so the $2\delta \times \delta$ rectangle can contain at most 4+4 = 8 data points, one of which is point P. That leaves at most 7 other points in the rectangle.

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

$$\begin{pmatrix} 4 & -1 \\ -3 & 6 \end{pmatrix} \begin{pmatrix} -3 & 4 \\ 2 & -5 \end{pmatrix}$$

Show all work. (13 pts)

Solution. We have

$$ae + bg = P_5 + P_6 - P_2 + P_4 = -80 + 21 - (-15) + 30 = -14,$$

$$af + bh = P_1 + P_2 = 36 + (-15) = 21,$$

$$ce + dg = P_3 + P_4 = -9 + 30 = 21,$$

and

$$cf + dh = -P_7 + P_5 + P_1 - P_3 = -7 + (-80) + 36 - (-9) = -42,$$

to yield the product matrix

$$\begin{pmatrix} -14 & 21 \\ 21 & -42 \end{pmatrix}.$$

LO4. Answer the following.

(a) Write each of the 3rd roots of unity in the form $a + bi$, where a and b are real numbers.

Solution. $w_3^0 = 1$, $w_3^1 = -\frac{1}{2} + \frac{\sqrt{3}}{2}i$, and $w_3^2 = -\frac{1}{2} - \frac{\sqrt{3}}{2}i$.

(b) Compute $\mathrm{DFT}_4(4, -3, 2, -1)$ 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.

Solution.
We have

$$\mathrm{DFT}_1(4) = 4 \text{ and } \mathrm{DFT}_1(2) = 2.$$

$$\mathrm{DFT}_2(4, 2) = (4, 4) + (1, -1) \odot (2, 2) = (6, 2).$$

$$\mathrm{DFT}_1(-3) = -3 \text{ and } \mathrm{DFT}_1(-1) = -1.$$

$$\mathrm{DFT}_2(-3, -1) = (-3, -3) + (1, -1) \odot (-1, -1) = (-4, -2).$$

$$\mathrm{DFT}_4(4, -3, 2, -1) = (6, 2, 6, 2) + (1, i, -1, -i) \odot (-4, -2, -4, -2) = (2, 2 - 2i, 10, 2 + 2i).$$

LO5. Answer the following with regards to a correctness-proof outline for the `Fuel Reloading` algorithm.

(a) Let $S = s_1 < s_2 < \cdots < s_m$ denote the stations selected by the algorithm. Assume S_{opt} is an optimal solution that contains stations $s_1, \ldots, s_{k-1}$, but does not contain station s_k. Explain why S_{opt} must have at least one station s for which $s > s_{k-1}$. Note: each station is identified with its location on the number line. Hint: what contradiction arises if there is no such station.

Solution. Otherwise, the traveler could reach the final destination from s_{k-1} without refueling, which contradicts the algorithm's calculated need for refueling at s_k.

(b) Let s be the least (most left on the number line) station in S_{opt} for which $s > s_{k-1}$. Explain why we must have $s < s_k$. Note: for any two stations s_1 and s_2, we assume that either $s_1 < s_2$ or $s_2 < s_1$ is true.

Solution. If $s > s_k$, then s is reachable from s_{k-1} and is further away from s_{k-1} than is s_k which implies the algorithm would have selected s over s_k, a contradiction.

(c) Explain why replacing s with s_k in S_{opt} represents a valid set of stations in the sense that the car will never run out of fuel. Hint: discuss the next station (or final destination if no next station exists) that comes after s in S_{opt}.

Solution. Case 1: s is the final station of S_{opt}. Then since $s_k > s$, after refueling at s_k the traveler will still be able to reach the final destination.

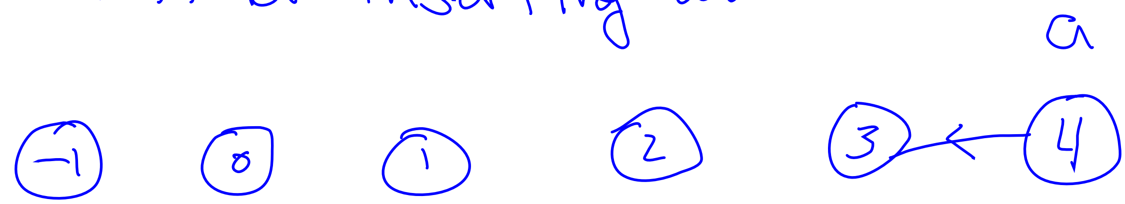
Case 2: $s' \in S_{\text{opt}}$ immediately follows s in the solution. Then since $s_k > s$, after refueling at s_k the traveler will still be able to reach s'. Therefore, $S_{\text{opt}} - s + s_k$ remains a valid and optimal set of stations that now contains tasks $s_1, \ldots, s_{k-1}, s_k$.

LO6. 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

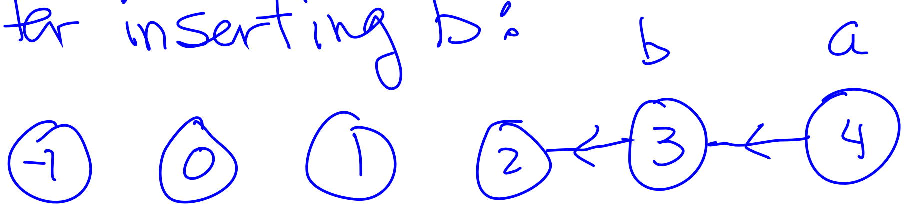
Task	a	b	c	d	e	f
Deadline Index	4	4	3	3	4	3
Profit	60	50	40	30	20	10

For each task, show the M-Tree 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 0, meaning that the earliest slot in the schedule array has index 0. Also, assume that an insert attempt that takes place at index i results in the function call `root`(i). Finally, to receive credit, your solution should show six different snapshots of the M-Tree forest.

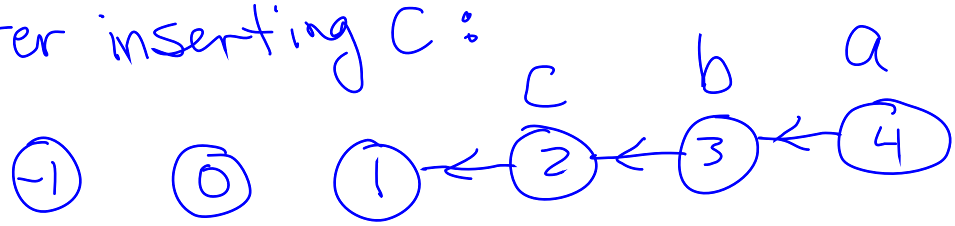
Solution. 1. After inserting a:



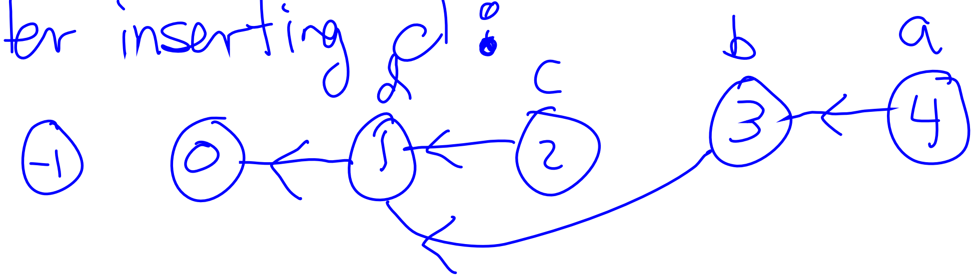
2. After inserting b:



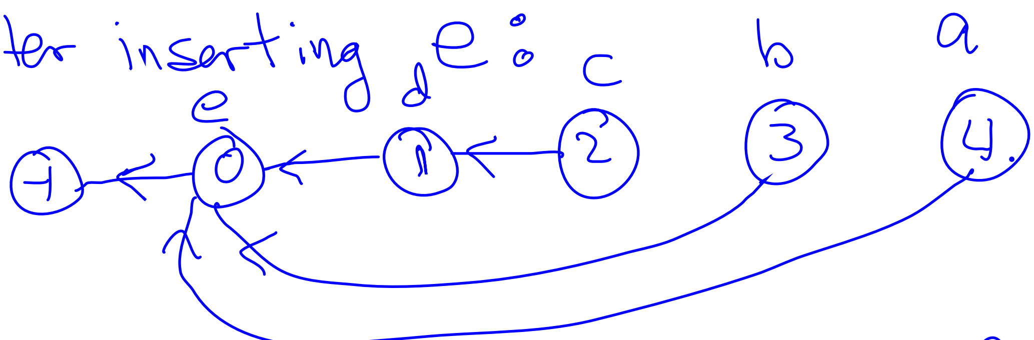
3. After inserting c:



4. After inserting d:



5. After inserting e:



6. After unsuccessfully inserting f: