CS 456: Advanced Algorithms

Homework #02

Assigned Date	:	Friday, October 03, 2014
Due Date	:	Tuesday, October 14, 2014 @ 12:29:59 p.m.

Instructions

- This is an individual assignment. **Do your own work**. Acts of academic misconduct (plagiarism, use of illegal solutions manuals, etc.) will be strictly monitored and will be subjected to one or more of the penalties outlined in the course syllabus.
- Your answers should be produced using a word processing application.
- Handover a **printed**, **stapled** copy of your solutions to the instructor at the beginning of class on due date. Make sure to include your name and the last 3 digits of your SIUE ID in the first page of your solutions sheet.
- A moodle dropbox will become available the day before the due date as a secondary submission option. Ensure to submit a PDF document if you decide to use the moodle dropbox. **DO NOT** email your solutions to the instructor.
- Make proper arrangements, after consulting the instructor, to deliver your solutions **BEFORE** the due date, if you have a planned absence on the due date.
- Answer all questions
- Your assignment is due on **Tuesday, October 14, 2014 @ 12:29:59 p.m.**
- Total points: [**200 points**] + Extra Credit: [**120 points**]

Questions

Q1. Find the order of growth for the following recurrences. Assume that $T(n)$ is constant for $n \leq 2$. Justify your answers [**50 points**]

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

(b) $T(n) = 2T(n/2) + n^4$

(c) $T(n) = 16T(n/4) + n^2$

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

(e) $T(n) = T(n-2) + n^2$

Q2. Verify the formulas underlying Strassen's algorithm for multiplying 2×2 matrices [**20 points**]

Q3. Apply Strassen's algorithm to compute [**20 points**] $\begin{bmatrix} 3 & 5 & 1 & 9 \\ 4 & 1 & 1 & 0 \\ 1 & 0 & 2 & 0 \\ 0 & 4 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} 4 & 2 & 0 & 1 \\ 2 & 0 & 0 & 1 \\ 1 & 7 & 1 & 5 \\ 2 & 2 & 0 & 1 \end{bmatrix}$ by exiting the recursion when $n = 2$.

Q4. There are n people, each in possession of a different rumor. They want ot share all the rumors with each other by sending electronic messages. Assume that a sender includes all the rumors he or she knowns at the time the messages is sent and a message may only have one addressee. Design a **greedy algorithm** that always yields the minimum number of messages they need to send to guarantee that every one of them gets all the rumors [**35 points**]

Q5. (a) Construct a Huffman code for the following data: [**20 points**]

symbol	A	B	C	D	-
frequency	0.4	0.3	0.2	0.025	0.075

(b) Encode BADABBADEAD [**10 points**]

(c) Decode 1100011001001011101100 [**10 points**]

Q6. Draw the recursion tree for $T(n) = 4T(\lfloor \frac{n}{2} \rfloor) + cn$, where c is a constant, and provide a tight asymptotic bound on its solution. Verify your answer using the Master's Theorem. [**20 points**]

Q7. Use the master's theorem to show that the solution to binary-search recurrence $T(n) = T(n/2) + \Theta(n)$ is $\Theta(log_2 n)$ [**15 points**]

Extra Credit

Q8. Give asymptotic upper and lower bounds for $T(n)$ in each of the following recurrences. Assume that $T(n)$ is constant for sufficiently small n. Make your bounds as tight as possible, and justify your answers [**50 points**]

(a) $T(n) = 4T(n/3) + nlog_2 n$

(b) $T(n) = T(n/2) + T(n/4) + T(n/8) + n$

(c) $T(n) = T(n-1) + 1/n$

(d) $T(n) = \sqrt{n} 2T(\sqrt{n}) + n$

(e) $T(n) = 3T(n/3 - 2) + n/2$

Q9. Write a pseudocode of the **greedy algorithm** for a the change-making problem, with an amount n and coin denominations $d_1 > d_2 > \dots d_m$ as its input. What is the time efficiency class of your algorithm? [**50 points**]

Q10. Draw the recursion tree for $T(n) = 3T(\frac{n}{4}) + n \cdot log_2 n$, where c is a constant, and provide a tight asymptotic bound on its solution. Verify your answer using the Master's Theorem. [**20 points**]