COMP 251 - Fall 2017 - Assignment 4

Due: 11:59pm Nov 18th

General rules: In solving these questions you may consult your book; You can discuss high level ideas with each other, but each student must find and write his/her own solution. You should upload the pdf file (either typed, or a clear scan) of your solution to mycourses.

1. (15 points) In this question you are going to prove the so called Master Theorem for divide and conquer algorithms. The proof is stated in several books and webpages, but you are required (and encouraged) to prove this by yourself without looking it up. Let $a, b > 0$ be integers and $c, d > 0$ be real numbers. Let $\alpha = \log_b a$. Suppose that $T(1) = c$, and for every n divisible by b, we have the recursion $T(n) = aT(n/b) + cn^d$. By using induction prove that for $n = b^m$ where $m \in \mathbb{N}$, we have
   - If $d < \alpha$, then $T(n) = \Theta(n^\alpha)$.
   - If $d = \alpha$, then $T(n) = \Theta(n^\alpha \log(n))$.
   - If $d > \alpha$, then $T(n) = \Theta(n^d)$.

2. We are given a sequence of positive integers $a_1, \dots, a_n$ corresponding to the prices of a stock in times $1, \dots, n$. We want to decide when to buy the stock and when to sell it. More precisely we want to find $i < j$ such that $a_j - a_i$ is maximized. For example if the sequence is $(30, 10, 5, 9, 6, 11, 10)$, then the optimal strategy is to buy at time 3 (at the price of 5) and sell at time 6 (at the price of 11).
   (a) (3 points) Describe a simple $O(n^2)$ solution to this problem.
   (b) (12 points) Improve the running time by using a divide-and-conquer recursive algorithm and analyze its correctness and running time.

3. (10 points) Use the Huffman code to convert the following sentence[1] into a sequence of bits: "that rug really tied the room together."

4. (20 points) Show that no compression scheme can be expected to compress all of the n-character files. Here a character is any of the 256, 8-bit characters. Here a compression scheme is an *injective* map $C : \Omega^n \to \{0, 1\}^*$, where Ω is the set of 256 characters, and $\{0, 1\}^*$ denotes the set of all finite strings of 0's and 1's. In other words, you need to show that for every such C, there exists some $x \in \Omega^n$ such that $|C(x)| \geq 8n$, as $8n$ is the initial size of the file.

5. (20 points) Consider a $2^n \times 2^n$ board missing one cell. We want to cover this board with $\frac{4^n-1}{3}$ tiles. The tiles are L-shaped and consist of three adjacent cells. We are given the location of the missing cell as input. Design a divide-and-conquer algorithm that achieves this task.

6. (20 points) We are given a directed graph on n vertices. Design a $o(n^3)$ algorithm (note that this is little-o) that counts the number of triples (v_1, v_2, v_3) such that v_1v_2, v_2v_3, v_3v_1 are all edges.

[1] do not forget to consider the space and the period as characters too