**CSE 5500 Algorithms. Fall 2018**

Home work 2. Due on November 14 (1:25 PM).

1. Present an $O(n)$ time algorithm to compute the coefficients of the polynomial $(1+x)^n$. How much time is needed if you use the FFT algorithm to solve this problem?

2. An $n \times n$ *Toeplitz* matrix is a matrix $A$ with the property that $A[i,j] = A[i-1, j-1],\ 2 \leq i, j \leq n$. Give an $O(n \log n)$ algorithm to multiply a Toeplitz matrix with an arbitrary $(n \times 1)$ column vector.

3. Let $f(x)$ be a polynomial of degree $n > 0$. This polynomial has $n$ derivatives, each one obtained by taking the derivative of the previous one. Devise an algorithm that computes all the derivatives of $f(.)$ at a given point $a$. Your algorithm should run in time $O(n \log^2 n)$.

4. Let $G(V, E)$ be any weighted connected graph. If $C$ is any cycle of $G$, then show that the heaviest edge of $C$ cannot belong to a minimum-cost spanning tree of $G$.

5. Given a sequence $X$ of symbols, a subsequence of $X$ is defined to be any contiguous portion of $X$. For example, if $X = x_1, x_2, x_3, x_4, x_5,\ x_2, x_3$ and $x_1, x_2, x_3$ are subsequences of $X$. Given two sequences $X$ and $Y$, present an algorithm that will identify the longest subsequence that is common to both $X$ and $Y$. This problem is known as *the longest common subsequence problem*. What is the time complexity of your algorithm?

6. Let $M_1 \times M_2 \times \cdots \times M_r$ be a chain of matrix products. This chain may be evaluated in several different ways. Two possibilities are $(\cdots((M_1 \times M_2) \times M_3) \times M_4) \times \cdots) \times M_r$ and $(M_1 \times (M_2 \times (\cdots \times (M_{r-1} \times M_r) \cdots)$. The cost of any computation of $M_1 \times M_2 \times \cdots \times M_r$ is the number of multiplications used. Let $M_{ij}$ denote the matrix product $M_i \times M_{i+1} \times \cdots \times M_j$. Let $D(i), 0 \leq i \leq r$, represent the dimensions of the matrices, i.e., $M_i$ has $D(i-1)$ rows and $D(i)$ columns. Let $C(i,j)$ be the cost of computing $M_{ij}$ using an optimal product sequence for $M_{ij}$. Observe that $C(i,i) = 0, 1 \leq i \leq r$, and that $C(i, i+1) = D(i-1)D(i)D(i+1), 1 \leq i \leq r$.

   (a) Obtain a recurrence relation for $C(i,j), j > i$.

   (b) Write an algorithm to solve the recurrence relation of (a) for $C(1,r)$. Your algorithm should be of complexity $O(r^3)$.

7. Let $f$ be a flow in a network and let $\alpha$ be a real number. The function $\alpha f : V \times V \to R$ is defined as $(\alpha f)(u,v) = \alpha f(u,v)$. Show that if $f_1$ and $f_2$ are flows in the network then so is $\alpha f_1 + (1-\alpha) f_2$ for all $\alpha$ in the range $[0,1]$.

8. Show the execution of the Ford-Fulkerson technique in finding a max-flow in the network $G(V,E)$ where $V = \{s, 1, 2, 3, 4, t\}$. $c(s,1) = 18; c(s,3) = 12; c(1,2) = 8;\ c(2,1) = 7; c(2,3) = 4; c(3,2) = 11; c(3,1) = 7;\ c(1,3) = 6; c(3,4) = 5; c(4,3) = 6;\ c(2,t) = 14; c(4,t) = 16.$