CS 170 Fall 2013 — Algorithms — Satish Rao — HW 4

Due September 27, 6:00pm

1. (10 pts) Problem 2.33

Suppose you are given $n \times n$ matrices A, B, C and you wish to check whether $AB = C$. You can do this in $O(n^{\log_2 7})$ using Strassen's algorithm. In this question, we will explore a much faster $O(n^2)$ randomized test.

(a) Let $\mathbf{v}$ be an n-dimensional vector whose entries are randomly and independently chosen to be 0 or 1 (each with probability $1/2$). Prove that if M is a non-zero $n \times n$ matrix, then $\Pr[M\mathbf{v} = 0] \leq 1/2$ (that is, the probability that $M\mathbf{v}$ equals the zero vector is at most 1/2).

(b) Show that $\Pr[AB\mathbf{v} = C\mathbf{v}] \leq 1/2$ if $AB \neq C$. Why does this give an $O(n^2)$ randomized test for checking whether $AB = C$?

2. (10 pts) Fast Fourier Transform

Consider the following polynomial:

$$A(x) = 1 + 2x - x^2 + 3x^3.$$

In this question, we will practice the FFT algorithm by doing the first few steps in computing the Fourier transform of A, for $n = 4$.

(a) The Fourier transform of A is the value of A at four inputs x_1, x_2, x_3 and x_4. What are those four inputs?

(b) Find polynomials $A_e(x)$ and $A_o(x)$ such that

$$A(x) = A_e(x^2) + xA_o(x^2)$$

(c) The next step is to recursively call the FFT algorithm to evaluate $A_e(x)$ and $A_o(x)$ for certain values of x. What are those values of x?

3. (10 pts) Fast Fourier Troubles

Ben Bitdiddle is multiplying two polynomials $A(x)$ and $B(x)$ that are each degree d. He plans on using the FFT algorithm, so he calculates ω, a primitive n^{th} root of unity, where n is the smallest power of 2 greater than $2d$. For this problem, you may assume that addition and multiplication take $O(1)$, but not exponentiation.

(a) Ben calculates the FFT of $A(x)$ and saves the value representation of $A(x)$. Unfortunately, Ben's computer malfunctions, and he loses one of the points $A(\omega^i)$. Given i $(0 \leq i \leq 2d)$, how can Ben recalculate $A(\omega^i)$ in $O(n)$ time?

(b) After recovering the value representation of $A(x)$, Ben is able to compute the value representation of $B(x)$ without a problem. He multiplies corresponding $A(x)$ and $B(x)$ points to get the value representaiton of $C(x) = A(x) \times B(x)$, and runs the inverse FFT algorithm to find the coefficient representation of $C(x)$. Then, his computer malfunctions again. This time, Ben loses one of the coefficients c_j. Given j $(0 \leq j \leq 2d)$, how can Ben recalculate c_j in $O(n)$ time?

4. (10 pts) Fast Pattern Matching

We are given a bit pattern s_1 of length m and a bit pattern s_2 of length n, where $m < n$. We wish to find the location of any occurrence of the pattern s_1 within s_2 with k or fewer mismatched bits.

(a) Give an $O(nm)$ algorithm for this problem.

(b) Give an $O(n \log n)$ algorithm for any k.

5. (15 pts) Problem 3.7

A *bipartite graph* is a graph $G = (V, E)$ whose vertices can be partitioned into two sets ($V = V_1 \cup V_2$ and $V_1 \cap V_2 = \emptyset$) such that there are no edges between vertices in the same set (for instance, if $u, v \in V_1$, then there is no edge between u and v).

(a) Give a linear-time algorithm to determine whether an undirected graph is bipartite.

(b) There are many other ways to formulate this property. For instance, an undirected graph is bipartite if and only if it can be colored with just two colors.

Prove the following formulation: an undirected graph is bipartite if and only if it contains no cycles of odd length.

(c) At most how many colors are needed to color in an undirected graph with exactly *one* odd-length cycle?

6. (5 pts) Problem 3.11

Design a linear-time algorithm which, given an undirected graph G and a particular edge e in it, determines whether G has a cycle containing e.

Week 4 Fun Fact

When the Eiffel Tower was built, Gustav Eiffel had the names of 72 French mathematicians, scientists, and engineers engraved on the four sides of the Eiffel Tower. For his contributions to mathematics, Joseph Fourier was one of the 72 individuals chosen for this honor. Others whose names are featured include Pierre-Simon Laplace, André-Marie Ampère, Antoine Lavoisier, Charles-Augustin de Coulomb, and Fourier's doctoral adviser Joseph-Louis Lagrange.