CS 630 - Fall 2021
Homework 2

Due: Friday, October 8 by 5:00pm - submit via Gradescope

Reading : Read Chapter 30, sections 1 and 2, on the FFT in the textbook, this is pages 898 - 914.

Take a look at the next topic, approximation algorithms, which we will be starting in a week or so.

Problems: Please limit your answer to the following problems to at most 1/2 a pages each.

1. Put the following 7 different "Big O" function in order from smallest to largest.
If any of them are equal just put an "=" sign between the equal ones in your ordered list.
(Note: The log function is base 2.)

$O(5^{3n}),\ \ O(15^n),\ \ O(5^n),\ \ O(n^{15}),\ \ O((log\ n)^n),\ \ O(5^{3n+5}),\ \ O(3^{5n})$

2. i. Find the FFT of the two polynomials f(x) $= 6x^3 + x^2 - 1$ and g(x) $= 2x^3 - 4x + 3$ at all of the four 4^{th} roots of unity.
Show some of your work that you did in this process.
Write the 4 point-values of f and of g that you get from computing the FFT values. ?

ii. Now state the 4 point-values of the product f times g by multiplying the values that you obtained in part i.

Are the 4 point-values of the product fg points enough to determine the coefficients of fg ? Why or why not ?

3. Use the inverse FFT to interpolate and find the coefficients of the polynomial f using the inverse FFT algorithm class and using the four point values of f from problem 2, part i as input. (You don't need to do this for g.)

You can do this by either the interpolation algorithm that uses the divide and conquer FFT or the algorithm using the FFT inverse matrix formulation to do this. You should show enough of your work so that we see some of the details of whichever algorithm you are using.

4. Do problem 30.2-6 on page 914.

Explanation: This problem shows that you can set up fast DFT algorithms in other way than using roots of unity. In this case we use computations with numbers mod m.

You need only to prove 3 things to show that the DFT and it's inverse are well defined in this mod m setting. These are the statements in i, ii, and iii below that you need to verify.

i. Show there is an n such that $\omega^n = 1$ and that for all the $t < n$ the ω^t's are distinct mod m. (So here 2^t acts as the primitive root of unity mod m.)

ii. Prove that the sum of all the ω^j's from j = 0 up to j = n-1 = 0. (This fact is analogous to lemma 30.6 in our book.)

iii. Show that for $1 \leq k \leq n/2$ we have that $\omega^{k+n/2} = -\omega^k$. (This factis analogous to lemma 30.4 in our book.)