csci-5454-hw3

Jake Krol

September 2024

1 Question 1

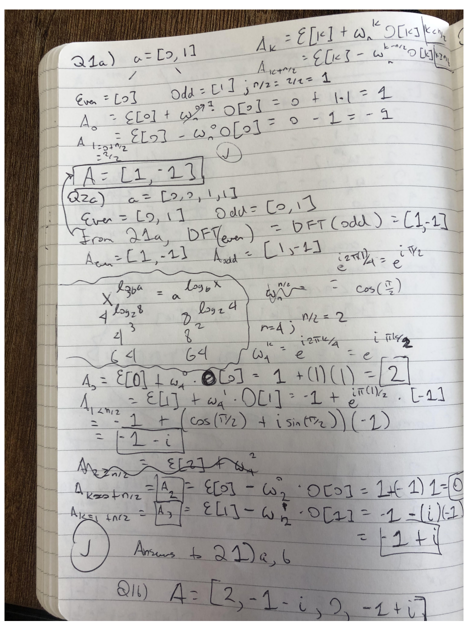
1.1 Q1a

$A = [1, -1]$

1.2 Q1b

$A = [2, (-1 - i), 0, (-1 + i)]$

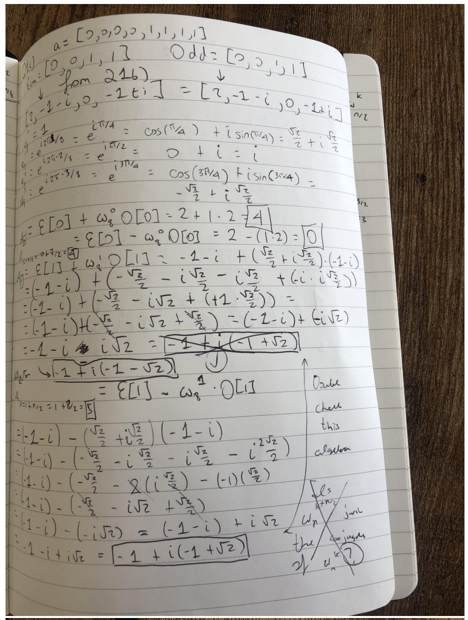
Work for Q1a and Q1b.



1.3 Q1c

$A = [$
$4, (-1 + i(-1 - \sqrt{2})), 0, (-1 + i(1 - \sqrt{2})),$
$0, (-1 + i(-1 + \sqrt{2})), 0, (-1 + i(1 + \sqrt{2}))$
$]$

Work for Q1c.



$a = [0,0,0,0,1,1,1,1]$

2(c) Even $= [0,0,1,1]$ Odd $= [0,0,1,1]$

from 2(b)

$[2, -1-i, 0, -1+i]$ $= [2, -1-i, 0, -1+i]$

$\omega_8^0 = 1$

$\omega_8^1 = e^{i2\pi/8} = e^{i\pi/4} = \cos(\pi/4) + i\sin(\pi/4) = \frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2}$

$\omega_8^2 = e^{i2\pi \cdot 2/8} = e^{i\pi/2} = 0 + i = i$

$\omega_8^3 = e^{i2\pi \cdot 3/8} = e^{i3\pi/4} = \cos(3\pi/4) + i\sin(3\pi/4) = -\frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2}$

$A_0 = \mathcal{E}[0] + \omega_8^0 O[0] = 2 + 1 \cdot 2 = \boxed{4}$

$A_{k+n/2=0+8/2=\boxed{4}} = \mathcal{E}[0] - \omega_8^0 O[0] = 2 - (1 \cdot 2) = \boxed{0}$

$A_1 = \mathcal{E}[1] + \omega_8^1 O[1] = -1-i + (\frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2}) \cdot (-1-i)$

$= (-1-i) + (-\frac{\sqrt{2}}{2} - i\frac{\sqrt{2}}{2} - i\frac{\sqrt{2}}{2} + (-i \cdot i\frac{\sqrt{2}}{2}))$

$= (-1-i) + (-\frac{\sqrt{2}}{2} - i\sqrt{2} + (+1 \cdot \frac{\sqrt{2}}{2})) =$

$= (-1-i) + (-\frac{\sqrt{2}}{2} - i\sqrt{2} + \frac{\sqrt{2}}{2}) = (-1-i) + (-i\sqrt{2})$

$= -1 - i - i\sqrt{2} = \cancel{\boxed{-1 + i(-1+\sqrt{2})}}$

$\boxed{-1 + i(-1-\sqrt{2})}$

$A_{k=1+n/2=1+8/2=\boxed{5}} = \mathcal{E}[1] - \omega_8^1 \cdot O[1]$

$= (-1-i) - (\frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2})(-1-i)$

$= (-1-i) - (-\frac{\sqrt{2}}{2} - i\frac{\sqrt{2}}{2} - i\frac{\sqrt{2}}{2} - i^2\frac{\sqrt{2}}{2})$

$= (-1-i) - (-\frac{\sqrt{2}}{2} - 2(i\frac{\sqrt{2}}{2}) - (-1)(\frac{\sqrt{2}}{2}))$

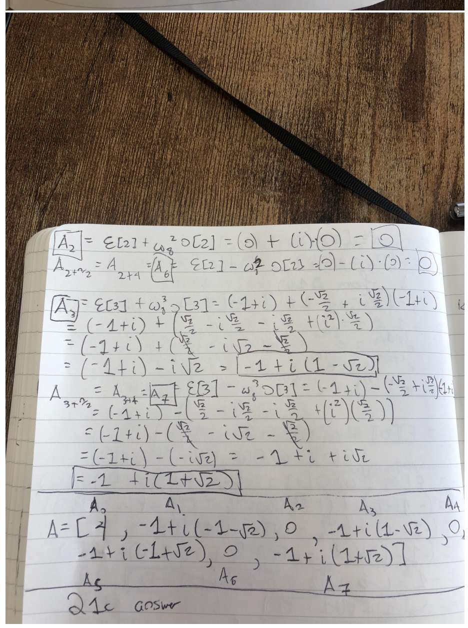
$= (-1-i) - (-\frac{\sqrt{2}}{2} - i\sqrt{2} + \frac{\sqrt{2}}{2})$

$= (-1-i) - (-i\sqrt{2}) = (-1-i) + i\sqrt{2}$

$= -1 - i + i\sqrt{2} = \boxed{-1 + i(-1+\sqrt{2})}$

Double check this algebra

~~Is $\omega_n^{k+n/2}$ just the conjugate of ω_n^k?~~



$\boxed{A_2} = \mathcal{E}[2] + \omega_8^2 O[2] = (0) + (i)\cdot(0) = \boxed{0}$

$A_{2+n/2} = A_{2+4} = \boxed{A_6} = \mathcal{E}[2] - \omega_8^2 O[2] = (0) - (i)\cdot(0) = \boxed{0}$

$\boxed{A_3} = \mathcal{E}[3] + \omega_8^3 O[3] = (-1+i) + (-\frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2})(-1+i)$

$= (-1+i) + (\frac{\sqrt{2}}{2} - i\frac{\sqrt{2}}{2} - i\frac{\sqrt{2}}{2} + (i^2)\cdot\frac{\sqrt{2}}{2})$

$= (-1+i) + (\frac{\sqrt{2}}{2} - i\sqrt{2} - \frac{\sqrt{2}}{2})$

$= (-1+i) - i\sqrt{2} = \boxed{-1 + i(1-\sqrt{2})}$

$A_{3+n/2} = A_{3+4} = \boxed{A_7} = \mathcal{E}[3] - \omega_8^3 O[3] = (-1+i) - (-\frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2})(-1+i)$

$= (-1+i) - (\frac{\sqrt{2}}{2} - i\frac{\sqrt{2}}{2} - i\frac{\sqrt{2}}{2} + (i^2)(\frac{\sqrt{2}}{2}))$

$= (-1+i) - (\frac{\sqrt{2}}{2} - i\sqrt{2} - \frac{\sqrt{2}}{2})$

$= (-1+i) - (-i\sqrt{2}) = -1 + i + i\sqrt{2}$

$\boxed{= -1 + i(1+\sqrt{2})}$

$A = [\underset{A_0}{4}, \underset{A_1}{-1+i(-1-\sqrt{2})}, \underset{A_2}{0}, \underset{A_3}{-1+i(1-\sqrt{2})}, \underset{A_4}{0}, \underset{A_5}{-1+i(-1+\sqrt{2})}, \underset{A_6}{0}, \underset{A_7}{-1+i(1+\sqrt{2})}]$

2(c) answer

2 Question 2

Question 2 considers the DFT of periodic binary, valued arrays with length n and entries equal to 1 at integer multiples, l, of the period, p where $lp|n$ and $lp \le \frac{n}{p} - 1$ (i.e., $l \in \{0, \ldots, \frac{n}{p} - 1\}$).

After empirically evaluating arrays of this type

$$a_j = \begin{cases} 1 & \text{if } j = lp \text{ where } l \in \{0, \dots, \frac{n}{p} - 1\} \\ 0 & \text{else} \end{cases}$$

a pattern emerged for DFTs of a ($A = DFT(a)$).

A is also periodic; the new period, p_A, is equal to the ratio of the size of the array to the original period: $p_A = \frac{n}{p_a}$. Similarly, values of A with an index in the set of integer multiples of the period are equal to p_A. Meaning, the jth index of is generalized by

$$A_j = \begin{cases} p_A & \text{if } j = lp_A \text{ where } l \in \{0, \dots, \frac{n}{p_A} - 1\} \\ 0 & \text{else} \end{cases}$$

3 Question 3

Context

The FFT algorithm splits a polynomial coefficient array into two sub arrays grouped by even and odd indices which can be recomposed by $A(x) = A_{even}(x^2) + xA_{odd}(x^2)$. Here, the coefficient array is split into 3 parts by indexing the original array, a, in steps of 3. C gets indices $\{0, 3, \ldots, \frac{n}{3} - 3\}$, D gets the indices of C incremented by 1, and E gets the indices of C incremented by 2; this continues until the final index $(n-1)$ is reached.

A can be re-written as a sum of these three polynomials, $A(x) = C(x^3) + xD(x^3) + x^2E(x^3)$. Evaluating at x^3 accounts for the split polynomials having $\frac{1}{3}$ the degree of the original polynomial, and the x and x^2 factors for D and E account for the offset/step procedure used to select the coefficients.

Similar to the FFT algorithm, evaluating the polynomials at the roots of unity (and taking advantage of special root of unity properties) can be used to compute the DFT.

For A_l, $A_{l+\frac{n}{3}}$and$A_{l+\frac{2n}{3}}$ in the conquered array, the exponent and cancellation properties of the roots of unity will allow us to express $\omega_n^{l+\frac{n}{3}}$ and $\omega_n^{l+\frac{2n}{3}}$ in terms of ω_n^l multiplied by the first $(\omega_{n=3}^1)$ or second $(\omega_{n=3}^2)$ of $n = 3$ roots of unity to reduce the amount of computation required at each l.

Here are some transformations of roots of unity, by cancellation property and general exponent properties, to simplify the FFT calculations:

- $(\omega_n^l)^3 = e^{\frac{i2\pi 3l}{n}} = e^{\frac{i2\pi l}{\frac{n}{3}}} = \omega_{\frac{n}{3}}^l$
- $\omega_n^{l+\frac{n}{3}} = \omega_n^l * \omega_n^{\frac{n}{3}}$
- $\omega_n^{\frac{n}{3}} = \omega_{3n}^n = \omega_3^1$

Therefore,

$$\omega_n^{l+\frac{n}{3}} = \omega_n^l * \omega_3^1$$
$$\omega_n^{l+\frac{2n}{3}} = \omega_n^l * \omega_3^2$$

allows calculating A_l, $A_{l+\frac{n}{3}}$and$A_{l+\frac{2n}{3}}$ for all l in terms of only $\omega_n^l, \omega_3^1, \omega_3^2$. ω_3^1, ω_3^2 are re-used across l iterations.

3.1 Q3a

Evaluate $A(x) = C(x^3) + xD(x^3) + x^2E(x^3)$ at ω_n^l.

$$A_l = C(\omega_n^{3l}) + \omega_n^l * D(\omega_n^{3l}) + \omega_n^{2l} * E(\omega_n^{3l})$$

3.2 Q3b

Evaluate $A(x) = C(x^3) + xD(x^3) + x^2E(x^3)$ at $\omega_n^{l+\frac{n}{3}}$.

$$A_{l+\frac{n}{3}} = C(\omega_n^{l+\frac{n}{3}})^3 + \omega_n^{l+\frac{n}{3}} * D(\omega_n^{l+\frac{n}{3}})^3 + (\omega_n^{l+\frac{n}{3}})^2(E(\omega_n^{l+\frac{n}{3}})^3)$$
$$= C(\omega_n^l * \omega_3^1)^3 + \omega_n^l * \omega_3^1(D(\omega_n^l * \omega_3^1)^3) + (\omega_n^l * \omega_3^1)^2(E(\omega_n^l * \omega_3^1)^3)$$

3.3 Q3c

Evaluate $A(x) = C(x^3) + xD(x^3) + x^2E(x^3)$ at $\omega_n^{l+\frac{2n}{3}}$.

$$A_{l+\frac{2n}{3}} = C(\omega_n^{l+\frac{2n}{3}})^3 + \omega_n^{l+\frac{2n}{3}} * D(\omega_n^{l+\frac{2n}{3}})^3 + (\omega_n^{l+\frac{2n}{3}})^2(E(\omega_n^{l+\frac{2n}{3}})^3)$$
$$= C(\omega_n^l * \omega_3^2)^3 + \omega_n^l * \omega_3^2(D(\omega_n^l * \omega_3^2)^3) + (\omega_n^l * \omega_3^2)^2(E(\omega_n^l * \omega_3^2)^3)$$

3.4 Q3d

Algorithm 1 FFT(a,n)

if $n == 1$ **then**
  return a
end if
$C = a[0 : \frac{n}{3} - 3 : 3]$ ▷ indexing syntax is start:end:step
$D = a[1 : \frac{n}{3} - 2 : 3]$
$E = a[2 : \frac{n}{3} - 1 : 3]$
$A^C = FFT(C, \frac{n}{3})$
$A^D = FFT(D, \frac{n}{3})$
$A^E = FFT(E, \frac{n}{3})$
$A = ARRAY(n)$ ▷ initialize an array of size n
$\omega_3^1 = e^{\frac{i2\pi}{3}}$ ▷ Precompute 1st and 2nd roots of unity for $n = 3$
$\omega_3^2 = e^{\frac{i4\pi}{3}}$
for $l = 0$ **upto** $\frac{n}{3} - 1$ **do**
  $\omega_n^l = e^{\frac{2\pi il}{n}}$ ▷ i is $\sqrt{-1}$
  $A[l] = A^C[l] + \omega_n^l * A^D[l] + \omega_n^{2l} * A^E[l]$
  $A[l + \frac{n}{3}] = A^C[l] + \omega_n^l * \omega_3^1 * A^D[l] + (\omega_n^l * \omega_3^1)^2 A^E[l]$
  $A[l + \frac{2n}{3}] = A^C[l] + \omega_n^l * \omega_3^2 * A^D[l] + (\omega_n^l * \omega_3^2)^2 A^E[l]$
end for
return A

The input array, a, is partitioned by index steps of 3 into three arrays: C, D, E. This is done recursively until the base case array of length 1 where the DFT is equal to itself. The combined DFT array is computed by looping over the sub arrays with a tracking index, l, and computing the lth root of unity ($\omega_n^l = e^{\frac{2\pi il}{n}}$) for $n =$ size of combined array. T, A, is filled at $A_l, A_{l+\frac{n}{3}}, A_{l+\frac{n}{3}}$ by the equations in Q3a to Q3c; where, A is the combined FFT.