Problem Set 5

1. **FACT 1**: Same. ~~...~~ DFT can still be seen as polynomial evaluation at roots-of-unity.

FACT 2: let N be divisible by 3, then $N = 3m$

$$p(z) = \sum_{n=0}^{3m-1} a_n z^n$$
$$= \sum_{n=0}^{m-1} a_{3n} z^{3n} + z \sum_{n=0}^{m-1} a_{3n+1} z^{3n} + z^2 \sum_{n=0}^{m-1} a_{3n+2} z^{3n}$$

$$\therefore \underbrace{p(z)}_{\text{degree } N-1} = \underbrace{p_1(z^3)}_{\text{degree } \frac{N}{3}-1} + z \underbrace{p_2(z^3)}_{\text{degree } N/3 - 1} + z^2 \underbrace{p_3(z^3)}_{\text{degree } N/3 - 1}$$

FACT 3: let $z_k = e^{-2\pi i k/N}$ be roots-of-unity, where $N = 3m$

Then $z_k^3 = \left(e^{-2\pi i k/N}\right)^3 = e^{-2\pi i k/(N/3)} = e^{-2\pi i k/m}$

so $z_0^3, \ldots, z_N^3$ are the roots-of-unity (m of them) repeated 3 times.

$$\therefore \begin{bmatrix} p(z_0) \\ \vdots \\ p(z_{m-1}) \end{bmatrix} = \quad \begin{bmatrix} p(z_0) \\ \vdots \\ p(z_{m-1}) \\ p(z_m) \\ \vdots \\ p(z_{2m-1}) \\ p(z_{2m}) \\ \vdots \\ p(z_N) \end{bmatrix} = \begin{bmatrix} I & I & I \\ I & e^{-2\pi i/3} & (e^{-2\pi i/3})^2 \\ I & (e^{-2\pi i/3})^2 & (e^{-2\pi i/3})^4 \end{bmatrix} \begin{bmatrix} p_1(z_0^3) \\ \vdots \\ p_1(z_{m-1}^3) \\ z_0 p_2(z_0^3) \\ \vdots \\ z_{m-1} p_2(z_{m-1}^3) \\ z_0^2 p_3(z_0^3) \\ \vdots \\ z_{m-1}^2 p_3(z_{m-1}^3) \end{bmatrix} \quad (*)$$

Therefore, $p(z_0), \dots, p(z_N)$ can be computed by

$\{p_1(z_0^3), \dots p_1(z_m^3)\}$, $\{p_2(z_0^3), \dots, p_2(z_m^3)\}$

, $\{p_3(z_0^3), \dots, p_3(z_m^3)\}$

Each one is the same ~~problem~~ as original problem with N replaced by $N/3$.

All that is required for full credit

Repeating gives an $O(N \log_3 N)$ algorithm

In $(*)$, I also used the fact that

$$z_{N/3+k} = e^{-2\pi i/3} z_k$$

and

$$z_{2N/3+k} = (e^{-2\pi i/3})^2 z_k$$

this is not neccessary to get full credit.

$(*)$

2. $q_1(z)q_2(z)$ is of degree 5, next power of 2 is 8.

Think of $q_1(z) = 0z^0 + z + 0z^2 + z^3 + 0z^4 + 0z^5 + 0z^6 + 0z^7$

↳ use FFT of length 8 to evaluate $q_1(z)$ at 8 roots-of-unity.

Think of $q_2(z) = 1 + z + z^2 + 0z^3 + \dots + 0z^7$

↳ Use FFT of length 8 to evaluate $q_2(z)$ at 8 roots of unity.

Multiply together: $q_1(z_k)q_2(z_k)$ (easy)

I'll get back 8 values, now do inverse FFT to get back

$$\cancel{h} \quad q_1(z) q_2(z) = a_0 z^0 + a_1 z + \dots + a_7 z^7.$$

3.

$$C_n = \begin{bmatrix} 2 & -1 & & & & -1 \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & -1 & 2 & -1 \\ -1 & & & & -1 & 2 \end{bmatrix}$$

~~For rows~~ For row $2 \le j \le n-2$

$$-\omega^{k(j-1)} + 2\omega^{kj} - \omega^{k(j+1)} = \omega^{kj}\left[-\omega^{-k} + 2 - \omega^{k}\right]$$

$$\cancel{= \omega^{kj}[2 - 2\cos(k2\omega)}$$

$$= \omega^{kj}\left[2 - 2\cos(2\pi k/n)\right]$$

For $j=1$: $-\omega^{k(n-1)} + 2 - \omega^{k} = -\omega^{-k} + 2 - \omega^{k}$

$$= \left[2 - 2\cos(2\pi k/n)\right]$$

For $j = n-1$: $-1 + 2\omega^{k(n-1)} - \omega^{k(n-2)} = -\omega^{kn} + 2\omega^{k(n-1)} - \omega^{k(n-2)}$

$$= \omega^{k(n-1)}\left[-\omega^{k} + 2 - \omega^{-kn}\right] = \omega^{k(n-1)}\left[2 - 2\cos(2\pi k/n)\right]$$

(b) $C_n = F^{-1} \Lambda F$, F = DFT matrix of size N.

$\Lambda = \text{diag}(\lambda_1 \dots \lambda_n)$.

$$\lambda_k = 2 - 2\cos(2\pi k/n).$$

~~(c)~~ $\cancel{C_n X + X C_n = F^{-1}\Lambda F X + X F^{-1} \Lambda F = G}$

So $\cancel{\Lambda F X F^{-1} +}$

4. $$p(t+1) = \sum_{k=0}^{N-1} c_k e^{2\pi i k(t+1)}$$

$$= \sum_{k=0}^{N-1} c_k e^{2\pi i k t} \underbrace{e^{2\pi i k}}_{=1} = p(t).$$

$p(t+1) = p(t)$, $\Rightarrow$ testosterone levels of athlete repeat every 24 hrs.

$$c_0 = \frac{1}{N}\sum_{k=0}^{N-1} p(t_n) e^{-2\pi i k \times 0} = \frac{1}{N}\sum_{n=0}^{N-1} p(t_n) = \text{mean of samples.}$$

(b) We want to compute:

$$c_K = \frac{1}{N}\sum_{n=0}^{N-1} p(s_n) e^{-2\pi i k s_n}, \quad 0 \le k \le N-1$$

In matrix form we have

$$\begin{bmatrix} c_0 \\ \vdots \\ c_{N-1} \end{bmatrix} = \tilde{F} \begin{bmatrix} p(s_0) \\ \vdots \\ p(s_{N-1}) \end{bmatrix}, \text{ where } (\tilde{F})_{kn} = e^{-2\pi i k s_n}.$$

(c) By Taylor we have,

$$e^{-x} = 1 - x + \frac{x^2}{2} - \frac{x^3}{6} + \dots = \sum_{j=0}^{\infty} (-1)^j \frac{x^j}{j!} \quad (\text{about } x=0)$$

$$\left|\begin{matrix}\text{Error} \\ \text{after } k \\ \text{terms}\end{matrix}\right| \le \frac{1}{(k+1)!} |0.1|^{k+1} < 10^{-3}$$

$$\Rightarrow k \ge 3$$

$$\therefore \; e^{-2\pi i k s_n} = e^{-2\pi i k(t_n + \varepsilon_n)}$$

$$= e^{-2\pi i k t_n} e^{-2\pi i k \varepsilon_n}$$

$$= e^{-2\pi i k t_n}\left[1 - (2\pi i k \varepsilon_n) + \frac{(2\pi i k \varepsilon_n)^2}{2} - \frac{(2\pi i k \varepsilon_n)^3}{6} + \dots\right]$$

I will need ~~at most~~ 3 terms to ~~approx~~ have error less than 10^{-3}

because $|2\pi i k \varepsilon_n| \le \left|2\pi N \cdot \frac{0.01}{N}\right| = 0.063 \le 0.1$

[2 is not enough!]

(d)

So $$(\tilde{F})_{kn} \approx (F)_{kn}\left[1 - 2\pi i k\varepsilon_n + \frac{(2\pi i k\varepsilon_n)^2}{2} - \frac{(2\pi i k\varepsilon_n)^3}{6}\right]$$

with error of $\leq 10^{-3}$.

Apply to every entry:

$$\tilde{F} \approx F - 2\pi i\begin{bmatrix}0 & & & \\ & 1 & & \\ & & 2 & \\ & & & \ddots \\ & & & & N-1\end{bmatrix} F \begin{bmatrix}\varepsilon_0 & & & \\ & \varepsilon_1 & & \\ & & \ddots & \\ & & & \varepsilon_{N-1}\end{bmatrix}$$

$$+ \frac{(2\pi i)^2}{2}\begin{bmatrix}0 & & & \\ & 1 & & \\ & & 2 & \\ & & & \ddots \\ & & & & N-1\end{bmatrix}^2 F \begin{bmatrix}\varepsilon_0 & & & \\ & \varepsilon_1 & & \\ & & \ddots & \\ & & & \varepsilon_{N-1}\end{bmatrix}^2$$

$$- \frac{(2\pi i)^3}{6}\begin{bmatrix}0 & & & \\ & 1 & & \\ & & 2 & \\ & & & \ddots \\ & & & & N-1\end{bmatrix}^3 F \begin{bmatrix}\varepsilon_0 & & \\ & \ddots & \\ & & \varepsilon_{N-1}\end{bmatrix}^3$$

Doctor needs:

$$\tilde{F}\underset{\sim}{p} \approx F\underset{\sim}{p} - 2\pi i\, D_k F D_{\varepsilon_n}\underset{\sim}{p} + \frac{(2\pi i)^2}{2} D_k^2 F D_{\varepsilon_n}^2 \underset{\sim}{p} - \frac{(2\pi i)^3}{6} D_k^3 F D_{\varepsilon_n}^3 \underset{\sim}{p}$$

~~Diagonal matrices (very fast)~~

$$\underset{\sim}{p} = \begin{bmatrix} p(s_0) \\ \vdots \\ p(s_{N-1})\end{bmatrix} \quad D_k = \begin{bmatrix}0 & & & \\ & 1 & & \\ & & \ddots & \\ & & & N-1\end{bmatrix}$$

$$D_{\varepsilon_n} = \begin{bmatrix}\varepsilon_0 & & \\ & \ddots & \\ & & \varepsilon_{N-1}\end{bmatrix}$$

Thus, we can approximate $\tilde{F}_f$ with

4 FFTs	$O(N \log_2 N)$
6 diagonal matrix products	$O(N)$
A few vector adds	$O(N)$
	$O(N \log_2 N)$

Doctor is happy.