Lecture 24

Tuesday, 5 October 2021 1:05 PM

N^{th} root of unity : $\omega^N = 1$

N^{th} Primitive root of unity : $\omega^N = 1$ and $\omega^i \neq 1$, $1 \leq i \lneq N$

Eg $N = 4$ roots : $1, i, -1, -i$ Primitive roots : $i, -i$

Properties of N^{th} root of unity (other than 1)

① If $\omega \, (\neq 1)$ is N^{th} root then $1 + \omega + \omega^2 + \cdots + \omega^{N-1} = 0$

Proof $1 + \omega + \omega^2 + \cdots + \omega^{N-1} = \dfrac{\omega^N - 1}{\omega - 1} = 0$ whenever $\omega \neq 1$

② If ω is N^{th} Primitive root & $1 \leq i \lneq N$ then $1 + \omega^i + \omega^{2i} + \omega^{3i} + \cdots + \omega^{i(N-1)} = 0$

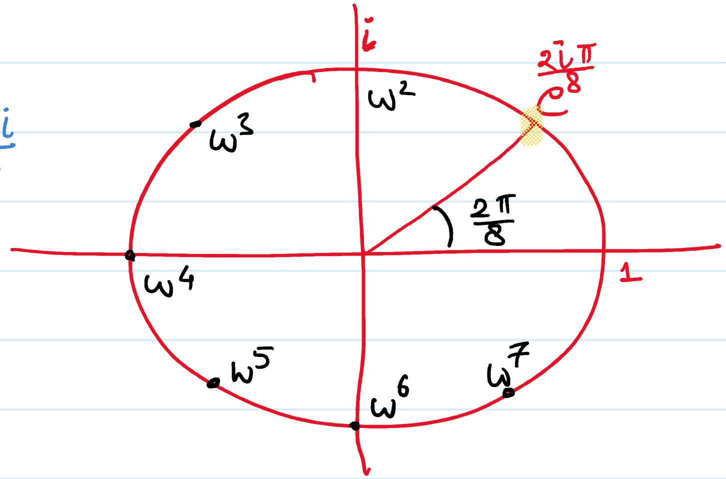
(Note: If ω is N^{th} Primitive root then $\omega^i \neq 1$) together with ① $\Rightarrow$ ②

$N = 8$

ω = primitive root $= e^{\frac{2\pi i}{8}}$

$\omega^8 = 1$

$\omega^{i<8} \neq 1$



i not primitive root when $N = 8$

($i^4 = 1$)

Given: Polynomials $A(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$

$B(x) = b_0 + b_1 x + b_2 x^2 + \dots + b_n x^n$

with

* integer coefficients
* degree $\leq n$

Find: $C(x) = A(x) \cdot B(x)$

$$= c_0 + c_1 x + c_2 x^2 + \dots + c_{2n} x^{2n}$$

$A(x)$ & $B(x)$ (deg $\leq n$)

$\downarrow$ DFT — STEP 1 — $O(n \log n)$ time ✔

$A(x_1)$ $B(x_1)$

$A(x_2)$ $B(x_2)$

$\vdots$ $\vdots$

$A(x_{2n+1})$ $B(x_{2n+1})$

EVALUATION AT $2n+1$ points

$\xrightarrow[\text{STEP 2}]{O(n) ✔}$

$C(x_1) = A(x_1) * B(x_1)$

$C(x_2) = A(x_2) * B(x_2)$

$\vdots$

$C(x_{2n+1})$

PAIRWISE PRODUCT

$\uparrow$ Inverse DFT — STEP 3 — $O(n \log n)$ time

$C(x) = A(x) \cdot B(x)$ (deg $\leq 2n$)

Interpolation

Def^n

⊕ Discrete Fourier Transform (DFT)

Discrete Fourier Transform (DFT)

The DFT of $[a_0\ a_1\ \dots\ a_n]$ (or $A(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$) is evaluation of $A(x)$ at

$$\{1, \omega, \omega^2, \dots, \omega^{N-1}\}$$

$N = n+1$

$\omega = N^{th}$ root of 1

$$DFT(a_0\ a_1\ \dots\ a_n) = \left(A(\omega^0), A(\omega^1), \dots, A(\omega^{N-1})\right)$$

Algorithm to find DFT that we studied (in last class) — Fast Fourier Transform (FFT)

Inverse Discrete Fourier Transform (Inverse DFT)

Given evaluations $(y_0\ y_1\ \dots\ y_n)$, Inverse DFT$(y_0\ y_1\ \dots\ y_n)$ is a polynomial $A(x) = a_0 + a_1x + \dots + a_nx^n$

s.t. $A(\omega^i) = y_i$

$$\text{Inverse-DFT}(y_0\ y_1\ \dots\ y_n) = [a_0\ a_1\ \dots\ a_n]$$

DFT as Matrix Product

DFT as Matrix Product

$$A(x) = \text{Dot Prod}\left(\begin{matrix} [a_0 \ \ a_1 \ \ldots \ a_n] \\ [1 \ \ x \ \ x^2 \ldots x^n] \end{matrix} \right)$$

$$\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix} \leftarrow \begin{bmatrix} A(1) \\ A(\omega^1) \\ A(\omega^2) \\ \vdots \\ A(\omega^i) \\ \vdots \\ A(\omega^n) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^j & \cdots & \omega^n \\ 1 \\ \vdots \\ 1 & \omega^i & (\omega^i)^2 & \cdots & (\omega^i)^j & \cdots & (\omega^i)^n \\ \vdots \\ 1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_j \\ \vdots \\ a_n \end{bmatrix}$$

Vandermonde Matrix

Inverse DFT

Pre multiplying vector $[y_0 \ldots y_n]$ by inverse of Vandermonde (ω).

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_j \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^j & \cdots & \omega^n \\ 1 \\ \vdots \\ 1 & \omega^i & (\omega^i)^2 & \cdots & (\omega^i)^j & \cdots & (\omega^i)^n \\ \vdots \\ 1 \end{bmatrix}^{-1} \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_i \\ \vdots \\ y_n \end{bmatrix}$$

How to find Inverse?

$\omega = e^{\frac{2\pi i}{N}}$ $\qquad$ $\omega^{-1} = e^{\frac{-2\pi i}{N}}$

Vandermonde (ω) $\xleftarrow{\text{(Inverse?)}}$ $\xrightarrow{}$ Vandermonde (ω^{-1})

$$\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^k & \cdots & \omega^n \\ 1 & & & & & & \\ \vdots & & & & & & \\ 1 & \omega^i & (\omega^i)^2 & \cdots & \underline{(\omega^i)^k} & \cdots & (\omega^i)^n \\ \vdots & & & & & & \\ 1 & & & & & & \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & \cdots & \underline{1} & \cdots & 1 \\ 1 & \omega^{-1} & (\omega^{-1})^2 & \cdots & \underline{(\omega^{-1})^j} & \cdots & (\omega^{-1})^n \\ 1 & & & & & & \\ \vdots & & & & & & \\ 1 & \omega^{-k} & (\omega^{-k})^2 & \cdots & \underline{(\omega^{-k})^j} & \cdots & (\omega^{-k})^n \\ \vdots & & & & & & \\ 1 & & & & & & \end{bmatrix}$$

$$(ij)^{th} \text{ entry} = \sum_{k=0}^{n} (\omega^i)^k (\omega^{-k})^j = \sum_{k=0}^{n} (\omega^{i-j})^k = \begin{cases} 0 & i \neq j \\ \underline{n+1} \; N & i = j \end{cases}$$

Inverse DFT

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_j \\ \vdots \\ a_n \end{bmatrix} = \frac{1}{N} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & \cdots & 1 \\ 1 & \omega^{-1} & (\omega^{-1})^2 & \cdots & (\omega^{-1})^j & \cdots & (\omega^{-1})^n \\ 1 & & & & & & \\ \vdots & & & & & & \\ 1 & \omega^{-i} & (\omega^{-i})^2 & \cdots & (\omega^{-i})^j & \cdots & (\omega^{-i})^n \\ \vdots & & & & & & \\ 1 & & & & & & \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_i \\ \vdots \\ y_n \end{bmatrix}$$

$\omega = e^{\frac{2\pi i}{N}}$

$\omega^{-1} = e^{\frac{-2\pi i}{N}}$

$$\begin{bmatrix} \vdots \\ a_n \end{bmatrix} \qquad \begin{bmatrix} \vdots \\ 1 \end{bmatrix} \qquad \begin{bmatrix} \vdots \\ y_n \end{bmatrix}$$

Vandermonde (ω^{-1})

Inverse DFT $(y_0\ y_1 \ldots y_N)$ is just polynomial evaluation over the set

$$S_{INV} = \{1, \omega^{-1}, \omega^{-2}, \ldots, \omega^{-n}\}$$

and the polynomial is $\left(\left(\frac{y_0}{N}\right) + \left(\frac{y_1}{N}\right)x + \left(\frac{y_2}{N}\right)x^2 + \ldots + \left(\frac{y_n}{N}\right)x^n\right)$

(*) Inverse DFT can be computed in $O(n \log n)$ time using $\underbrace{\text{FFT algo.}}_{\text{Divide + Conquer}}$

> **Theorem:** Polynomials of degree $\leq n$ can be multiplied in $O(n \log n)$ time.

Application

$[0, 9]$

① Suppose you have 2 integers $a = (a_n \; a_{n-1} \; \ldots \; a_0)$
$b = (b_n \; b_{n-1} \; \ldots \; b_0)$

Find $(x \cdot y)$ in $O(n \log n)$ time

H.W. (Using polynomial product)

(i) Find poly $P_a(x)$ using a
(ii) Find poly $P_b(x)$ using b
(iii) In $O(n \log n)$ time find $P_a(x) * P_b(x)$
(iv) Find $a \cdot b$ using $P_a(x) \cdot P_b(x)$

② Given two sets $A = \{a_1, a_2 \ldots a_n\}$ and $B = \{b_1 \ldots b_n\}$ with distinct values in range $[1, M]$. $n \leq M$

Find $C = \{a_i + b_j \mid a_i \in A \text{ and } b_j \in B\}$ in $O(M \log M)$ time

H.W.

$\alpha_0 + \alpha_1 x + \alpha_2 x^2 + \ldots + \alpha_i x^i + \ldots$

$$\beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_i x^i + \ldots$$

Coeff of $x^{\boxed{i}}$ $(\alpha_0 \beta_i + \alpha_1 \beta_{i-1} + \ldots + \quad)$

$0/i \qquad 1/i-1 \qquad 2/i-2 \qquad 3/i-3$

Hint : Powers are added on taking product $x^a x^b$

$$x^{a+b}$$