競技プログラミング
だれでもわかる FFT/NTT 入門

monkukui
HCPC M1

はじめに

- 前提知識は高校数学までです.
- 数式を追う部分がかなりあるので,
  頑張ってついてきてください.
- NTT の部分は自分の理解が曖昧なので，逆に教えてください
- 実装には触れません（再帰/非再帰など，色々あるみたい）

FFT って何？

- 高速フーリエ変換（fast Fourier transform, FFT）：
  - 離散フーリエ変換を計算機上で高速に計算するアルゴリズム
    - （ウィキペディアから引用）
- 離散フーリエ変換 $\rightarrow$
  - $f(x)$：$n-1$ 次多項式
  - ζ_n：1 の n 乗根

$$\hat{f}(t) = \sum_{i=0}^{n-1} f(\zeta_n^i) t^i$$

- 高速って？
  - $\mathcal{O}(n^2)$ から $\mathcal{O}(n \log n)$ へ
- 何が嬉しいの？
  - 多項式乗算ができる

どんな問題が解けるか

- 最強コン A：Equal Weight

どんな問題が解けるか

- 最強コン A：Equal Weight

  - $(x + x^2 + x^4)(x^3 + x^6 + x^{10} + x^{15})[x^7] = 2$

    - $f(x)[x^a] := x^a$ の係数

  - 多項式乗算は殴り性能がかなり高い！

多項式乗算問題

- 多項式乗算問題 [定義]
  - 入力：多項式 $f(x), g(x)$ を表す，長さ n と m の配列
  - 出力：$f(x)g(x)$ を表す，長さ $n+m-1$ の配列

- 入力例：$f(x) = (1+2x),\ g(x) = (3+x+4x^2)$

- 出力例：$f(x)g(x) = 3+7x+6x^2+8x^3$

どんな計算量で解けるの？

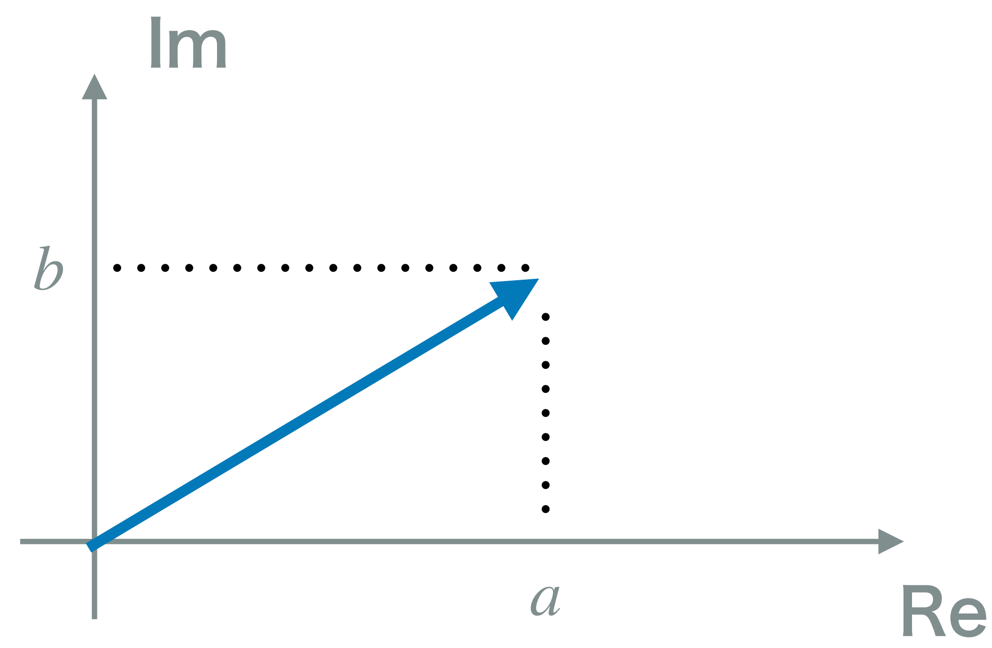
- 多項式乗算問題 [定義]
  - 入力：多項式 $f(x), g(x)$ を表す，長さ n と m の配列
  - 出力：$f(x)g(x)$ を表す，長さ $n+m-1$ の配列

- 自明な for loop アルゴリズムで $\mathcal{O}(nm)$

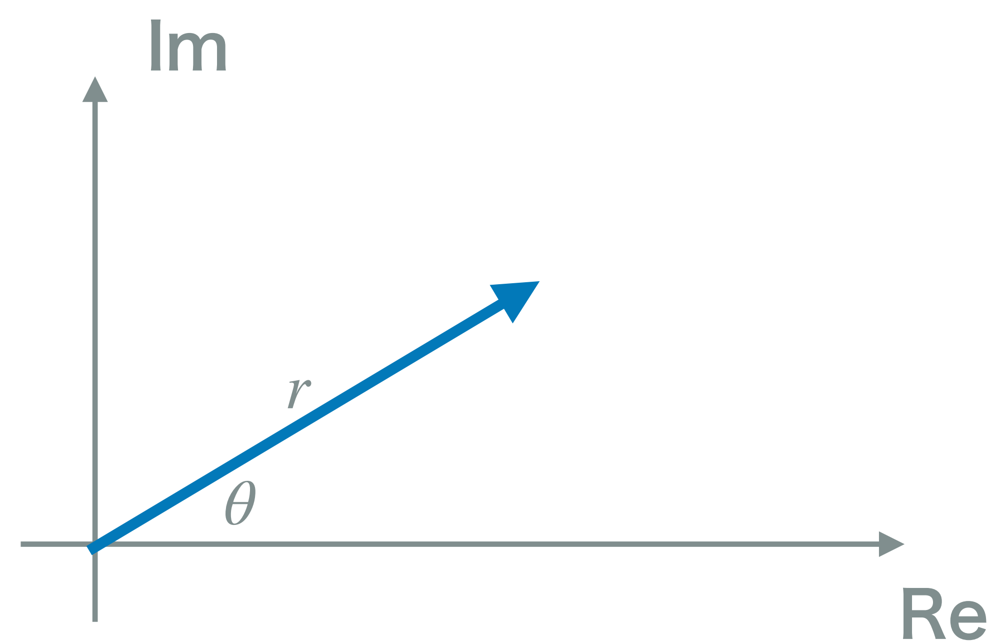
- 高速フーリエ変換で $\mathcal{O}(n \log n)$

準備・複素数（1/3）

- 複素数 $z = a + bi$



- 複素数の極座標形式 $z = r(\cos\theta + i\sin\theta)$

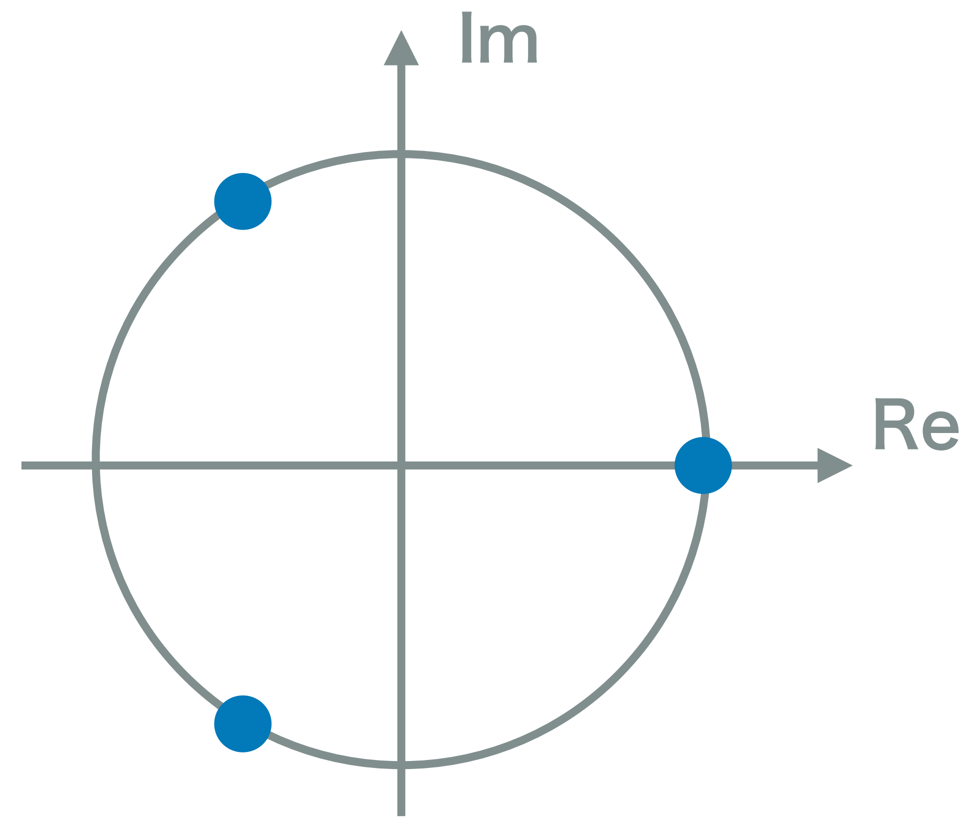


準備・複素数（2/3）

- ド・モアブルの定理：$(\cos\theta + i\sin\theta)^n = \cos n\theta + i\sin n\theta$
  - 帰納法で示せる

準備・複素数（3/3）［重要］

- 1 の n 乗根：$x^n = 1$ の根
  - n 乗して 1 になる複素数全体



性質１：
1 の n 乗根は複素数平面の
単位演周上に等間隔で並ぶ

性質２：
1 の n 乗根は全部で n 個あり，
それらの和は 0 になる

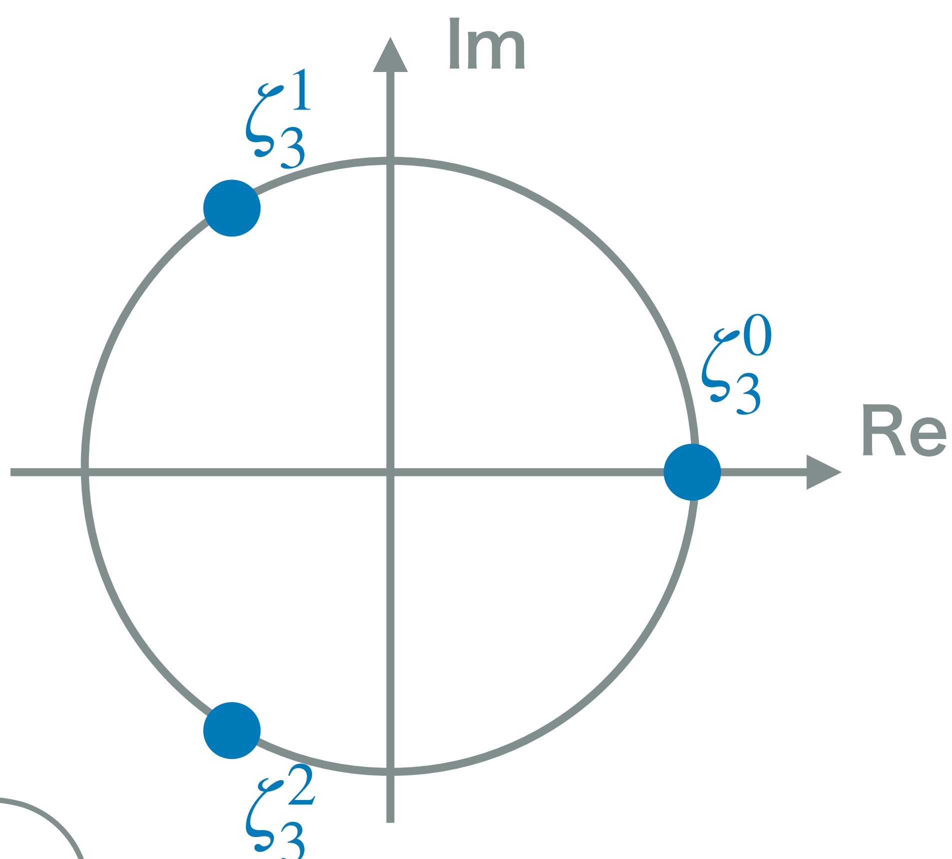
1 の 3 乗根：

$$1, \frac{-1+\sqrt{3}i}{2}, \frac{-1+-\sqrt{3}i}{2}$$

ζ_N の性質

- 1 の 原始 N 乗根：N 乗して初めて 1 になる数
  - 複素数の範囲では,

$$\zeta_N = \cos\frac{2\pi}{N} + i\sin\frac{2\pi}{N}$$ は 1 の原始 N 乗根



- ζ_N にはいくつかの重要な性質がある

ζ_N を ζ_N^{-1} に置き換えても，これらの性質は成り立つ

性質 1：$\zeta_N^i = \zeta_N^{i+N}$

性質 2：$\displaystyle\sum_{i=0}^{N-1} \zeta_N^{i(j-k)} = \begin{cases} N & \text{if } j \equiv k \mod N, \\ 0 & \text{otherwise}. \end{cases}$

ζ_N の性質 1 証明

性質 1：$\zeta_N^i = \zeta_N^{i+N}$

証明： $\zeta_N^i = \zeta_N^i \cdot 1 = \zeta_N^i \cdot \zeta_N^N = \zeta_N^{i+N}$

ζ_N の性質 2 証明

性質 2： $\displaystyle\sum_{i=0}^{N-1} \zeta_N^{i(j-k)} = \begin{cases} N & \text{if } j \equiv k \mod N, \\ 0 & \text{otherwise}. \end{cases}$

- 証明：

(1) $j \equiv k \mod N$ のとき，$j - k = Nm$ と置ける．

$$（与式） = \sum_{i=0}^{N-1} \zeta_N^{Nim} = \sum_{i=0}^{N-1} 1 = N$$

(2) $j \not\equiv k \mod N$ のとき，

$$\frac{1 - (\zeta_N^{j-k})^N}{1 - \zeta_N^{j-k}} = \frac{1 - (\zeta_N^N)^{j-k}}{1 - \zeta_N^{j-k}} = 0 \text{ より，}\quad （与式） = 0$$

離散フーリエ変換の定義

- 離散フーリエ変換（Discrete Fourier Transform, DFT）：

N 次の複素多項式 $f(x)$ から 複素多項式 $\hat{f}(t)$ への写像（??）

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

$$= f(\zeta_N^0)t^0 + f(\zeta_N^1)t^1 + \ldots + f(\zeta_N^{N-2})t^{N-2} + f(\zeta_N^{N-1})t^{N-1}$$

なんか式変形してます

$f(x) = \sum_{j=0}^{N-1} c_j x^j$ とすると,

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

$$= \sum_{i=0}^{N-1} \left(\sum_{j=0}^{N-1} c_j (\zeta_N^i)^j \right) t^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j t)^i$$

$\hat{f}(t)$ に $t = \zeta_N^{-k}$ を代入すると,

$$\hat{f}(\zeta_N^{-k}) = \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j \zeta_N^{-k})^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} \zeta_N^{i(j-k)}$$

$$= N c_k$$

なんか式変形してます

$f(x) = \sum_{j=0}^{N-1} c_j x^j$ とすると,

これは多項式

と,

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

$$= \sum_{i=0}^{N-1} \left(\sum_{j=0}^{N-1} c_j (\zeta_N^i)^j \right) t^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j t)^i$$

$$\zeta_N^{-k})^i$$

$$j=0 \quad i=0$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} \zeta_N^{i(j-k)}$$

$$= N c_k$$

なんか式変形してます

$f(x) = \sum_{j=0}^{N-1} c_j x^j$ とすると,

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

$$= \sum_{i=0}^{N-1} \left(\sum_{j=0}^{N-1} c_j (\zeta_N^i)^j \right) t^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j t)^i$$

$\hat{f}(t)$ に $t = \zeta_N^{-k}$ を代入すると,

DFT の定義

$$\cdots {}_N^{-k})^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} \zeta_N^{i(j-k)}$$

$$= N c_k$$

なんか式変形してます

$f(x) = \sum_{j=0}^{N-1} c_j x^j$ とすると,

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

$$= \sum_{i=0}^{N-1} \left(\sum_{j=0}^{N-1} c_j (\zeta_N^i)^j \right) t^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j t)^i$$

$\hat{f}(t)$ に $t = \zeta_N^{-k}$ を代入すると,

$$\hat{f}(\zeta_N^{-k}) = \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j \zeta_N^{-k})^i$$

代入しただけ

$k)$

$$= N c_k$$

なんか式変形してます

$f(x) = \sum_{j=0}^{N-1} c_j x^j$ とすると,

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

$$= \sum_{i=0}^{N-1} \left(\sum_{j=0}^{N-1} c_j (\zeta_N^i)^j \right) t^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j t)^i$$

$\hat{f}(t)$ に $t = \zeta_N^{-k}$ を代入すると,

$$\hat{f}(\zeta_N^{-k}) = \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j \zeta_N^{-k})^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} \zeta_N^{i(j-k)}$$

シグマの分解

参考：[高校数学の美しい物語]()

なんか式変形してます

動機？

知りません

$$
= \sum_{i=0}^{N-1} \left(\sum_{j=0}^{N-1} c_j (\zeta_N^i)^j \right) t^i
$$

$$
= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j t)^i
$$

$\hat{f}(t)$ に $t = \zeta_N^{-k}$ を代入すると,

$$
\hat{f}(\zeta_N^{-k}) = \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j \zeta_N^{-k})^i
$$

$$
= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} \zeta_N^{i(j-k)}
$$

$$
= N c_k
$$

なんか式変形してます

$f(x) = \sum_{j=0}^{N-1} c_j x^j$ とすると,

$$\hat{f}(t) = \sum$$

代入しただけ

$$= \sum_{i=0} \left(\sum_{j=0} c_j (\zeta_N^i)^j \right) t^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j t)^i$$

$\hat{f}(t)$ に $t = \zeta_N^{-k}$ を代入すると,

$$\hat{f}(\zeta_N^{-k}) = \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j \zeta_N^{-k})^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} \zeta_N^{i(j-k)}$$

$$= N c_k$$

なんか式変形してます

$f(x) = \sum_{j=0}^{N-1} c_j x^j$ とすると,

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

指数法則

$$= \sum_{j=0} c_j \sum_{i=0} (\zeta_N^j t)^i$$

$\hat{f}(t)$ に $t = \zeta_N^{-k}$ を代入すると,

$$\hat{f}(\zeta_N^{-k}) = \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j \zeta_N^{-k})^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} \zeta_N^{i(j-k)}$$

$$= N c_k$$

なんか式変形してます

$f(x) = \sum_{j=0}^{N-1} c_j x^j$ とすると,

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

性質 2

$$\sum_{i=0}^{N-1} \zeta_N^{i(j-k)} = \begin{cases} N & \text{if } j = k \mod N, \\ 0 & \text{otherwise}. \end{cases}$$

より

$\hat{f}(t)$ に $t = \zeta_N^{-k}$ を代入すると,

$$\hat{f}(\zeta_N^{-k}) = \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} (\zeta_N^j \zeta_N^{-k})^i$$

$$= \sum_{j=0}^{N-1} c_j \sum_{i=0}^{N-1} \zeta_N^{i(j-k)}$$

$$= N c_k$$

重要な気づき

$$f(x) = \sum_{i=0}^{N-1} c_i x^i$$

$$= \frac{1}{N} \sum_{i=0}^{N-1} \hat{f}(\zeta_N^{-i}) x^i$$

となるので，$\hat{f}$ から f を**復元できる**

しかも，ζ_N を ζ_N^{-1} と置き換えたら

DFT と同じ形！

これを，**離散フーリエ逆変換**と呼ぶ．

多項式

$$f(x) = \sum_{i=0}^{N-1} c_i x^i$$

DFT の定義

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

さっきのやつ

$$\hat{f}(\zeta_N^{-k}) = N c_K$$

FFT による多項式乗算

$$\widehat{f \cdot g}(t) = \sum_{i=0}^{N-1} (f \cdot g)(\zeta_N^i) t^i$$
$$= \sum_{i=0}^{N-1} f(\zeta_N^i) g(\zeta_N^i) t^i$$

f と g をそれぞれ DFT して係数ごとの積を

計算すると，$\widehat{f \cdot g}$ の DFT が求まる

➡ $\widehat{f \cdot g}$ を離散フーリエ逆変換することで，

所望の $f \cdot g$ を得ることができる！

多項式

$$f(x) = \sum_{i=0}^{N-1} c_i x^i$$

DFT の定義

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

Inverse DFT の定義

$$f(x) = \frac{1}{N} \sum_{i=0}^{N-1} \hat{f}(\zeta_N^{-i}) x^i$$

FFT による多項式乗算 まとめ

- **<u>多項式乗算を求めるためには，</u>**

1. $N > n + m$ となる 最小の2 冪 を見つける
2. DFT の定義に従い，$\hat{f}(t), \hat{g}(t)$ を求める
3. $\hat{f}(t), \hat{g}(t)$ を系数ごとに掛け，
   $\widehat{f \cdot g}\,(t)$ を求める
4. $\widehat{f \cdot g}\,(t)$ を inverse DFT をして，
   $(f \cdot g)(x)$ を復元する

多項式

$$f(x) = \sum_{i=0}^{N-1} c_i x^i$$

DFT の定義

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

Inverse DFT の定義

$$f(x) = \frac{1}{N} \sum_{i=0}^{N-1} \hat{f}(\zeta_N^{-i}) x^i$$

FFT による多項式乗算 まとめ

- 多項式乗算を求めるためには,

1. $N > n + m$ となる 最小の2 冪 を見つける
2. DFT の定義に従い, $\hat{f}(t), \hat{g}(t)$ を求める
3. $\hat{f}(t), \hat{g}(t)$ を係数ごとに掛け,

   $\widehat{f \cdot g}\,(t)$ を求める
4. $\widehat{f \cdot g}\,(t)$ を inverse DFT をして,

   $(f \cdot g)(x)$ を復元する

$\mathcal{O}(\log(n+m))$

でできる

$N = \mathcal{O}(n+m)$

が成り立つ

FFT による多項式乗算 まとめ

- **多項式乗算を求めるためには,**

1. $N > n + m$ となる 最小の2 冪 を見つける
2. DFT の定義に従い, $\hat{f}(t), \hat{g}(t)$ を求める
3. $\hat{f}(t), \hat{g}(t)$ を係数ごとに掛け,
   $\widehat{f \cdot g}\,(t)$ を求める
4. $\widehat{f \cdot g}\,(t)$ を inverse DFT をして,
   $(f \cdot g)(x)$ を復元する

for loop
$O(N)$ でできる

FFT による多項式乗算 まとめ

- **多項式乗算を求めるためには,**

1. $N > n + m$ となる 最小の2 冪 を見つけ
2. DFT の定義に従い, $\hat{f}(t), \hat{g}(t)$ を求める
3. $\hat{f}(t), \hat{g}(t)$ を系数ごとに掛け,
   $\widehat{f \cdot g}(t)$ を求める
4. $\widehat{f \cdot g}(t)$ を inverse DFT をして,
   $(f \cdot g)(x)$ を復元する.

先述した通り,
DFT と inverseDFT は
ζ_N と ζ_N^{-1} の違いと
N で割る部分を除いて同じ

DFT の時間計算量は？

DFT の時間計算量

- DFT 問題 [定義]
  - 入力：多項式 $f(x)$ を表す，長さ N の配列
  - 出力：多項式 $\hat{f}(x)$ を表す，長さ N の配列

DFT の定義

$$\hat{f}(t) = \sum_{i=0}^{N-1} f(\zeta_N^i) t^i$$

- 自明な for loop アルゴリズムで $\mathcal{O}(N^2)$
- 高速フーリエ変換で $\mathcal{O}(N \log N)$

基本アイディア：問題のサイズを半分にして，
再帰的に解く（分割統治法）

高速フーリエ変換

多項式 $f(x) = \sum_{i=0}^{N-1} c_i x^i$ に対して,

$$f_0(x) = \sum_{i=0}^{\frac{n}{2}-1} c_{2i} = c_0 x^0 + c_2 x^1 + c_4 x^2 + \ldots,$$

$$f_1(x) = \sum_{i=0}^{\frac{n}{2}-1} c_{2i+1} = c_1 x^0 + c_3 x^1 + c_5 x^2 + \ldots$$

とすると,

$$f(x) = f_0(x^2) + x f_1(x^2)$$

が成り立ち, f_0 と f_1 はそれぞれ $\frac{N}{2}$ 次以下

高速フーリエ変換

$\hat{f}$ を求めるには,

$$f(\zeta_N^0), f(\zeta_N^1), \ldots, f(\zeta_N^{N-1})$$

を求める必要があった.

$f(x) = f_0(x^2) + xf_1(x^2)$ より,

$$\boxed{f_0(\zeta_N^0), f_0(\zeta_N^2), \ldots, f_0(\zeta_N^{2(N-1)})}$$

$$\boxed{f_1(\zeta_N^0), f_1(\zeta_N^2), \ldots, f_1(\zeta_N^{2(N-1)})}$$

を求めれば良い.

高速フーリエ変換

$\zeta_N^2 = \zeta_{N/2}$ より,

$$\boxed{f_0(\zeta_N^0), f_0(\zeta_N^2), \ldots, f_0(\zeta_N^{2(N-1)})}$$

$$\boxed{f_1(\zeta_N^0), f_1(\zeta_N^2), \ldots, f_1(\zeta_N^{2(N-1)})}$$

は,

$$\boxed{f_0(\zeta_{N/2}^0), f_0(\zeta_{N/2}^1), \ldots, f(\zeta_{N/2}^{N-1})}$$

$$\boxed{f_1(\zeta_{N/2}^0), f_1(\zeta_{N/2}^1), \ldots, f_1(\zeta_{N/2}^{N-1})}$$

と同じ.

高速フーリエ変換

性質 1: $\zeta_{N/2}^{i} = \zeta_{N/2}^{i+N/2}$ より,

$$f_0(\zeta_{N/2}^{0}), f_0(\zeta_{N/2}^{1}), \ldots, f_0(\zeta_{N/2}^{N/2-1}), f_0(\zeta_{N/2}^{0+N/2}), f_0(\zeta_{N/2}^{1+N/2}), \ldots, f_0(\zeta_{N/2}^{N/2-1+N/2}),$$

前半と後半が同じなので，前半の

$$f_0(\zeta_{N/2}^{0}), f_0(\zeta_{N/2}^{1}), \ldots, f_0(\zeta_{N/2}^{N/2-1}),$$

だけを求めれば良い

高速フーリエ変換

- サイズが半分の同じ問題を二つ解けば良い

元問題：

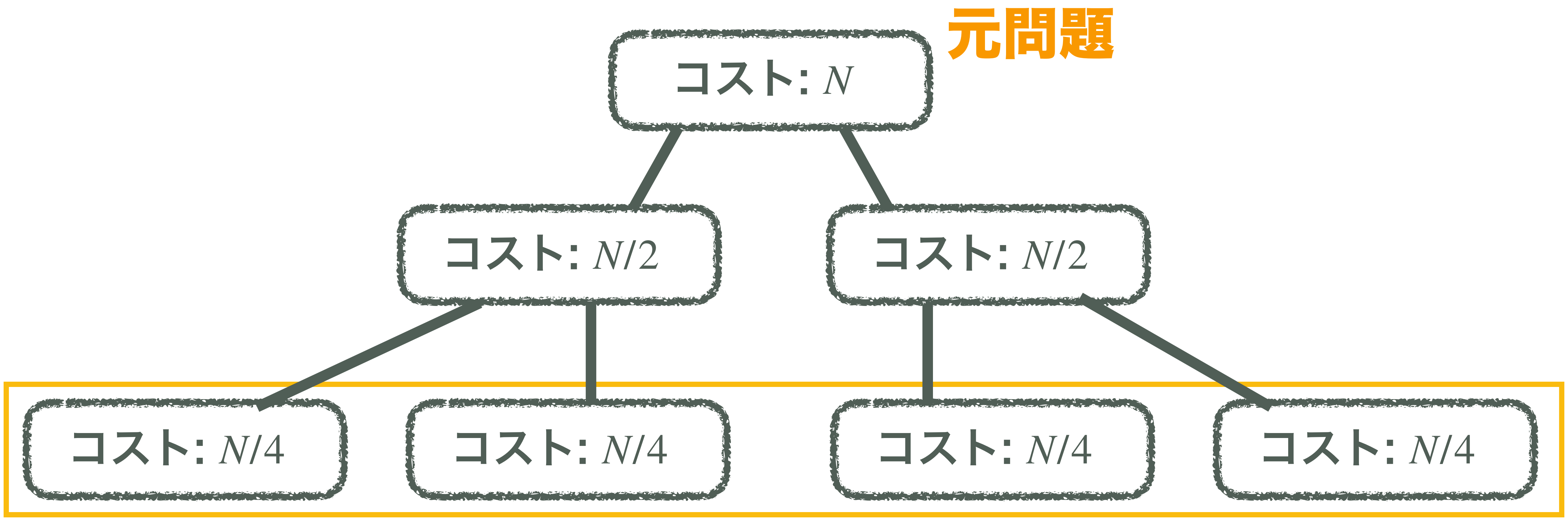
$$f(\zeta_N^0), f(\zeta_N^1), \ldots, f(\zeta_N^{N-1}) \text{ を求める}$$

分割後の問題：

$$f_0(\zeta_{N/2}^0), f_0(\zeta_{N/2}^1), \ldots, f_0(\zeta_{N/2}^{N/2-1}),$$

$$f_1(\zeta_{N/2}^0), f_1(\zeta_{N/2}^1), \ldots, f_1(\zeta_{N/2}^{N/2-1}), \text{ を求める}$$

計算量解析の気持ち



総和は N

高さが $\log N$ で，各段のコストの総和が N なので，

全体の時間計算量は $\mathcal{O}(N \log N)$

NTT ってなに？

- 数論変換（number theoretic transform, NTT）：
  - $p = u \times 2^N + 1$ を mod とした環の上で FFT をする手法

- ζ_N は，下記 2 つの性質があったので FFT が動作した

性質 1：$\zeta_N^i = \zeta_N^{i+N}$

性質 2：$$\sum_{i=0}^{N-1} \zeta_N^{i(j-k)} = \begin{cases} N & \text{if } j \equiv k \mod N, \\ 0 & \text{otherwise}. \end{cases}$$

- これらの性質を満たすものは他にあるか？**ある**

NTT の概要

- $998244353 = 119 \times 2^{23} + 1$ などの，
  特殊な素数 $p = u \times 2^e + 1$ 上で行う FFT のこと
- $2^e = N$ とおくと，N 要素の FFT ができる
- 複素数を用いた FFT と違い，**誤差が出ないことがメリット**
- 有名な素数に対しては，最小の原始根がすでに知られている

p	$u \times 2^e + 1$	16 進表記	最小の原始根
998244353	$119 \times 2^{23} + 1$	0x3b800001	3
163577857	$39 \times 2^{22} + 1$	0x9c00001	23
167772161	$5 \times 2^{25} + 1$	0xa000001	3
469762049	$7 \times 2^{26} + 1$	0x1c000001	3

引用：整数論テクニック集

フェルマーの小定理

フェルマー（Fermat）の小定理：

p を素数とし，a を p で割り切れない整数とすると

$$a^{p-1} \equiv 1 \mod p$$

が成り立つ

原始根とは？

原始根

p を法として，a が $p-1$ 乗して初めて 1 と合同になるとき，

a を p の**原始根**という

- 原始根の定義と周期性より，

  $\{1, a, a^2, \ldots, a^{p-2}\} = \{1, 2, 3, \ldots, p-1\}$ が成り立つ.

参考：原始根

原始根の例

剰余 a	1	2	3	4	5	6	7	8	9	10	11	12
a^2		4	9	3	12	10	10	12	3	9	4	1
a^3		8	1	12	8	8	5	5	1	12	5	
a^4		3		9	1	9	9	1		3	3	
a^5		6		10		2	11			4	7	
a^6		12		1		12	12			1	12	
a^7		11				7	6				2	
a^8		9				3	3				9	
a^9		5				5	8				8	
a^{10}		10				4	4				10	
a^{11}		7				11	2				6	
a^{12}	1	1	1	1	1	1	1	1	1	1	1	1

引用：原始根

NTT で使う原始根

$p = u \times 2^e + 1$ の原始根を g とする.

$g^{p-1} = g^{u \times 2^e}$ は初めて 1 と合同になるので,

$(g^u)^{2^e}$ とみると，g^u は 2^e 乗して初めて 1 と合同になる.

$2^e = N$ とすると，**以下の二つの性質が共に成り立つ！**

性質 1 ：$(g^u)^i = (g^u)^{i+N}$

性質 2 ：$\displaystyle\sum_{i=0}^{N-1} (g^u)^{i(j-k)} = \begin{cases} N & \text{if } j \equiv k \mod N, \\ 0 & \text{otherwise}. \end{cases}$

g^u の性質１証明（説明？）

性質１：$(g^u)^i = (g^u)^{i+N}$

g^u は N 乗して初めて 1 になる.

余りは周期性があるので，下のような感じになる.

$$(g^u)^0 = (g^u)^N = (g^u)^{2N} = \dots$$

$$(g^u)^1 = (g^u)^{1+N} = (g^u)^{1+2N} = \dots$$

$$\vdots$$

$$(g^u)^{N-1} = (g^u)^{N-1+N} = (g^u)^{N-1+2N} = \dots$$

g^u の性質 2 証明（説明？）

性質 2： $\displaystyle\sum_{i=0}^{N-1}(g^u)^{i(j-k)} = \begin{cases} N & \text{if } j \equiv k \mod N, \\ 0 & \text{otherwise}. \end{cases}$

- 証明：

(1) $j \equiv k \mod N$ のとき，$j - k = Nm$ と置ける．

$$（与式） = \sum_{i=0}^{N-1}(g^u)^{Nim} = \sum_{i=0}^{N-1} 1 = N$$

(2) $j \not\equiv k \mod N$ のとき，

$$\frac{1-((g^u)^{j-k})^N}{1-(g^u)^{j-k}} = \frac{1-((g^u)^N)^{j-k}}{1-(g^u)^{j-k}} = 0 \text{ より，}\quad（与式） = 0$$

まとめ

- 998244353 などの特殊な素数上で FFT ができた！
- （御免なさい実装はできていません）
- 僕の理解，あってますか？