7 - Schonhage-Strassen

Saturday, 22 June 2024 10:51

Mnożenie dwóch liczb binarnych.

Chcemy wykonać NTT pewnym $\mathbb{Z}_p$,

dla na tyle dużego p, żeby mod nie miało znaczenia.

Co musimy potrafić robić?

- branie reszty mod p
- dodawanie mod p
- mnożenie przez w mod p (składanie A_0, A_1)
- mnożenie dwóch liczb w $\mathbb{Z}_p$ ($A_i \cdot B_i = C_i$)
- dzielenie przez N (jedno, po inverse)

P nie musi być pierwsze, wystarczy aby istniało w rzędu N.

Weźmy $p = 2^m + 1$ oraz $w = 2^\alpha$, aby $w^N = 1$

Wtedy wszystkie operacje się upraszczają.

fastmul (U, V, N) - mnoży dwie N-cyfrowe liczby mod $2^N + 1$

Rozbijamy U i V na bloki $\sim N^{\frac{1}{2}}$ cyfr.

$U = u_0 + u_1 \cdot 2^b + \dots + u_{t-1} \cdot 2^{(t-1)b}$ t, b to potęgi 2.

$V = v_0 + v_1 \cdot 2^b + \dots + v_{t-1} \cdot 2^{(t-1)b}$

Czyli mnożymy dwie t-cyfrowe liczby w bazie 2^b.

Chcemy je policzyć w $\mathbb{Z}_p$ dla $p = 2^{b'} + 1$. $b' \approx 2b + \log t$; $t \mid b'$

Dlatego, że możemy zabrać $w = 2^{2b'/t} \mod p$.

Teraz robimy NTT i w mnożeniu $U_i \cdot V_i$ (liczby mod $2^{b'} + 1$)

wywołujemy się rekurencyjnie.

Po dopchaniu „technicznych szczegółów" dostajemy $O(N \log N \log\log N)$ □