Math 288X: Algorithms in Algebra and Number Theory

Fall 2021

Problem set #2

Let R be a ring. We assume we can do arithmetic in R in time $\mathcal{O}(1)$ throughout the problem set.

Problem 1. Show that we can compute the convolution of two tuples $a, b \in \prod_{\mathbb{Z}/n\mathbb{Z}} R$ in time $\mathcal{O}(n \log n \log\log n)$ on an $\mathcal{O}(\log n)$-bit RAM.

Problem 2 (Rader's FFT)**.** Let p be a prime number.

a) Show that you can find a generator g of the (cyclic) group $\mathbb{F}_p^\times$ in time $\mathcal{O}(p)$ on an $\mathcal{O}(\log p)$-bit RAM.

b) Given a root $\zeta_p \in R$ of the cyclotomic polynomial ϕ_p and a tuple $a = (a_i)_i \in \prod_{i \in \mathbb{Z}/p\mathbb{Z}} R$, show that you can compute the Fourier transform $b = \mathcal{F}_{\zeta_p}(a)$ in time $\mathcal{O}(p \log p \log\log p)$ on an $\mathcal{O}(\log p)$-bit RAM.

   Hints: $b_{g^j} = a_0 + \sum_{i \in \mathbb{Z}/(p-1)\mathbb{Z}} a_{g^i} \zeta_p^{g^{i+j}}$. The second sum looks like the convolution of two tuples in $\prod_{i \in \mathbb{Z}/(p-1)\mathbb{Z}} R$.

Problem 3. Show that you can multiply two binary integers with less than n bits in time $\mathcal{O}(n)$ on an $\mathcal{O}(\log n)$-bit RAM without using the intrinsic multiplication, division, or modulo operation on numbers in $\{0, \ldots, 2^b - 1\}$. (In other words: on a "restricted" RAM, where the fundamental operations $r_i := r_j \cdot r_k$, $r_i := \lfloor r_j / r_k \rfloor$, $r_i := r_j \bmod r_k$ are forbidden.)
Hint: Precomputation.

Problem 4. a) Let $B = \bigsqcup_{n \geqslant 0} \{0,1\}^n$ be the set of binary strings. We denote the length of a string s by $l(s)$. Think of a (natural) binary representation of a rational number, i.e. a subset $B' \subseteq B$ and a surjective function $\rho : B' \to \mathbb{Q}$, which satisfies the following properties:

i) For $x, y \in B'$, you can compute some $z \in B'$ satisfying $\rho(x) + \rho(y) = \rho(z)$ and $l(z) \leqslant l(x) + l(y)$ in time $\mathcal{O}(n)$ on an $\mathcal{O}(\log n)$-bit RAM, where $n = l(x) + l(y)$.

ii) For $x, y \in B'$, you can compute some $z \in B'$ satisfying $\rho(x) \cdot \rho(y) = \rho(z)$ and $l(z) \leqslant l(x) + l(y)$ in time $\mathcal{O}(n)$ on an $\mathcal{O}(\log n)$-bit RAM, where $n = l(x) + l(y)$.

iii) For $x \in B'$, you can compute some $z \in B'$ satisfying $-\rho(x) = \rho(z)$ and $l(z) = l(x)$ in time $\mathcal{O}(n)$ on an $\mathcal{O}(\log n)$-bit RAM, where $n = l(x)$.

iv) For $x \in B'$ with $\rho(x) \neq 0$, you can compute some $z \in B'$ satisfying $\rho(x)^{-1} = \rho(z)$ and $l(z) = l(x)$ in time $\mathcal{O}(n)$ on an $\mathcal{O}(\log n)$-bit RAM, where $n = l(x)$.

v) For $x \in B'$, you can determine whether $\rho(x) = 0$ in time $\mathcal{O}(n)$ on an $\mathcal{O}(\log n)$-bit RAM, where $n = l(x)$.

b) Do the same as in a), but with $\mathbb{Q}$ replaced by the ring of $k \times k$-matrices with entries in $\mathbb{Q}$. (The constants in $\mathcal{O}(\cdot)$ may depend on k.)