COL202: Discrete Math **Tutorial 3**

Instructor: Venkata Koppula Release date: 23rd August, 2024

The solutions for the ($\star$) marked problems must be submitted at the start of the tutorial. The ($\blacklozenge$) marked problems will be discussed in the tutorial (if time permits, you can also discuss the other problems).

Notations. For any natural number $n \geq 2$, let $\mathbb{Z}_n$ denote the set of natural numbers $\{0, 1, \ldots, n-1\}$. We define operations $+_n$ and $\times_n$ on $\mathbb{Z}_n$ as follows:

- for any $a, b \in \mathbb{Z}_n$, $a +_n b = (a + b) \bmod n$.
- for any $a, b \in \mathbb{Z}_n$, $a \times_n b = (a \cdot b) \bmod n$.

For the rest of this tutorial, p denotes a prime number. We define polynomials over $\mathbb{Z}_p$ similar to how we defined polynomials over $\mathbb{Q}$.

Definition 1. *For any prime p, let $\mathbb{Z}_p[x]$ denote the set of polynomials with coefficients from $\mathbb{Z}_p$. Given a polynomial $f(x) \in \mathbb{Z}_p[x]$, we say that $\alpha \in \mathbb{Z}_p$ is a root of $f(x)$ if $f(\alpha) = 0$.*

Analogous to what you proved in the quiz, one can prove the following lemma for $\mathbb{Z}_p[x]$.

Lemma 1. *Let p be a prime. For any non-zero polynomial $f(x) \in \mathbb{Z}_p[x]$ of degree $d \geq 1$, $\alpha \in \mathbb{Z}_p$ is a root of $f(x)$ if and only if $(x - \alpha)$ divides $f(x)$.*

Lemma 2 (Bézout's Lemma)**.** *For any natural numbers a, b, there exist integers x, y such that*

$$x \cdot a + y \cdot b = \gcd(a, b).$$

1 ($\star$) Tutorial Submission Problems

There is only one submission problem this time, and this is one of the questions from Quiz 1. Please submit a detailed proof for the same (even if you received full score for this problem in the quiz).

An infinite set $\mathcal{S}$ is countable if there exists a bijection between $\mathbb{N}$ and $\mathcal{S}$. If there exists an injective function $f : \mathbb{N} \to \mathcal{S}$ but there does not exist an injective function $g : \mathcal{S} \to \mathbb{N}$, then we say that the set is uncountable.
In class/tutorials, we proved that the following sets are countable: $\mathbb{Z}$, $\mathbb{N} \times \mathbb{N}$, $\bigcup_{i \in \mathbb{N}} \mathbb{N}^i$, $\mathcal{T}$ = set of infinite strings with finitely many ones, $\mathcal{F} = \{f : \{0,1\} \to \mathbb{N}\}$.
We also showed that the following sets are uncountable: $\mathbb{R}$, $\mathcal{S}$ = set of all infinite bit strings, $\mathcal{W}$ = set of all infinite bit strings with infinitely many occurrences of 11, $\mathcal{G} = \{f : \mathbb{N} \to \{0,1\}\}$.

Given any function $f : \mathbb{N} \to \mathbb{N}$, we say that f is non-increasing if for all $i, j \in \mathbb{N}$ such that $i < j$, $f(i) \geq f(j)$. Similarly, we say that f is non-decreasing if for all $i, j \in \mathbb{N}$ such that $i < j$, $f(i) \leq f(j)$. Consider the following two sets:

$$\mathcal{S}_1 = \{f : \mathbb{N} \to \mathbb{N} \text{ such that } f \text{ is non-increasing}\}$$

$$\mathcal{S}_2 = \{f : \mathbb{N} \to \mathbb{N} \text{ such that } f \text{ is non-decreasing}\}$$

Is $\mathcal{S}_1$ countably infinite? Is $\mathcal{S}_2$ countably infinite? You must have two claims, one for $\mathcal{S}_1$ and another for $\mathcal{S}_2$. Prove your claims by showing a bijection to one of the sets listed above.

2 Problems: Modular Arithmetic

2.1. (♦) **Chinese remainder theorem.** Let $m, n \in \mathbb{N}$ such that $\gcd(m, n) = 1$. Let $a \in \mathbb{Z}_m$ and $b \in \mathbb{Z}_n$. Prove that there exists a **unique** $x \in \mathbb{Z}_{m \cdot n}$ such that $x \bmod m = a$ and $x \bmod n = b$. Provide an algorithm for computing this x efficiently, given m, n, a, b.

Can we make a similar claim if m, n have a common factor greater than 1?

2.2. Solve the following problems using **modular arithmetic**:

a What is the last digit of 7^{100}? 1

b What is the value of $\left(2^{100} \cdot 3^{60}\right) \bmod 5$? 1

c For any natural number n, $n! = 1 \times 2 \times \ldots \times (n-1) \times n$. What is the remainder when $\sum_{i=1}^{100}(i)!$ is divided by 9? 0

d Prove or disprove: $(2^n + 6 * 9^n)$ is divisible by 7 for every $n \geq 0$. true

2.3. Prove that for any prime p, any $a \in \mathbb{Z}_p$ such that $a \neq 0$, there exists a **unique** number $b \in \mathbb{Z}_p$ such that $a \times_p b = 1$.

Is this true for non-prime moduli also? Let n be a composite modulus, and let $a \in \mathbb{Z}_n$ such that there exists $b \in \mathbb{Z}_n$ satisfying $a \times_n b = 1$. Are we guaranteed that for all $b' \neq b$, $a \times_n b' \neq 1$?

2.4. Fibonacci numbers are defined recursively, using the following recurrence: $F_1 = 1$, $F_2 = 1$, $F_n = F_{n-1} + F_{n-2}$ for all $n \geq 3$. Prove that for any $n \in \mathbb{N}$, $\gcd(F_n, F_{n+1}) = 1$.

2.5. Let $a, b \in \mathbb{N}$. Show that $\left(2^a - 1\right) \bmod \left(2^b - 1\right) = 2^{a \bmod b} - 1$.

Use this to prove that $\gcd\left(2^a - 1, 2^b - 1\right) = 2^{\gcd(a,b)} - 1$.

3 Problems: Polynomials over $\mathbb{Z}_p$

3.1. (♦) Let p be a prime. Prove that any non-zero polynomial $f(x) \in \mathbb{Z}_p[x]$ has at most d distinct roots, where d is the degree of $f(x)$.

3.2. **Error Detecting/Correcting Codes.** In this exercise, we will develop some of the theory behind error detecting/correcting codes. Error correcting codes are widely used in practice (e.g. QR codes).

Consider a communication channel where you can feed in data packets on one end, and the data packets are received on the other end. Each data packet is a number in $\mathbb{Z}_p$, where p is a prime. We say that a communication channel is t-lossy, for $t \in \mathbb{N}$, if the channel corrupts at most t of the data packets.[1]

[1] By corruption, we mean that the channel can replace certain packets with some other number in $\mathbb{Z}_p$. It does not rearrange the packets.

Definition 2. *Let $t \in \mathbb{N}$. A t-lossy communication channel takes as input $\mathbf{z} \in \mathbb{Z}_p^\ell$ (for some $\ell \in \mathbb{N}$), and outputs $\mathbf{z}' \in \mathbb{Z}_p^\ell$, with the following guarantee: let $\mathbf{z} = (z_1, \ldots, z_\ell)$ and $\mathbf{z}' = (z_1', \ldots, z_\ell')$, then*

$$\left|\left\{i : z_i \neq z_i'\right\}\right| \leq t$$

We want to transmit an n-packet message (that is, our message is $\mathbf{m} \in \mathbb{Z}_p^n$). Clearly, if we transmit $\mathbf{m}$ directly over a t-lossy channel, then we cannot hope to recover $\mathbf{m}$, since the channel can corrupt t out of the n packets. Therefore, we will encode the message $\mathbf{m}$ using an encoding algorithm Encode. Let $\mathbf{z} = \mathsf{Encode}(\mathbf{m})$. This encoded message is sent over the t-lossy channel, and suppose we receive $\mathbf{z}'$ on the other end. We want to decode $\mathbf{z}'$ to recover $\mathbf{m}$, or at the very least, detect that there are corruptions.

Definition 3. *An error-detecting code for a t-lossy channel consists of an algorithm $\mathsf{Encode} : \mathbb{Z}_p^n \to \mathbb{Z}_p^\ell$, and an algorithm $\mathsf{Detect} : \mathbb{Z}_p^\ell \to \{0, 1\}$ such that the following guarantee holds: for all $\mathbf{m} \in \mathbb{Z}_p^n$, if $\mathbf{z} = \mathsf{Encode}(\mathbf{m})$, and $\mathbf{z}'$ is the resulting message after passing $\mathbf{z}$ through a t-lossy channel, then Detect outputs 1* ***if and only if*** *$\mathbf{z}' = \mathbf{z}$.*

1. Here is the most basic error detecting code for a t-lossy channel: map every element in $\mathbb{Z}_p^n$ to an element in $\mathbb{Z}_p^{(t+1)n}$ by making $(t+1)$ copies. What is the corresponding detection algorithm? Would it suffice to make just t copies of the input message?

2. Can we have a more efficient encoding algorithm, one that maps $\mathbb{Z}_p^n$ to $\mathbb{Z}_p^\ell$ for some $\ell \approx (n+t)$, so that we can still detect corruptions in communication? Prove correctness of your algorithm.

   Hint: polynomials.

Definition 4. *An error-correcting code for a t-lossy channel consists of an algorithm $\mathsf{Encode} : \mathbb{Z}_p^n \to \mathbb{Z}_p^\ell$, and an algorithm $\mathsf{Decode} : \mathbb{Z}_p^\ell \to \mathbb{Z}_p^n \cup \{\perp\}$ such that the following guarantee holds: for all $\mathbf{m} \in \mathbb{Z}_p^n$, if $\mathbf{z} = \mathsf{Encode}(\mathbf{m})$, and $\mathbf{z}'$ is the resulting message after passing $\mathbf{m}$ through a t-lossy channel, then $\mathsf{Decode}(\mathbf{z}')$ outputs $\mathbf{m}$.*

3. Here is the most basic error detecting code for a t-lossy channel: map every element in $\mathbb{Z}_p^n$ to an element in $\mathbb{Z}_p^{(2t+1)n}$ by making $(2t+1)$ copies. What is the corresponding decoding algorithm? Would it suffice to make fewer copies of the input message?

4. Propose a better solution using polynomials where $\ell \approx n + 2t$.

   (This is more challenging than error detection.)

This is version 1.0 of the tutorial sheet. Let me know if something is unclear. In case of any doubt or for help regarding writing proofs, feel free to contact me or TAs.

Venkata Koppula - kvenkata@iitd.ac.in
Ananya Mathur - cs5200416@iitd.ac.in
Anish Banerjee - cs1210134@cse.iitd.ac.in
Eshan Jain - cs5200424@cse.iitd.ac.in
Mihir Kaskhedikar - cs1210551@iitd.ac.in
Naman Nirwan - Naman.Nirwan.cs521@cse.iitd.ac.in
Pravar Kataria - Pravar.Kataria.cs121@cse.iitd.ac.in
Shashwat Agrawal - csz248012@cse.iitd.ac.in
Subhankar Jana - csz248009@iitd.ac.in