CS 710: Complexity Theory Date: Apr. 2nd & 4th, 2024

Lec. 19, 20: Randomness: PTM, Hashing, MAXCUT, Isolation Lemma, BPP $\subseteq \Sigma_2^p \cap \Pi_2^p$

Instructor: Jin-Yi Cai Scribe: Hao Lin

1 Overview

In these two lectures [Cai03], we are going to talk about computational complexity on random algorithms. Here are some intriguing questions:

- Can we use computation models to capture random algorithms?
- What are randomness? Does there exist true randomness?
- Is derandomization universal? (i.e. to transform every random algorithm into a deterministic one)
- Does randomness give more power to simplify the complexity of problems? (Randomness vs. Hardness)
- What are the relationships between probability complexity classes and other classes we have visited?

We will also cover some preliminaries on *Inequalities and Bounds* and *Abstract Algebra* in Appendix.

2 Probabilistic Turing Machine and its Complexity Classes

A probabilistic Turing machine (PTM) syntactically is equivalent to a non-deterministic Turing machine (NTM) where there are also two transition functions δ_1 and δ_2. The difference is: for NTM, it magically guess the best transition function to pick at each point that eventually leads to an accepted state if exists; for PTM, it pick either one of them randomly according to a probabilistic function (can be uniform), which can lead to ”wrong” results.

Therefore, even for the same input and finite state control, it may give different results for various runs, or even not halt for certain runs. Then, how can we say if one language is accepted by this PTM? It turns out the definition is different from what we have encountered before, requiring some promises (which are not decidable, but we will not cover it here). It is related to the probability of getting the input accepted.

There are three kinds of basic probabilistic complexity classes that capture: two-sided error (BPP), one-sided error (RP) and zero error (ZPP). PP is something different and will be covered at the end of this section.

Definition 1 (BPP) BPP is called the bounded-error probabilistic polynomial-time class. A language L is said to be in BPP iff

$$\begin{aligned} x \in L &\Longrightarrow \Pr[x \text{ is accepted by PTM}] \geq \frac{2}{3} \\ x \notin L &\Longrightarrow \Pr[x \text{ is accepted by PTM}] \leq \frac{1}{3} \end{aligned}$$

Instead of saying the probability of rejecting by PTM in the negative case, we say the probability of being accepted is low. This is simply because there exists situations that PTM may not even halt.

Notice that the predicate above can be more mathematically replaced by $D(x, y) = 1$ defined as usual where y is the path in the PTM, $D(\cdot, \cdot)$ is a boolean predicate.

Definition 2 (RP) RP is called the randomized polynomial time classes, and its name is given because it is the earliest such classes. A language L is said to be in RP iff

$$\begin{aligned} x \in L &\Longrightarrow \Pr[x \text{ is accepted by PTM}] \geq \frac{1}{2} \\ x \notin L &\Longrightarrow \Pr[x \text{ is accepted by PTM}] = 0 \end{aligned}$$

Example: Randomized primality testing is in coRP, which means it will never wrongly claim a primary number to be composite; while a composite number have a chance to be correctly determined if we can find one of its factor in polynomial trials. ⊠

Remark 1 Notice again that the constant $\frac{1}{2}$ is also a flexible constant, which can be amplified towards both sides exponentially.

Definition 3 (ZPP) ZPP stands for zero-error probabilistic polynomial classes. The predicate D is now a polynomial time computable function $\{0,1\}^* \times \{0,1\}^* \longrightarrow \{0,1,?\}$. A language L is said to be in ZPP iff

$$\begin{aligned} x \in L &\Longrightarrow \forall y, D(x, y) = \{1, ?\} \wedge \Pr[D(x, y) = 1] \geq \frac{1}{2} \\ x \notin L &\Longrightarrow \forall y, D(x, y) = \{0, ?\} \wedge \Pr[D(x, y) = 0] \geq \frac{1}{2} \end{aligned}$$

In other words, the PTM will at least not make false negatives or positives.

2.1 Amplification of BPP

The number $\frac{2}{3}$ is some pretty flexible constant that we can then replaced to be at least $\varepsilon = \frac{1}{n^c}$ greater than $\frac{1}{2}$. And by constructing a new PTM M' which runs the original PTM M $2m+1$ times and accepts if at least $m+1$ times M accepts it, we can amplify this threshold to accept any input with exponentially small error $e^{-q(n)}$.

More precisely, let X_i be the 0-1 random variable of the i-th outcome of M, p be the probability to accept x by M, $S = \sum X_i - p$. Suppose $x \notin L$, then $p \leq \frac{1}{2} - \varepsilon$, we have

$$\Pr[M' \text{ accepts } x] = \Pr[\sum X_i \geq m+1] \leq \Pr[S \geq (2m+1)\varepsilon] \leq e^{-q(n)}$$

The last inequality comes from the Chernoff bound (see Appendix A) and a carefully selected large m.

Remark 2 It is the Chernoff bound showing that polynomial repetition gives exponentially small error, which further serves for many non-trivial results between randomness and determinism worked under polynomial time bound.

Remark 3 The ε cannot be exponentially small, since otherwise the gap between positive and negative cases are too small to amplify them with polynomial repetitions.

3 Universal Hashing Function Family

Here goes the topic on a formal definition of the widely seen and used hash function, which will also play a great role in random algorithms, derandomization and the proof of probabilistic complexity classes, notably the isolation lemma.

Definition 4 (`UHF Family`) The universal hashing function family is the set $\{h_s : U \to T\}_{s \in S}$, where S is an index set, U and T are finite sets. Furthermore, the USF family has to have the following "randomness" property: *if* $\forall x, y \in U, \forall \alpha, \beta \in T, x \neq y$, *then*

$$\Pr_{s \in S}[h_s(x) = \alpha \wedge h_s(y) = \beta] = \frac{1}{|T|^2}$$

Remark 4 The definition is based on the probability over USF Family, not a single USF, since the behavior for a single USF is determined, whereas we are looking at the collaborative behavior.

Theorem 1 (`Pairwise independence of UHF Family`) If we fix x, but treat s as the variable, then we define the random variable $Z_x(s) = h_s(x)$. We have $\{Z_x\}_{x \in U}$ is pairwise independent and uniformly distributed random variables on T.

Proof. Notice that if the hashing function is uniformly onto T, implies that we want to show $\forall x \in U, \alpha \in T$, $\Pr_{s \in S}[Z_x(s) = \alpha] = \frac{1}{|T|}$.

This is obtained by $\forall x, y \in U, y \neq x, \forall \alpha, \beta \in T$, we have

$$\Pr_{s \in S}[Z_x(s) = \alpha] = \sum_{\beta} \Pr_{s \in S}[Z_x(s) = \alpha \wedge Z_y(s) = \beta] = \frac{1}{|T|}$$

□

Example: $ax + b$ forms a UHF family for $a, b \in \mathbb{Z}/p$, where p is a prime, and $\mathbb{Z}/p = 0, 1, \ldots, p-1$ with $+$ and $\cdot$ form a finite field. ⊠

Proof. Let us treat a and b as the variables in a linear system

$$\begin{aligned} ax + b &= \alpha \\ ay + b &= \beta \end{aligned}$$

Since the determinant of this linear system is

$$\det \begin{pmatrix} x & 1 \\ y & 1 \end{pmatrix} = x - y \neq 0$$

the solution to a and b is unique, and hence $\forall x, y \in U, \forall \alpha, \beta \in T, x \neq y$, then

$$\Pr_{s \in S}[h_s(x) = \alpha \wedge h_s(y) = \beta] = \frac{1}{|T|^2}$$

□

Example: This can be generalized to any finite field $\mathrm{GF}(p^n)$. ⊠

4 MAXCUT

We know that the MINCUT problem has a dual problem MAXFLOW, which is found to have polynomial solution, and hence it is in P.

However, another counter part is the MAXCUT, which is proved to be NP-Hard from e.g. 3-SAT. Here we will show the "power" (or probably no extra "power") of random algorithm in approximation to one of the NP-Hard problems, i.e. MAXCUT, by comparing with deterministic approximation and progressively improving the results.

Definition 5 (`MAXCUT`) Give a graph $G = (V, E)$, we can make a disjoint vertex partition $V = V_1 \sqcup V_2$. A cut is defined to be the edge subset $C \subseteq E$ such that it contains all edges lying as the bridge between V_1 and V_2. MAXCUT is defined to be the maximum of the cut size $|C|$.

$$|C^*| = \max_{V_1} |C|$$

Algorithm 1 1/2 Approximation using Deterministic Algorithm

Let us partition the set E into n disjoint subsets according to vertex indices, $E_i = \{(v_i, v_j) \mid j \geq i\}$. The algorithm iteratively forms V_1 and V_2 by categorizing vertex starting from n down to 1. At each vertex i, we put it in V_1 if $|\{v_j \mid v_j \in V_2 \wedge (v_i, v_j) \in E_i\}| \geq |E_i|/2$; otherwise, put it in V_2.

```
procedure DetApproxMAXCUT($G = (V, E)$)
    $E_i = \{(v_i, v_j) \mid j \geq i\}$
    $V_1 \leftarrow \emptyset, V_2 \leftarrow \emptyset$
    for $i \leftarrow n$ to 1 do
        if $|\{v_j \mid v_j \in V_2 \wedge (v_i, v_j) \in E_i\}| \geq |E_i|/2$ then
            $V_1 \leftarrow V_1 \cup v_i$
        else
            $V_2 \leftarrow V_2 \cup v_i$
        end if
    end for
    return $V_1$, $V_2$
end procedure
```

Proof. For each vertex i, let $C_i = \{v_j \mid (v_j \in V - V_b, v_i \in V_b) \wedge (v_i, v_j) \in E_i\}$, by definition $C_i \subseteq E_i$ and $C_i \subseteq C$, $|C_i| \geq |E_i|/2$. Since all E_i are disjoint partitions of E, we have

$$|C| = \sum_i |C_i| \geq \sum_i |E_i|/2 \geq \sum_i |C_i^*|/2 = |C^*|/2$$

□

The "power" of the randomized algorithm can be its astonishing elegance:

Algorithm 2 1/2 Approximation using a Simple Randomized Algorithm

```
procedure RandApproxMAXCUT($G = (V, E)$)
    $V_1 \leftarrow \emptyset, V_2 \leftarrow \emptyset$
    for $i \leftarrow 1$ to $n$ do
        if $p \sim \mathcal{U}(0, 1) > 0.5$ then
            $V_1 \leftarrow V_1 \cup v_i$
        else
            $V_2 \leftarrow V_2 \cup v_i$
        end if
    end for
    return $V_1$, $V_2$
end procedure
```

Proof. For each edge $e_{ij} \in C$, the probability is

$$\Pr[v_i \in V_b \wedge v_j \in V - V_b] = \sum_b \Pr[v_i \in V_b] \cdot \Pr[v_j \in V - V_b] = 1/2$$

so $\mathbb{E}[|C|] \geq \mathbb{E}[|E|]/2 \geq \mathbb{E}[|C^*|]/2$. □

4.1 Derandomizing MAXCUT Algorithm using Universal Hashing Function

The procedure is fairly simple: take the least bit of the hashed value for categorization. Specifically, let us pick k such that $2^{k-1} < |V| \leq 2^k$, which forms a $GF[2^k]$, and hence our UHF family under $ax + b$ for $a, b \in GF[2^k]$. For each vertex v_i, we apply all the hash functions to $x = i$ to get the least bit, and take the majority on the least bit.

There are $O(|V|^2)$ different hash functions in this UHF family, and hence overall is still a polynomial deterministic algorithm. In addition, it can be accelerated by parallel computation.

Remark 5 Notice that all derandomization on discrete finite seeds can be done in this manner according to Nisan and Wigderson [NW94], with an exponential slow down in terms of hashing function bit length (e.g. here is k instead of $|V|$).

4.2 Goemans-Williamson Algorithm: An 87.8% Approximation to MAXCUT

We will reduce this problem to another NP-Hard problem, the quadratic programming problem, and then relax it to semi-definite programming, combined with randomization, to get an approximation for about 12% error.

Let us assign $x_i = 1$ if v_i is in V_1, and $x_i = -1$ otherwise. Then the max cut become a quadratic programming:

$$
\begin{aligned}
\max_{\{x_i\}\subseteq\{-1,1\}^n} \frac{1}{4}\sum_{e_{ij}}(x_i - x_j)^2 \\
\text{s.t. } x_i^2 = 1
\end{aligned}
$$

Then linearize it with the "intention" that $y_{ij} = x_i x_j$, and we find a relaxed problem

$$
\begin{aligned}
\max_{\{y_{ij}\in\{-1,1\}^n} \frac{1}{4}\sum_{e_{ij}}(y_{ii} + y_{jj} - 2y_{ij}) \\
\text{s.t. } y_{ii} = 1 \\
v^T Y v \geq 0,\ \forall v \in \mathbb{R}^n
\end{aligned}
$$

Notice that this is not equivalent to the original problem any more, since in the original problem, the rank of $Y = XX^T$ should be 1 since we have a rank 1 matrix of X, while here it may be not. Except for the rank, the relax Y preserve all the properties: symmetric and positive semi-definite. It is known that there exists polynomial time to solve semi-definite programming, which we will not cover here but just assume it works.

Therefore, we can find Y^* such that we maximize the value, let say M^*. As we know $Y^* = UU^T$, where $U = \begin{pmatrix} u_1 \\ u_2 \\ \cdots \\ u_n \end{pmatrix}$, where $u_i \in \mathbb{R}^n$. We have $M^* = \frac{1}{4}\sum_{e_{ij}} ||u_i - u_j||^2$. Now, as we know Y^* is not likely to be rank 1, it is less likely to find a correspondent solution given U. However, we can perform a random strategy to determine X based on U, with a slight deviation from the upper bound M^* (though this upper bound may not be strict).

The random strategy is fairly simple: random choose a hyperplane Π to split u_i into two partition, and hence the cut.

How good is this partition? The cut size is then $M = \sum_{e_{ij}} \Pr[\Pi \text{ separates } u_i, u_j]$. With the help of simple geometry, we know $\Pr[\Pi \text{ separates } u_i, u_j] = \theta_{ij}/\pi$, and $||u_i - u_j||^2 = 4\sin^2\theta_{ij}/2$. So, we have:

$$
\begin{aligned}
\frac{M}{M^*} = \frac{\theta_{ij}}{\pi \sin^2 \theta_{ij}/2} = f(\theta_{ij}) \\
f_{min} \approx f(2.33) \approx 0.878
\end{aligned}
$$

Therefore, this Goemans-Williamson algorithm achieves at least 87.8%-approximation to the optimal value.

5 Isolation Lemma

Back to the abstract properties and theorems, we continue to discover the power of universal hashing function family: the UHF family behaves somehow randomly according to the definition, whereas it only takes $2n$ bits.

How can we utilize this to do something more than the derandomization we have just seen? Approximate the size of a set.

The idea is if the domain of UHF is too small compared to the range, the "random" mapping will most likely to be a one-on-one, i.e. no "collision". On the other hand, if the domain is too big compared to the range, most of the value will be squeezed and map onto the same value, i.e. many "collisions". We call S in the first case "thin", in the second case "fat".

Definition 6 Let $\mathcal{H}$ be the UHF family, with each UHF $h : S \to T$. $x, y \in S$ is said to be a collision under h if $h(x) = h(y)$.

h is said to isolate x if $\forall y \in S, x \neq y$, $h(x) \neq h(y)$.

h is said to isolate S if $\forall x \in S, h$ isolates x.

$\mathcal{H}$ is said to isolate x if $\exists h \in \mathcal{H}$, h isolates x.

$\mathcal{H}$ is said to isolate S if $\forall x, \exists h \in \mathcal{H}$, h isolates x.

Theorem 2 `(Isolation Lemma)` Let $\mathcal{H}$ be the UHF family, with each UHF $h : S \to T$, and pick $H^r = \{h_1, \ldots, h_r\}$ randomly from $\mathcal{H}$.

(1) If $|S| \geq r|T|$, then

$$\Pr_{H^r}[H^r \text{ isolates } S] = 0$$

(2) If $|S|^{r+1} \leq |T|^r$, then

$$\Pr_{H^r}[H^r \text{ isolates } S] > 1 - \frac{|S|^{r+1}}{|T|^r}$$

Proof. In case (1), S is "fat". Each h can isolate at most $|T| - 1$ elements by performing a one-to-one mapping on the first $|T| - 1$ elements and dumping the remained elements to a single value $|T|$. So, H^r can isolate at most $r(|T| - 1)$ elements. No H^r can isolate S when $|S| \geq r|T|$.

In case (2), S is "thin". We prove the converse situation where H^r fails to isolate S. For any $x \in S$

$$\Pr_{h \in \mathcal{H}}[h \text{ does not isolate } x] \leq \sum_{y \in S - \{x\}} \Pr_{h \in \mathcal{H}}[x \text{ collides with } y \text{ under } h]$$

$$\frac{|S|}{|T|}$$

Then we have a collective behavior

$$\Pr_{H^r \in \mathcal{H}}[H^r \text{ does not isolate } x] < \left(\frac{|S|}{|T|}\right)^r$$
$$\Pr_{H^r \in \mathcal{H}}[H^r \text{ does not isolate } S] < \frac{|S|^{r+1}}{|T|^r}$$

□

6 Sipser–Lautemann Theorem: BPP $\subseteq \Sigma_2^p \cap \Pi_2^p$

Let us take a look at how isolation lemma can be used to prove a non-trivial theorem.

Sipser first made a significant contribution to unveil the connection between PTM and PH, showing that BPP $\subseteq$ PH. Then, Lautemann improved the bound to be BPP $\subseteq \Sigma_2^p \cap \Pi_2^p$.

Theorem 3 BPP $\subseteq \Sigma_2^p \cap \Pi_2^p$

Proof. Since BPP is symmetric and hence closed under complement, we only need to prove BPP $\in \Pi_2^p$.

With the amplification of BPP, we can have $L \in$ BPP such that given a polynomial computable predicated $D(\cdot,\cdot)$, for any input $x \in \{0,1\}^n$,

$$x \in L \Leftrightarrow \Pr_{y \in \{0,1\}^m}[D(x,y) = 1] \geq \frac{1}{2}$$
$$x \notin L \Leftrightarrow \Pr_{y \in \{0,1\}^m}[D(x,y) = 1] \leq \frac{1}{4m}$$

W.o.l.o.g, let us assume m be the power of 2, pick $r = m$, $|T| = 2^m/(2m)$. Let the witness set $W_x = \{y \mid D(x,y) = 1\}$.

If $x \in L$, we have $|W_x| \geq 2^{m-1} = r|T|$. By isolation lemma, no H^r can isolate W_x.

If $x \notin L$, isolation lemma gives that $\Pr_{H^r}[H^r \text{ isolates } W_x] \geq 1 - 1/(4m)$.

In particular, let us take the $x \notin L$ case,

$$\begin{aligned} x \notin L &\Leftrightarrow \exists h_1, \ldots, h_r [H^r \text{ isolates } x] \\ &\Leftrightarrow \exists h_1, \ldots, h_r \forall y \in W_x \forall y' \in W_x - \{y\}[h_1(y) \neq h_1(y') \vee \cdots \vee h_r(y) \neq h_r(y')] \end{aligned}$$

So, $L \in \Pi_2^p$.

□

Appendix

A Preliminary Inequalities and Bounds

The Markov's inequality connects the CDF (cumulative density function) of the tails to the expectation.

Theorem 4 `(Markov's Inequality)` Let $X \geq 0$, and $0 \leq E[X] < \infty$, then

$$\Pr[X \geq a] \leq \frac{E[X]}{a}$$

The Chebyshev's inequality concerns on the deviation to the expectation.

Theorem 5 `(Chebyshev's Inequality)`

$$\Pr[|X - E[X]| \geq a] \leq \frac{\mathrm{Var}[X]}{a^2}$$

Proof. By expanding the definition $\mathrm{Var}[X] = E[(X - E[X])^2] = E[X^2] - E[X]^2$ and using Markov's inequality. □

The Chernoff bound takes a step more forward, by concerning on the deviation of the sum of n i.d r.v. to the expectation.

Theorem 6 `(Chernoff Bound)` Let $X_i \in \{-1, 1\}$ with equal probability, and let $S_n = \sum X_i$, we have

$$\Pr[S \geq a] \leq e^{-a^2/2n}$$

Proof. The proof starts from the trick of computing the expectation of the $e^{\lambda X_i}$, and hence $E[e^{\lambda X_i}] = \frac{e^{\lambda}+e^{-\lambda}}{2} = \cosh \lambda \leq e^{\lambda^2/2}$. The latter inequality comes from the Taylor expansion. And then goes the Markov's inequality and also the monotone property of e^n:

$$\Pr[S_n \geq a] \leq \Pr[e^{\lambda S_n} \geq e^{\lambda a}] \leq \frac{E[e^{\lambda S_n}]}{e^{\lambda a}} \leq e^{-a^2/2n}$$

□

Remark 6 This also gives the central limit theorem by taking $a = \alpha\sqrt{n}$ and do the integral.

$$\lim_{n \to \infty} [S_n \geq \alpha\sqrt{n}] = \int_{\alpha}^{\infty} \frac{1}{\sqrt{2\pi}} e^{x^2/2}\, dx$$

Remark 7 The Chernoff bound as well as the broader field, concentration theory, contains certain amplification from polynomial entities to a exponentially small bound. This amplification can lead to incredible results, including the Nisan-Wigderson pseudorandom generator.

A variant to $X_i \in \{0,1\}$, where $\Pr[X_i = 1] = p_i$, $\Pr[X_i = 0] = 1 - p_i$, goes

Corollary 1 Let $p = \sum p_i/n$, $S_n = \sum X_i - pn$ For $\Delta > 0$,

$$\Pr[S_n \geq \Delta] \leq e^{-2\Delta^2/n}$$

B Preliminary Abstract Algebra [Gal21]

Definition 7 (Group) Group G is a set S with a binary operation (a function on $S \times S \to S$, usually denoted with ab) with three properties:

- Associativity: $(ab)c = a(bc)$
- Identity: $\exists e \in S, ea = ae = e$
- Inverses: $aa^{-1} = a^{-1}a = e$

Abelian Group satisfies an additional property:

- Commutativity: $ab = ba$

Example: Common Groups:

- Z_n with addition $\bmod n$, whose identity is 0, inverse is $n - j$ for $j > 0$
- $GL(2, R) = \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} \middle| a, b, c, d \in \mathbb{R}, ad - bc \neq 0 \right\}$ with matrix multiplication, whose identity is $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$.
  - This non-Abelian group is called generalized linear group of 2×2 matrices over $\mathbb{R}$.
  - Notice that the determinant has to be non-zero; otherwise, the matrix does not have inverse.
- $U(n) = \{a \mid \gcd(a, n) = 1,\ a \in \mathbb{Z}_n^*\}$ with multiplication $\bmod n$ is an Abelian group, whose identity is 1, inverse exists and is (by definition) the solution to $ax \mod n = 1$
  - If n is prime, then $U(n) = \mathbb{Z}_n^*$
  - The existence of the multiplicative inverse is due to Euler, proving that $\gcd(a, b) = 1 \Leftrightarrow ab \mod n = 1$

⊠

Definition 8 (Order) Order of the group $|G|$ is the number of elements in the group G.

Order of the element $g \in G$ is the smallest positive number n such that $g^n = e$.

Definition 9 (Subgroup) Subgroup $H \leq G$ iff H is a subset of G and is closed under the operation of G

Definition 10 `(Ring)` Ring R is an extension to Group with two operations (usually called addition and multiplication), satisfying the following six properties:

- Associativity on Addition: $(a+b)+c = a+(b+c)$
- Commutativity on Addition: $a+b = b+a$
- Additive Identity: $\exists 0 \in R$, $a+0 = 0+a = a$
- Additive Inverse: $(-a)+a = 0$
- Associativity on Multiplication: $(ab)c = a(bc)$
- Distributive Law: $a(b+c) = ab+ac$ and $(b+c)a = ba+ca$

Notice that there are three more optional properties for multiplication

- Commutativity on Multiplication: we call it a *commutative ring*: $ab = ba$
- Multiplicative Identity: we call it a *unity*: $\exists e \in R$, $ea = a$
- Multiplicative Inverse: it must satisfy the above two properties, and we call it *units*: $\forall a \neq 0$, $\exists a^{-1}$, $a^{-1}a = aa^{-1} = e$

Definition 11 `(Integral Domain)` Integral Domain D is an alias to a kind of ring: commutative ring with unity and "no zero-divisor", i.e. $\forall a \neq 0, b \neq 0$, $ab \neq 0$

Example: $\mathbb{Z}$ is an integral domain

$\mathbb{Z}_p$ is an integral domain iff p is prime ⊠

Definition 12 `(Field)` Field F is an alias to a kind of ring: commutative ring with unity in which every non-zero element is a unit.

Example: $\mathbb{Z}_p$ is a field iff p is prime

$\mathbb{Z}[x] = \{a_n x^n + \cdots + a_1 x + a_0 \mid a_i \in \mathbb{Z}\}$ is a commutative ring with unity $f(x) = 1$, but not a field since $f(x) = x$ has no multiplicative inverse. It is called polynomial rings. ⊠

Theorem 7 Every fields are integral domain

Theorem 8 Every finite integral domains are fields

Theorem 9 If D is an integral domain, then $D[x]$ is also an integral domain. This directly gives $\mathbb{Z}[x]$ is an integral domain.

Theorem 10 If p is prime, then $\forall n$, $GF[p^n]$ is a unique, up to isomorphism, Galois field (an alias to the finite field) of order p^n (i.e. $|GF[p^n]| = p^n$). Therefore, $GF[p^n]$ is the alias to $\mathbb{Z}/p^n$.

References

[Cai03] Jin-Yi Cai. *Lectures in Computational Complexity.* `https://pages.cs.wisc.edu/~jyc/710-draft-book.pdf`. [Online; accessed 03-May-2024]. 2003.

[Gal21] Joseph Gallian. *Contemporary abstract algebra.* Chapman and Hall/CRC, 2021.

[NW94] Noam Nisan and Avi Wigderson. “Hardness vs randomness”. In: *Journal of computer and System Sciences* 49.2 (1994), pp. 149–167.