Note 6: Modular Arithmetic 0183 dot

Thrm 6.1. For all $n \geq 1$ and a,b, c,d ∈ Z, the following are true: 1. If $a \equiv_n b$ and $c \equiv_n d$, then $a+c \equiv_n b+d$ 2. If $a \equiv_n b$ and $c \equiv_n d$, then $a \cdot c \equiv_n b \cdot d$ *(Direct proof)*

Multiplicative Inverse: $a \cdot b \equiv 1 \pmod n$

Thrm 6.2. Let n, x be positive integers. Then x has a multiplicative inverse modulo n if and only if gcd(n, x) = 1. Moreover, if it exists, then the multiplicative inverse is unique. *(Two-step proof by contradiction)*

Euclid's Algorithm: Assumes $x \geq y \geq 0$ and $x > 0$. Outputs gcd(x, y).

> **gcd(x, y):** if y = 0, then return x; else, return gcd(y, x (mod y))

Thrm 6.3. Let $x \geq y$ and let q,r be natural numbers such $x = yq+r$ and $r < y$. Then gcd(x, y) = gcd(y,r). *(Direct proof)*

Extended Euclid GCD: Assumes $x \geq y \geq 0$ and $x > 0$. Outputs (d,a,b) where d = gcd(x, y) and a,b ∈ Z with $d = ax+by$.

> **extended-gcd(x, y):** if y = 0, then return (x,1,0); else, let (d,a,b) := extended-gcd(y, x (mod y)); return (d, b, a− [x/y]b)

Thrm 6.4. If x and y satisfy the preconditions of extended-gcd, then the output(d,a,b) of extended-gcd(x, y) satisfy its postconditions. *(Strong induction)*

Note 7: Bijections and RSA 8707 ∃ 8704 ∀ 8715 ∋

Bijection: a function f : A → B is a bijection iff for all b ∈ B, ∃ a unique pre-image a ∈ A such that f(a) = b.

> onto or surjective: ∀ b ∈ B, ∃ s.t. f(a) = b.
>
> one-to-one or injective: no two inputs -> same output unless they are the same input

Lemma 7.1. A function f : A → A is a bijection iff there is an inverse function g : A → A such that g(f(x)) = x and f(g(y)) = y for all x, y ∈ A *(Direct proof)*

Thrm 7.1. [Fermat's Little Theorem] For any prime p and any a ∈ {1,2,..., p−1}, we have ap−1 ≡ 1 mod p. *(Two-step direct proof)*

RSA: two-key cipher using primes and bijections; $E(x) \equiv x^e \bmod N$; $D(y) \equiv y^d \bmod N$; where N=pq for large primes p&q, e relatively prime to (p-1)(q-1), E={0,…,N-1}. Inverse functions.

Thrm 7.2. For E and D as defined above, we have D(E(x)) = x mod N for all x ∈ {0,1,...,N −1}

Note 8: Polynomials (also 9)

Property 1: A non-zero polynomial of degree d has at most d roots. *(Two-step direct proof)*

Property 2: Given d +1 pairs (x1, y1),...,(xd+1, yd+1), with all the xi distinct, there is a unique polynomial p(x) of degree (at most) d such that p(xi) = yi for 1 ≤ i ≤ d +1.

Polynomial Division: p(x) = q(x) q'(x) +r(x)

GF(m) [Galois Field]: polynomials in mod m (prime m)

Lagrange Interpolation: $p(x) = \sum_{d+1} y_i \Delta_i(x)$; where $\Delta_i(x) = \Pi_{j!=i}(x-x_j)/ \Pi_{j!=i}(x_i-x_j)$.

Secret Sharing: create n-1 degree polynomial where any n people can decode secret via interpolation, GF(m) where m is a large prime, secret is P(0)

Erasure Correction: create n-1 degree polynomial, send n+k packets, work over GF(q) where q is sufficiently large (k is errors)

Corruption Correction: same as erasure, send 2k additional packets; need ≥ n+k packets in agreement (k is possible errors)

Berlekamp-Welch algorithm:
$Q(x) = a_{n+k-1}x^{n+k-1} +\cdots+a_1x+a_0$
$E(x) = x^k +b_{k-1}x^{k-1} +\cdots+b_1x+b_0$

n+2k linear equations in n+2k unknowns. Unknowns correspond to the coefficients of E(x) and Q(x). (where we define Q(x) = P(x)E(x)). Once Q(x) and E(x) are known, we can divide Q(x) by E(x) to obtain P(x).

$Q(x) = P(x)E(x) = r_xE(x)$, where $E(x) = (x - e_1)\ldots(x - e_k)$

Note 10: Infinity and Countability

Cardinality: size of a set; to prove same cardinality demonstrate a bijection between sets

Countable: bijection between S and N or a subset of N

Cantor-Bernstein Thrm: If there is a one-to-one function f : A → B, then the cardinality of A is less than or equal to that of B. Show cardinality of A and B are equal by showing |A| ≤ |B| and |B| ≤ |A| (there is a one-to-one function f : A → B and a one-to-one function g : B → A). The existence of these two one-to-one functions implies that there is a bijection h : A → B, thus showing that A and B have the same cardinality.

Cantor's Diagonalization Proof: Suppose towards a contradiction that there is a bijection f : N → R[0,1]. Enumerate list with real numbers $0.d_1d_2d_3\ldots$; diagonal is a real number D, make number s by adding 2 mod 10 to every each digit. Number is either n^{th} on the list (contradiction b/c of n^{th} digits of n^{th} number, D, and s) or not (contradictions f's bijectivity)

Note 11: Self-Reference and Uncomputability

Quine: a program that prints itself

Recursion Thrm: given any program P(x, y), can always convert it to another program Q(x) such that Q(x) = P(x,Q), i.e., Q behaves exactly as P would if its second input is the description of the program Q

Halting Problem: does the program go in an infinite loop? *(Proof involving self-reference and non-separation of programs and data. Proof by diagonalization)*

> **TestHalt(P,x):** if halts on P, "yes"; else, "no"
>
> Consider *TestHalt(P,P)*. Define *Turing(P)*.
>
> **Turing(P):** if TestHalt(P,P), then loop; else, halt
>
> What about *Turing(Turing)*? If halts, TestHalt() should have returned "no" in Turing(), but by definition should have returned "yes". Vice versa.
>
> All halting problems reduce to this. (e.g. show if we can solve Easy Halting Problem, we can solve Halting Problem; but we can't solve Halting Problem)

Godel's Incompleteness Theorem: Arithmetic cannot be both consistent and complete (i.e. axioms exist). If a system T contains statement of its own consistency, T is inconsistent.

(You're probably fucked for this section.)

Note 12: Counting

First Rule of Counting: With k choices, n_1 ways for first choice, n_2 for the second for each result of first choice, etc, to the k^{th} choice (n_k ways), total number of results is product of number of ways.

Second Rule of Counting: If order of choices doesn't matter, use first rule then divide by number of orderings. (e.g. $A_1NA_2GRA_3M$ = 7!/3!)

N choose K: (n k) = n!/[(n-k!)k!]

Balls and bins: How many ways to put k balls into n bins? *(e.g. 3 balls 5 bins; dist = distinguishable)*

> Balls dist, bins dist: n^k (e.g. 5^3)
>
> Balls indist, bins dist: (k+n-1 k) (e.g. (7 3))
>
> Balls indist, bins indist: brute force (e.g. 3)
>
> Balls dist, bins indist: brute force/Stirling #'s (e.g. 5)
>
> Bins need to be dist for ez stuff. Bit strings similar. Indist. balls = replacement, order does not matter.

Note 25: Probability (also 13-16)

Probability Space: a sample space Ω, together with a probability Pr[ω] for each sample point ω, such that

- 0 ≤ Pr[ω] ≤ 1 for all ω ∈ Ω.
- $\sum_{\omega\in\Omega}$ Pr[ω] = 1, i.e., the sum of the probabilities of all outcomes is 1.
- For any event A ⊆ Ω, we define the probability of A to be Pr[A] = $\sum_{\omega\in A}$ Pr[ω].

Sample point: outcome of random experiment

Sample space: Ω, set of all possible outcomes

Definition 14.1 (Conditional Probability). For events A,B in the same probability space, such that Pr[B] > 0, the conditional probability of *A given B* is
Pr[A|B] = Pr[A∩B]/Pr[B].

Bayes' Rule: (useful when given Pr[B|A])
Pr [A|B] = Pr[A∩B]/Pr[B] = Pr[B|A] Pr[A]/Pr[B].
(Pr[B|A] = Pr[B∩A]/Pr[A])

Total Proabilty Rule: (dividing Pr[B] into cases)
Pr[B] = Pr[A∩B] +Pr[A∩~B] = Pr[B|A]Pr[A] +Pr[B|~A](Pr[~A]).
Pr[A|B] = Pr[B|A]Pr[A] / Pr[B|A]Pr[A] +Pr[B|~A](Pr[~A]).

Definition 14.2 (Independence). Two events A,B in the same probability space are independent if Pr[A∩ B] = Pr[A]×Pr[B]

Definition 14.3 (Mutual independence). Events A1,...,An are mutually independent if for every subset I ⊆ {1,...,n}, Pr[$\cap_{i\in I} A_i$] = $\prod_{i\in I}$ Pr[A_i]. *(Basically everything is independent of everything else.)*
Pr[$A_{i1} \cap A_{i2} \cap \cdots \cap A_{in}$] = Pr[$A_{i1}$]× ··· ×Pr[$A_{in}$]

Definition 16.1 (Random Variable). A random variable X on a sample space Ω is a function X : Ω → R that assigns to each sample point ω ∈ Ω a real number X(ω).

> Random variables discrete; countably infinite.Not actually random and not actually a variable. What *is* random is which sample point of the experiment is realized and hence the value that the random variable maps the sample point to. *(Kinda like a histogram?)*

Definition 16.2 (Distribution). The distribution of a discrete random variable X is the collection of values {(a,Pr[X = a]) : a ∈ A }, where A is the set of all possible values taken by X.

Definition 16.3 (Expectation). The expectation of a discrete random variable X is defined as
E(X) = $\sum_{a\in A}$ a × Pr[X = a], where the sum is over all possible values taken by the r.v. *(Think back to Buhler)*