COURSE NOTES

CS 2050 - Discrete Mathematics

Jerry Chen

Georgia Institute of Technology

Contents

1 Logical Form

1.1 Propositions

Learning how to write logic in mathematical notation builds up to ultimately constructing proofs (e.g., showing that two things are logically equivalent).

Definition and Properties

A proposition is either true or false. There is no in-between; this is called the *Law of Excluded Middle*. Propositions are pretty much like declarative English phrases which are either objectively true or false.

Letters like p, q, and r are used to represent propositions in the form of propositional variables via statements: *Let p be that the sky is blue.*

Connectives

Like mathematical operators, there are a few ways to chain propositions during evaluations:

- Negation ($\neg$): negative for propositions; "not"
- Conjunction ($\wedge$): combines propositions; logical "and"
- Disjunction ($\vee$): inclusive logical "or"
- Exclusive Or ($\oplus$): exclusive logical "or" (one or the other, but not both)

Operator Precedence

When evaluating complex logical expressions, the operators are evaluated in a specific order, similar to the order of operations in arithmetic. The precedence of logical operators is as follows, from highest to lowest:

1. Negation ($\neg$)
2. Conjunction ($\wedge$)
3. Disjunction ($\vee$)
4. Implication ($\rightarrow$)
5. Biconditional ($\leftrightarrow$)

Parentheses can be used to override the default precedence.

Truth Tables

A truth table is an easy way to show the evaluations of propositions, where each column shows a different proposition and where there are 2^n rows (where n is the number of propositional variables) due to each variable having a True/False state.

Two propositions are equivalent if their truth tables are identical (in other words, they evaluate to the same no matter how).

p	q	$p \vee q$	$p \wedge q$	$\neg p$	$\neg\neg p$	$p \oplus q$	$p \implies q$	$q \implies p$	$(\neg p) \implies (\neg q)$	$(\neg q) \implies (\neg p)$
T	T	T	T	F	T	F	T	T	T	T
T	F	T	F	F	T	T	F	T	T	F
F	T	T	F	T	F	T	T	F	F	T
F	F	F	F	T	F	F	T	T	T	T

Notice the $p \implies q$ and its contrapositive have identical columns in the truth table. That means that these two logical statements are equivalent. This is true, even in english:

- If it is raining, then the bus is late.
- If the bus is not late, then it is not raining.

From here on, if we say two propositions are equivalent, we mean their truth tables have identical columns. We will detail equivalence of propositions later.

(credit to Professor Abrahim Ladha)

Implications (Logical Consequence / Conditional Statements)

If one proposition leads to another, and there is some causal relationship between propositions p and q, then it is represented as "if p then q" $p \to q$.

For example, let p be that I ate pizza, and let q be that I am not hungry. The truth table here would look like this:

p	q	p -> q
T	T	T
T	F	F
F	T	T
F	F	T

- Importantly, if p was False, then there's no way to know that the statement was a lie (e.g., not true), so by the Law of Excluded Middle, the implication is true.

Also note that $p \to q$ is equivalent to $\neg q \to \neg p$. If p has occurred, q must have also occurred. Therefore if q has not occurred, p must not have happened.

Additional Implications:

- Converse: $q \to p$
- Inverse: $\neg p \to \neg q$
- Contrapositive: $\neg q \to \neg p$ (combination of converse + inverse)

A biconditional is equivalent to "if and only if": $p \iff q$. That is, they must be either both true or both false. Logically, it's the same thing as saying $(p \to q) \wedge (q \to p)$.

- If p occurred, q must have also occurred.
- But if p did not occur, then q must have also not occurred, because q leads to p.

1.2 Propositional Logic

Laws of Thought

All Laws

Here is a small cheatsheet of exactly and only the laws you can use.

• $p \wedge T \equiv p$	**Identity**
• $p \vee F \equiv p$	
• $p \vee T \equiv T$	**Domination**
• $p \wedge F \equiv F$	
• $p \wedge p \equiv p$	**Idempotent**
• $p \vee p \equiv p$	
• $\neg\neg p \equiv p$	**Double Negation**
• $p \wedge q \equiv q \wedge p$	**Communitivity**
• $p \vee q \equiv q \vee p$	
• $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	**Associativity**
• $(p \vee q) \vee r \equiv p \vee (q \vee r)$	
• $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	**Distributive Laws**
• $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	
• $\neg(p \wedge q) \equiv (\neg p \vee \neg q)$	**DeMorgan's Laws**
• $\neg(p \vee q) \equiv (\neg p \wedge \neg q)$	
• $p \vee (p \wedge q) \equiv p$	**Absorption**
• $p \wedge (p \vee q) \equiv p$	
• $p \vee \neg p \equiv T$	**Negation**
• $p \wedge \neg p \equiv F$	
	Implication
• $p \implies q \equiv \neg q \implies \neg p$	contrapositive
• $p \implies q \equiv \neg p \vee q$	conditional disjunction equivalence
• $p \iff q \equiv (p \implies q) \wedge (q \implies p)$	**Biconditional**

(credit to Professor Abrahim Ladha)

Other Laws

These are some laws which may be demonstrated from those previous. You may not apply these, but you should know them.

- $p \vee q \equiv \neg p \implies q$
- $p \wedge q \equiv \neg(p \implies \neg q)$
- $\neg(p \implies q) \equiv p \wedge \neg q$
- $(p \implies q) \wedge (p \implies r) \equiv p \implies (q \wedge r)$
- $(p \implies r) \wedge (q \implies r) \equiv (p \vee q) \implies r$
- $(p \implies q) \vee (p \implies r) \equiv p \implies (q \vee r)$
- $(p \implies q) \vee (p \implies r) \equiv (p \wedge q) \implies r$
- $p \iff q \equiv (p \implies q) \wedge (q \implies p)$
- $p \iff q \equiv \neg p \iff \neg q$
- $p \iff q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$
- $\neg(p \iff q) \equiv (p \iff \neg q)$

(credit to Professor Abrahim Ladha)

1.3 Predicates and Quantifiers

Sometimes, we can't determine the truth value of a statement because it depends on a variable.

Predicates

A predicate is a statement that contains a variable, and it becomes a proposition when the variable is assigned a specific value. It's also called a propositional function. For example, let $P(n)$ be the statement $n > 7$.

- $P(10)$ is true.
- $P(2)$ is false.

The set of possible values for the variable (like n) is called the universe of discourse (otherwise known as the domain).

Quantifiers

Quantifiers tell us how much of the universe of discourse a predicate applies to.

- Existential Quantifier ($\exists$): This means "there exists", "for some", or "at least one".
  - For example, $\exists n(P(n))$ means there is at least one value of n in the universe of discourse for which $P(n)$ is true.
- Universal Quantifier ($\forall$): This means "for all" or "for every".
  - For example, $\forall n(P(n))$ means that for every value of n in the universe of discourse, $P(n)$ is true.
- Uniqueness Quantifier ($\exists!$): This means "there exists a unique" or "exactly one".

When a variable is bound by a quantifier, it's no longer a free variable. A statement with no free variables is a proposition.

Multiple Quantifiers

The order of quantifiers matters. For example:

- $\forall x \exists y(Q(x, y))$: "For every x, there is a y such that Q(x,y) is true."
- $\exists y \forall x(Q(x, y))$: "There is a y such that for every x, Q(x,y) is true." These two statements are not the same!

Negating Quantifiers

We can negate quantifiers using these rules (similar to De Morgan's laws):

- $\neg(\forall x P(x)) \equiv \exists x(\neg P(x))$
- $\neg(\exists x P(x)) \equiv \forall x(\neg P(x))$

2 Inference

2.1 Arguments

Definition and Properties

An argument is a sequence of propositions (called premises) that lead to a conclusion. An argument is valid if the premises logically lead to the conclusion. In other words, if the premises are all true, the conclusion must also be true.

This can be written as a large implication, where the premises are all in a conjunction: $(p_1 \land p_2 \land ... \land p_k) \to q$

For an argument to be correct, its premises must be true. If the set of premises are not true, then the implication is trivially true.

Rules of Inference

Name	Rule of Inference	Tautology
Modus Ponens	p $p \to q$ $\therefore q$	$(p \land (p \to q)) \to q$
Modus Tollens	$\neg q$ $p \to q$ $\therefore \neg p$	$(\neg q \land (p \to q)) \to \neg p$
Hypothetical Syllogism	$p \to q$ $q \to r$ $\therefore p \to r$	$((p \to q) \land (q \to r)) \to (p \to r)$
Disjunctive Syllogism	$p \lor q$ $\neg p$ $\therefore q$	$((p \lor q) \land \neg p) \to q$
Addition	p $\therefore p \lor q$	$p \to (p \lor q)$
Simplification	$p \land q$ $\therefore p$	$(p \land q) \to p$
Conjunction	p q $\therefore p \land q$	$((p) \land (q)) \to (p \land q)$
Resolution	$p \lor q$ $\neg p \lor r$ $\therefore q \lor r$	$((p \lor q) \land (\neg p \lor r)) \to (q \lor r)$

Fallacies

Fallacies are common errors in reasoning that result in an invalid argument.

- Fallacy of Affirming the Conclusion: $((p \to q) \land q) \to p$ is *not* a tautology. Just because the conclusion is true doesn't mean the premise was the cause.
- Fallacy of Denying the Hypothesis: $((p \to q) \land \neg p) \to \neg q$ is *not* a tautology. Just because the hypothesis is false doesn't mean the conclusion is false.

Inference with Quantifiers

These rules let us use quantified statements in our arguments.

- Universal Instantiation: If $\forall x P(x)$ is true, then $P(c)$ is true for any specific element c from the universe of discourse.

- Universal Generalization: If $P(c)$ is true for an arbitrary element c, then $\forall x P(x)$ is true.
- Existential Instantiation: If $\exists x P(x)$ is true, we can say that $P(c)$ is true for *some* element c. We might not know which one.
- Existential Generalization: If $P(c)$ is true for some element c, then $\exists x P(x)$ is true.

De Morgan's Laws for Quantifiers

- $\neg \forall x P(x) \equiv \exists x \neg P(x)$
- $\neg \exists x P(x) \equiv \forall x \neg P(x)$

3 Proofs

Definition and Properties

The purpose of a proof is to establish the complete and convincing truth of a mathematical statement.

Conjecture A conjuncture is a statement which has not been proven yet.

Axioms Because a truth can only be derived from other truths, we need a base starting point. That's what an axiom is. An axiom is something that can be assumed to be true without proof, acting as a fundamental building block for other proofs.

Theorems A theorem is a statement which is not an axiom but has already been proven true.

Lemmas A lemma is kind of a smaller helper theorem that is used to prove the main theorem that we're trying to show.

Corollaries Finally, a corollary is a theorem that is kind of a bonus after a more general theorem is shown. It's kind of a follow-up conclusion that can be directly drawn from that result.

Universes of Discourse

Natural numbers are the set of all numbers which are non-negative integers, sometimes including 0 based on the definition used.

Integers are the set of all real numbers which do not have a decimal place and also are not irrational. Basically natural numbers but also can be negative.

Rational numbers can be expressed as some fraction of two integers, a and b, but b cannot be zero.

Irrational numbers are all real numbers that are not rational (any real number is either rational or irrational).

Complex numbers are any a + bi where a and b are any reals and $i^2 = -1$.

Proving Something is False To show that a universally quantified statement is false, you just need to find a single counterexample for where it is false.

Direct Proof

A direct proof starts with the assumption that the hypothesis P is true, and then uses a series of steps and definitions—as well as previously established axioms and theorems—to arrive at the conclusion Q.

Proof by Contraposition

Because the contrapositive of an implication is logically equivalent to that implication, it can also be used to prove a statement. It is also often easier if the direct proof doesn't work out, because the contrapositive may create a more concrete starting point and provide more useful information for the subsequent steps of the proof.

Proof by Cases

Proof by cases (also called proof by exhaustion) is usable when you can show that a finite number of cases can cover all possibilities in the universe of discourse, and each case is disjoint. Then you can prove that the statement is true for each of these individual cases—which would show that the overall conditional is also true.

Proof by Contradiction

Begin with "Assume to the contrary that..." and end with "contradiction".

To demonstrate a proposition is true, you can also show that the negation of the proposition leads to a logical inconsistency. This works due to the law of excluded middle, where if false is false, then it must be true.

This inconsistency can either be shown through a direct contradiction—something where one thing would lead to something like 0=1, which is clearly false—or it can contradict an axiom or a previously established theorem, or it can contradict an initial assumption that you're starting the proof with.

3.1 The Principle of Mathematical Induction

Definition of Induction

Let $\Phi(\cdot)$ be any predicate over the natural numbers. The Principle of Mathematical Induction states

$$(\Phi(0) \land \forall k \in \mathbb{N}(\Phi(k) \implies \Phi(k+1))) \implies \forall n \Phi(n)$$

Every induction proof has two parts.

- The base case must be proven true. Induction requires that there be a "discernible start" for the universe of discourse. For a statement over all naturals, this could be 0 or 1.
- In the induction step, assume the induction hypothesis. Let k be a fixed number, and assume $\Phi(k)$ is true. This is then used to prove that $\Phi(k+1)$ must be true.

Use Cases Induction is typically used to prove a statement for a larger generalization, commonly over the natural numbers or integers which are greater than or equal to some starting value.

Note that induction requires the use of different proof techniques to show both the base case and the induction step implication. By itself, it does not prove anything.

General Template We proceed by induction. Our base case is... Now assume the induction hypothesis that `<Induction Hypothesis in terms of k (where k has same range as n)`. We prove that `<Predicate for k + 1>`... (use the induction hypothesis).

Since we have shown that `<Predicate for k + 1>`, we have proven for `<constraint/range of n>` that `<conclusion>`.

The Well-Ordering Principle (Limits of Induction)

The Well-Ordering principle is an axiom of set theory, and it states that every non-empty subset $S \subseteq N$ has a least element.

In the case of induction, the "least" element is the base case; the starting value for that set. This means that induction would not work for the universe of discourse of all integers, because there is no smallest number, unless you limited it to something like all integers ≥ -5.

Furthermore, while induction is able to work with infinite sets (e.g., all natural numbers), the elements within that set must have a discrete, well-defined successor. This means that it would not work if the universe of discourse was all reals ≥ -5, because between any two reals there is always another real number.

Use Cases The well-ordering principle can be used to prove all sorts of properties pertaining to inductive definitions. A common way to proceed is:

1. Proceed with a proof by contradiction. By starting with "Assume to the contrary...", this means that there must be at least 1 element for which the property `P` does not hold.
2. Define the set of counterexamples. The assumption in step 1 means that the set of counterexamples is a non-empty set.
3. Use the Well-Ordering Principle to state that because it is a non-empty set of positive integers, it must contain a smallest element.
4. Based on this definition of the smallest counterexample (let's call it `m`), we know that for all `k < m`, `k` cannot be a counterexample (for all `k < m`, `P(k)` must be true).

5. Use the inductive definition to create a contradiction. Note that the base case will typically be true and will not be `m`. Then by the inductive definition, show that `k = m - 1` or some other combination of prior conditions leads to `P(m)` being true.
6. But this leads to the contradiction. `P(m)` cannot be both true and false. Thus, the assumption that the set of contradictions is nonempty must be false. Thus, we have proven the property true for all elements.

3.2 The Principle of Strong Induction

Definition and Properties

The principle of strong induction is a variation of mathematical induction that is useful for proofs where the truth of a statement depends not only on the immediately preceding case, but potentially multiple cases before that as well (e.g., how the definition of a Fibonacci number depends on two preceding terms, not just one).

$$(\Phi(0) \land \forall k \in \mathbb{N}((\Phi(0) \land ... \land \Phi(k)) \implies \Phi(k+1))) \implies \forall n \in \mathbb{N}\, \Phi(n)$$

Note: String induction is logically equivalent to normal induction

A proof by strong induction follows a similar structure but with a key difference in the inductive hypothesis:

- Base Case: As with regular induction, you must prove that the statement P(n) is true for the starting integer(s). For strong induction, it's sometimes necessary to prove multiple base cases depending on the recurrence of the statement.
- Inductive Hypothesis: You assume that the statement P(i) is true for *all* integers i such that the starting integer $\leq$i$\leq$k. This provides a more comprehensive set of assumptions to work with in the inductive step.
- Inductive Step: Using this stronger hypothesis, you must then prove that the statement P(k+1) is true. You can use the fact that the statement holds for any previous integer, not just the single one right before it.

General Template We proceed by strong induction on `<some countable thing>`.

4 Notes on 1st Exam

4.1 Types of Problems

1. Given predicates or logical expressions evaluate quantified logical expressions (determine whether the statement is true).
2. The converse, inverse, and contrapositive of logical expressions in English or logical expressions.
3. Translate English into Quantified Logical Expressions. Translate Logical Expressions into English.
4. Give a counterexample to a statement, or affirm the statement's truth.
5. Show a conclusion given a set of hypotheses. That is, using the laws of thought show that either a set of premises can lead to a specific conclusion, or that one logical expression is equivalent to another.
6. Including direct, contraposition, contradiction, exhaustive/cases, and trivial/vacuous.
7. "Knights & Knaves" type word-logic problems.

4.2 Notes on Writing Solutions

For #5, number the steps moving down, and also choose one (more complex) side and stick with that expression (on the left).

- Write the law of thought on the right side in parentheses, and reference the steps that the laws of inference apply to if applicable.

Example:

Part 3

$$((p \wedge \neg q) \rightarrow r) \wedge ((\neg p \vee q) \rightarrow r) \equiv ((\neg p \vee q \vee r)) \wedge (((p \wedge \neg q) \vee r))$$

Proof:

1. $((p \wedge \neg q) \rightarrow r) \wedge ((\neg p \vee q) \rightarrow r)$ (Start with the left-hand side)
2. $((p \wedge \neg q) \rightarrow r) \wedge ((\neg(\neg p \vee q) \vee r))$ (Conditional Disjunction Equivalence)
3. $((\neg(p \wedge \neg q) \vee r)) \wedge ((\neg(\neg p \vee q) \vee r))$ (Conditional Disjunction Equivalence)
4. $((\neg p \vee \neg\neg q \vee r)) \wedge ((\neg(\neg p \vee q) \vee r))$ (De Morgan's Law)
5. $((\neg p \vee \neg\neg q \vee r)) \wedge (((\neg\neg p \wedge \neg q) \vee r))$ (De Morgan's Law)
6. $((\neg p \vee q \vee r)) \wedge (((\neg\neg p \wedge \neg q) \vee r))$ (Double Negation)
7. $((\neg p \vee q \vee r)) \wedge (((p \wedge \neg q) \vee r))$ (Double Negation)

4.3 Mistakes to Avoid

Translating logical statements into English statements

- Be sure to make it sound like a normal English sentence and not a direct “mathematical symbol –> English” translation.

Using Laws of Thought/Inference

- Do not combine associativity and commutativity into a single line.
- Do not apply the same law of thought/inference multiple times within a single line.
- Do not write the other side with the “≡” symbol; just turn one side into the other through a series of steps.

Truth Tables

- Write out the negation of statements as a column too.
- Knights/knaves problems are iff relationships; the intersection of all iff statements is the solution / set of possibilities for character identities.

Proofs

- Start with “Proof” and end with halmos symbol.
- Always restate the universe of discourse for propositional variables at the beginning (if provided) and state the proven statement at the end.
- If contraposition, start with “We prove the contrapositive: …”
- If contradiction, start with “Assume to the contrary that…”
-

5 Set Theory

5.1 Basic Definitions

Note that there is a great difference between $\in$ and $\subseteq$. $x \in A$ is used to denote that a single element x is within the set A. $X \subseteq A$ is used to denote that *every single element* in X is also in A.

5.2 Cardinalities

Size of $A \times B$ is equal to $|A| \times |B|$. Size of $\mathcal{P}(A) = 2^{|A|}$.

"or" typically indicates proof by cases

To prove that $LHS \subseteq RHS$, always start with LHS and let $x \in LHS$. Work more with the definition and show that for all $x \in LHS$, $x \in RHS$.

To prove that $LHS = RHS$, start with "we proceed by double set containment". Then, show both sides of the relationship, such that $LHS \subseteq RHS$ and $RHS \subseteq LHS$.

TABLE 1 Set Identities.

Identity	*Name*
$A \cap U = A$ $A \cup \emptyset = A$	Identity laws
$A \cup U = U$ $A \cap \emptyset = \emptyset$	Domination laws
$A \cup A = A$ $A \cap A = A$	Idempotent laws
$\overline{(\overline{A})} = A$	Complementation law
$A \cup B = B \cup A$ $A \cap B = B \cap A$	Commutative laws
$A \cup (B \cup C) = (A \cup B) \cup C$ $A \cap (B \cap C) = (A \cap B) \cap C$	Associative laws
$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$	Distributive laws
$\overline{A \cap B} = \overline{A} \cup \overline{B}$ $\overline{A \cup B} = \overline{A} \cap \overline{B}$	De Morgan's laws
$A \cup (A \cap B) = A$ $A \cap (A \cup B) = A$	Absorption laws
$A \cup \overline{A} = U$ $A \cap \overline{A} = \emptyset$	Complement laws

6 Functions and Complexity

6.1 Properties of Functions

Domain, Codomain, and Range/Image

The domain of a function is the range of all actual possible inputs.

- For $f(x) = x^2$, the domain is $\mathbb{R}$.

The codomain of a function is a broader set of all potential values that a function can possibly produce.

- For $f(x) = x^2$, the codomain is $\mathbb{R}$.

The range is always a subset of the codomain which is the set of values actually able to be produced by any input.

- For $f(x) = x^2$, the range is from $(0, \infty]$.

Definition of a Function

A relationship is a function if a single element in the domain X is only mapped to a single element in codomain Y (that is, one input cannot have two different outputs).

- Totality: For every $a \in A$, there is some $b \in B$ with $(a, b) \in f$.
- Uniqueness: For every $a \in A$, there is exactly one $b \in B$ with $(a, b) \in f$.

Injective: One to One

A function is injective if every distinct element in domain X is mapped to a distinct element in codomain Y. That is, no two different elements will map to the same element in the codomain.

- Mathematical definition: $\forall x, y \in X, (f(x) = f(y) \implies x = y)$
  - Alternatively, understand the contrapositive that $x_1 \neq x_2 \rightarrow f(x_1) \neq f(x_2)$
- How to prove –> assume different inputs for f (e.g., x_1, x_2) and show that after algebraic manipulation $x_1 = x_2$ for any general x_1, x_2.

Surjective: Onto

- A function is surjective if every single element in the codomain Y is the output of one or more elements in domain X. That is, all outputs are covered by inputs.
- Mathematical definition: $\forall y \in Y(\exists x \in X(f(x) = y))$
- How to prove –> Provide a constructive proof (a logical algorithm) to show that for any arbitrary element in the codomain Y, there exists an input within the proper domain (important that it's within the domain!).
  - This is done by solving for the inverse and noting any holes in the domain of the inverse

Bijective

Both injective and surjective; the function has an inverse which may map all values in the codomain (which is domain of inverse) back to the original domain (codomain of the inverse).

Monotonically Increasing/Decreasing

The definition of a monotonically increasing function is that $f(x)$ on set A is monotonically increasing if and only if $\forall x_1, x_2 \in A[x_1 < x_2 \implies f(x_1) \leq f(x_2)]$.

- It is *strictly* monotonically increasing if $\forall x_1, x_2 \in A[x_1 < x_2 \implies f(x_1) < f(x_2)]$.

The definition of a monotonically decreasing function is a function $f(x)$ on set A such that $\forall x_1, x_2 \in A[x_1 < x_2 \implies f(x_1) \geq f(x_2)]$.

- It is *strictly* monotonically decreasing if $\forall x_1, x_2 \in A[x_1 < x_2 \implies f(x_1) > f(x_2)]$.

Images of Functions

Let f be a function from set A (the domain) to set B (the codomain), which we write as $f : A \to B$.

The image of the function, denoted $f(A)$, is the set of all elements in the codomain B which are actually mapped to by at least one element A:

$$f(A) = \{y \in B | \exists x \in A \text{ such that } f(x) = y\}$$

6.2 The Growth of Functions

Big-O Notation

$f(n) = O(g(n))$ means that $\exists c \in \mathbb{R}^+$ such that $0 \leq f(n) \leq cg(n)$ for $n \geq k$ where k is a non-negative constant.

Proving that $f(n)$ is $O(g(n))$

- Use a witness proof. That is, proceed to find reasonable witnesses such that $f(n) \leq cg(n), \forall n \geq k$ for witnesses $c, k \in \mathbb{R}$.
  - If $f(n)$ is a polynomial, sum the coefficients and use that as c, so naturally $k = 1$.
- After finding witnesses, the proof is concluded, since their existence is shown.

Proving that $f(n)$ is not $O(g(n))$

- Use a proof by contradiction.
- After setting up that $f(n) \leq cg(n), \forall n \geq k$ for witnesses $c, k \in \mathbb{R}$, simplify so that c is isolated on the RHS and some expression is on the left hand side.
- Find some convenient value $n = y(a)$ (e.g., $n = 2^{2^a}$) which will allow for simpler operations with log/square root, since that is what often makes algorithm asymptotic analysis challenging.
  - *Not allowed*: Take the limit of the LHS as $n \to \infty$ to show that it grows without bounds, and therefore there does not exist a constant c such that LHS $\leq c$.

Llittle-O Notation

Big-Omega Notation

$f(n) = \Omega(g(n))$ means that $\exists c \in \mathbb{R}^+$ such that $0 \leq cg(n) \leq f(n)$ for $n \geq k$ where k is non-negative constant.

Proving that $f(n)$ is $\Omega(g(n))$

- Use a constructive witness proof. Pick a positive number c in order to state the inequality $f(n) \geq cg(n)$.
- Proceed with algebraic simplifications until you can solve for values of n which make the inequality true. Confirming this threshold value is positive is sufficient.
- Conclude by stating the two found witnesses, completing the proof using the definition of Big-Omega notation.

Little-Omega Notation

$f(n) = \omega(g(n))$ means that for all $c \in \mathbb{R}^+$, there exists $k > 0$ such that: $0 \leq cg(n) < f(n)$ for $n \geq k$.

Proving that $f(n)$ is $\omega(g(n))$ Note that you can no longer pick a specific c. Instead, it must be an arbitrary positive constant value.

For the scratch work, just assume that $f(n) > cg(n)$ (which would risk begging the question on an actual proof but is necessary/convenient for scratch). Then proceed to find an expression for n in terms of c. Then whatever c can be, there will always be a value n fulfilling the inequality.

For the formal proof:

1. Consider $k =$ some expression in terms of c. (This is the result of the scratch work, which forms the starting point).
2. Now, let n be any integer such that $n \geq k$. By our definition of k, $n \geq$ some expression in terms of c.
3. Reconstruct the left side from n to $f(n)$ and show that it is still greater than $cg(n)$.
4. This completes the proof.

Big-Theta Notation

$f(n) = \Theta(g(n))$ means that $\exists c_1, c_2 \in \mathbb{R}^+$ such that $0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$ for $n \geq k$ where k is a non-negative constant.

7 Relations

Definition and Properties

A relation R is any subset of a cartesian product of sets $R \subseteq A \times B$. When $(a, b) \in R$, we mean "a relates to b" and write aRb.

- In a sense, it is like a proposition with a specific truth value. It is true if the pair (a, b) satisfies the condition of the relation, and false otherwise.
- $aRb \iff (a, b) \in R$

"R is a relation over a set A" means $R \subseteq A \times A$. R describes a relationship between the elements within that single set A.

- For example, "$\leq$ over $\mathbb{R}$" denotes the set $\{(x, y) \in \mathbb{R} \times \mathbb{R} | x \leq y\}$.

Definition of a Function

A relation becomes a function whenever every input has exactly one output.

This is denoted as $f : A \to B$, where for each element in the domain $x \in A$, there is exactly one element $y \in B$ for which (x, y) is in the relation. That is, $\forall x \in A, \exists! y \in B$ such that $(x, y) \in f$.

7.1 Equivalence Relations

Definition and Properties

An equivalence relation is a generalized form of "equals". It's a way to partition a set into disjoint and distinguishable subsets called "equivalence classes". Everything within one class is basically equivalent according to the relation.

- For example, congruence in modular arithmetic is a type of equivalence relation.

Three properties need to be true for a relation to be an equivalence relation: reflexive, symmetric, and transitive. To prove an equivalence relation, prove all three of the following.

Reflexive Property

Reflexive means that if it's the same element it must belong to the same group. Every element a is related to itself.

For a relation R over A, R is reflexive if $\forall a \in A[aRa]$.

To prove the reflexive property, we must show for any arbitrary $a \in A$, aRa is true. Use the definition provided for the relation and show that after plugging in a for both values in the relation that the relationship still holds.

Symmetric Property

Symmetric means that if a is related to b, then b is related to a. We cannot yet say that a belongs in the same group as b, but that within a group, all members are treated equally, since any relation from a to b is also shared from b to a. There is no direction to the connection, and no ordering within the group.

For a relation R over A, R is symmetric if $\forall a, b \in A[aRb \iff bRa]$.

To prove the symmetric property, we assume aRb and we proceed to show this implies bRa.

Transitive Property

Transitive means that if a is related to b and b is related to c, then a too is related to c. Transitivity is the property that solidifies the "grouping" aspect of an equivalence relation.

For a relation R over A, R is transitive if $\forall a, b, c \in A[aRb \land bRc \implies aRc]$.

To prove the transitive property, we assume both aRb and bRc and through algebraic manipulation find some way to eliminate the middle b. Link a to c and show aRc.

Equivalence Classes

Let R be an equivalence relation on a set A. The set of all elements related to that element $a \in A$ is the equivalence class of a. The equvalence class of a with respect to R is $[a]_R$.

- That is, if R is an equivalence relation over set A, the equivalence class of $a \in A$ is: $[a]_R = \{s|(a, s) \in R\} = \{s|(s, a) \in R\}$.

An equivalence relation on set A partitions it into subsets, called equivalence classes (or partitions), which are both disjoint and exhaustive.

8 Number and Group Theory

8.1 Modular Arithmetic

Definition and Properties of "Divides"

For any two numbers a, b, we say that a divides b (denoted as $a \mid b$) if there exists a number c such that $ac = b$.

- This is equivalent to thinking that a divides into b evenly, such that b/a is some whole number.
- Trivially true cases: $a|b$ always when $a = b$, $b = 0$ (given $a \neq 0$), or $a = 1$.

Properties/Theorems:

- Additive: if $a|b$ and $a|c$, then $a|(b + c)$.
- Multiplicative: if $a|b$, then $a|bc$ for all c.
- Transitive: if $a|b$ and $b|c$, then $a|c$.

Additional Note: The first two combined create the divisibility of linear combinations. If $a|b$ and $a|c$, then for any integers x, y, $a|(bx + cy)$.

The Division Algorithm

For any number n, there is a d such that there exists unique numbers q, r such that $n = dq + r$.

- The d is the divisor while the r is an integer remainder $0 \leq r < d$.
- The q is the quotient, representing the number of times d goes evenly into n.

Definition of Modular Arithmetic

Modular arithmetic is a system for integers that wrap around after reaching a certain value, called the modulus.

Two integers a and b are congruent *modulo* n if their difference is an even multiple of n:

$$a \equiv b \pmod{n} \iff n|(a - b)$$

- An alternative way of thinking about it is that both have the same remainder when dividing by n.
- $a \equiv b \pmod{n}$ defines an equivalence relation on the set of integers while a statement like $a \bmod n = b$ is actually representing a function on a.

The set $\mathbb{Z}_n = \{0, \ldots n-1\}$ represents the set of all possible remainders when dividing by n, and in modular arithmetic, all integers in existence must be congruent to exactly one of the numbers in this set.

- Each element is representative of an equivalence class since $\pmod{n}$ partitions the set of all integers into these n classes.

Equivalence Classes

Sometimes, the notation $[i] \in \mathbb{Z}_n$ will be used. This means the equivalence class of all integers such that any integer within the equivalence class will have i remainder divided by n (where n is the number of partitions / equivalence classes formed).

In set builder notation, this looks like: $[i] = \{x \in \mathbb{Z} | x \equiv i \pmod{\text{n}}\}$

Properties of Modular Arithmetic

Valid Properties:

- All properties of equivalence relations:
  - Reflexivity: $a \equiv a \pmod{n}$

    - Symmetry: $a \equiv b \pmod{n} \implies b \equiv a \pmod{n}$
    - Transitivity: $a \equiv b \pmod{n} \land b \equiv c \pmod{n} \implies a \equiv c \pmod{n}$
- Addition/Subtraction: $a \equiv b \pmod{n} \land c \equiv d \pmod{n} \implies a + c \equiv b + d \pmod{n}$
- Integer Multiplication: $a \equiv b \pmod{n} \land c \equiv d \pmod{n} \implies ac \equiv bd \pmod{n}$
- Exponentiation: $a \equiv b \pmod{n} \implies a^k \equiv b^k \pmod{n}$

Finally, if $a \equiv b \pmod{n}$, then we can replace it in any expression without division and it remains true:

1. For example, we can attempt to find the last digit of 7^{2025}.
2. That is equivalent to finding x within the expression $7^{2025} \equiv x \pmod{10}$.
    - $7 \equiv 7 \pmod{10}$
    - $7^2 \equiv 49 \equiv 9 \pmod{10}$
    - $7^3 \equiv 7^2 \times 7 \equiv 9 \times 7 \equiv 63 \equiv 3 \pmod{10}$
        - Replace 49 with 9.
    - $7^4 \equiv 3 \times 7 \equiv 21 \equiv 1 \pmod{10}$
3. Now we proceed, knowing that $7^{2025} \equiv 7^{2024} \times 7 \equiv (7^4)^{506} \times 7 \equiv (1)^{506} \times 7 \pmod{10}$.
4. Then $7^{2025} \equiv 7 \pmod{10}$.

Proof by Cases

Proof by cases often works well when dealing with modular arithmetic and you're trying to prove (whether for a lemma or for an actual proof) that all numbers of a certain type (e.g., perfect square) have a specific remainder when divided by a smaller number.

An example of this is showing that the remainder of any perfect square divided by 3 can only be 0 or 1.

- The perfect square is k^2, with $k \in \mathbb{N}$.
- Split into cases, where $k \equiv 0 \pmod 3$, or $k \equiv 1 \pmod 3$, or $k \equiv 2 \pmod 3$.
- Show for each of these cases that a certain outcome is guaranteed.

Potentially-Helpful Theorems and Axioms

Perfect Square Test

The remainder of any perfect square divided by 4 will always be either 0 or 1.

- Two cases, where the perfect square is k^2, and k can either be even or odd.

Coprime and Modular Inverses

An integer a has an inverse modulo $n \iff gcd(a, n) = 1$.

8.2 Integer Representations and Algorithms

Basis Representation Theorem

Let b be an integer greater than 1. Then if n is a positive integer, it can be expressed uniquely in the form:

$$n = a_k b^k + a_{k-1} b^{k-1} + \cdots + a_1 b + a_0$$

- Where k is a non-negative integer, $a_0 \ldots a_k$ are non-negative integers less than b, and $a_k \neq 0$.

8.3 Primes, FTA, GCD, and LCM

Definitions and Properties

What is a Prime Number?

An integer p greater than 1 is called *prime* if the only positive factors of p are 1 and p. Any positive integer greater than 1 that is not prime is *composite*.

- Note that 1 is not prime, because it only has a single positive factor.
- An integer is composite $\iff \exists a \in \mathbb{Z}[(a \mid n) \wedge (1 < a < n)]$.

Additional Definition for Composite Integer If n is a composite integer, then n has a smallest prime divisor less than or equal to $\sqrt{n}$.

- Using its contrapositive, we can show that so long as n has no prime divisors less than for equal to $\sqrt{n}$, n is a prime integer.

Fundamental Theorem of Arithmetic

Every integer greater than 1 can be uniquely written as either a prime or the product of two or more primes, where the prime factors are written in order of non-decreasing size.

- Essentially, this theorem guarantees unique prime factorizations.

Greatest Common Divisor

The *Greatest Common Divisor* (GCD) of two integers a and b, denoted as $\gcd(a, b)$, is the largest positive integer d that divides both a and b (meaning $d|a$ and $d|b$). This definition requires that a and b are not both zero.

Intuitively, the GCD can be understood by considering the prime factorizations of a and b. The $\gcd(a, b)$ is the product of the common prime factors, where each common prime is raised to the *lowest* power it appears in either factorization. This is equivalent to finding the "intersection" of their prime factorizations.

Least Common Multiple

The *Least Common Multiple* (LCM) of two integers a and b, denoted as $\text{lcm}(a, b)$, is the smallest positive integer l that is a multiple of both a and b (meaning $a|l$ and $b|l$).

Coprime

Two integers a and b are coprime, or relatively prime, if their greatest common divisor is 1.

Significance of Coprime: Existence of Multiplicative Inverse Modulo n An integer a has a multiplicative inverse modulo n if and only if it is coprime with n.

$$a^{-1} \pmod{n} \iff gcd(a, n) = 1$$

1. This follows from considering an element x such that $ax \equiv 1 \pmod{n}$.
2. Then by the definition of congruence mod n, we have $ax - kn = 1$.
3. Using Bezout's Theorem, we know $as + bt = gcd(a, b)$ as long as a and b are integers not both zero. Then, plugging this form in, we see a maps to a and b maps to n, with $s = x$ and $t = -k$, so the only integer solution is when $gcd(a, n) = 1$.

This multiplicative inverse is unique modulo n.

- To find the actual inverse number, either modular arithmetic or Extended Euclidean Algorithm can be used. That is, the equation $ax = kn + 1$ can be molded to look like Bezout's Identity, and then Extended Euclidean Algorithm can be used to find the coefficients (x corresponds to the inverse).
- This is because modding both sides of the equation results in the kn dropping off and becoming $ax \equiv 1 \pmod{n}$.

The Euclidean Algorithm to Find GCD (Forward Pass)

Consider the division algorithm: for any number n, there exists a d such that there exists unique numbers q, r so: $n = dq + r$.

Now consider two numbers a and b, not both zero. Without loss of generality, let a be the larger of the two numbers. Then by the division algorithm, $a = bq + r$.

Then we must notice that for some GCD k, if $k|a$ and $k|b$, then $k|(a - qb)$ (shown through the definition of divides). Since $r = a - qb$, $k|r$.

This all goes to show that finding the GCD of (a, b) is the same as fnding the GCD of $(a \pmod b, b)$. Now this can be done repeatedly by interchanging the positions of each number in the pair such that the larger can always be modded by the smaller.

- This continues until you reach zero in one number (then the non-zero number in the pair is the GCD).

The GCD Used to Find LCM

- The LCM is directly related to the GCD. The formula to calculate the LCM is:

$$\text{lcm}(a, b) = \frac{|a \cdot b|}{\gcd(a, b)}$$

- In the special case where a and b are *coprime* (or relatively prime), their $\gcd(a, b) = 1$. This simplifies the formula, and their $\text{lcm}(a, b) = |a \cdot b|$.

The Extended Euclidean Algorithm (Backward/Upward Pass)

Using the Euclidean Algorithm, we obtain $\gcd(a, b)$. Then, we can move upward, taking the last equation which does not have 0 as the remainder.

Each time, substitute by writing the remainder in terms of a previous term. Continually substitute back, keeping the left side of the equation (the remainder) the same while replacing a factor in the right side with an equivalent expression in an above equation.

Example: Find integers with $187s + 102t = \gcd(187, 102)$, and determine the s which is closest to 0.

$187 = 102 \cdot 1 + 85$
$102 = 85 \cdot 1 + 17$
$85 = 17 \cdot 5 + 0 \Rightarrow \gcd(187, 102) = 17.$

Back-substitution $17 = 102 - 85$
$= 102 - (187 - 102) = 2 \cdot 102 - 187.$
Thus a particular solution is $s_0 = -1,\ t_0 = 2$ with $187(-1) + 102(2) = 17$.

A note on the solution The algorithm generates a unique pair that satisfies $|s_0| \leq |\frac{b}{\gcd(a,b)}|$ and $|t_0| \leq |\frac{a}{\gcd(a,b)}|$.

All solutions are

$$s = s_0 + \frac{b}{d}k = -1 + 6k, \qquad t = t_0 - \frac{a}{d}k = 2 - 11k,$$

where $d = \gcd(187, 102) = 17$ and $k \in \mathbb{Z}$.

Minimizing |s|

Choose k to minimize $|s| = |-1 + 6k|$. The nearest candidates are $k = 0$ ($s = -1$) and $k = 1$ ($s = 5$); $|-1| < |5|$, so the closest to 0 is $s = -1$.

Answer. $s = -1$ (with $t = 2$) satisfies $187s + 102t = 17$ and has minimum absolute value among all valid s.

Bezout's Identity

Bezout's Identity states that for any integers a and b, there exists $s, t \in \mathbb{Z}$ (where not both are zero) such that:

$$as + bt = gcd(a, b)$$

- $gcd(a, b)$ is always going to be the least positive linear combination of a, b.
  - All solutions will be some integer multiple of $gcd(a, b)$.
- There are infinitely many pairs of solutions s, t for each integer multiple of $gcd(a, b)$.

We can prove the existence of these integers x and y by considering the set of all possible positive linear combinations of a and b.

Let S be the set of all positive integers that can be written in the form $ax + by$ for some integers x and y. So, $S = \{ax + by \mid x, y \in \mathbb{Z} \text{ and } ax + by > 0\}$.

First, we must show that this set S is non-empty. Since a and b are not both zero, let's assume $a \neq 0$. If $a > 0$, we can choose $x = 1, y = 0$, so $a(1)+b(0) = a$ is in S. If $a < 0$, we can choose $x = -1, y = 0$, so $a(-1)+b(0) = -a = |a|$ is in S. If $a = 0$, then $b \neq 0$, and a similar argument shows that $|b|$ is in S. In all cases, S contains at least one positive integer.

Since S is a non-empty set of positive integers, the Well-Ordering Principle states that S must have a least element. Let's call this least element d.

Since d is an element of S, by the definition of S, there must exist some integers x_0 and y_0 such that $d = ax_0 + by_0$.

We now claim that this d is, in fact, the greatest common divisor of a and b. To prove $d = \gcd(a, b)$, we must show two things:

1. d is a common divisor (it divides both a and b).
2. d is the *greatest* common divisor (any other common divisor c must also divide d).

First, we prove that d is a common divisor. Let's show $d \mid a$. We can use the Division Algorithm to write $a = qd + r$, where q is the quotient and r is the remainder, with $0 \leq r < d$.

We want to show that r must be 0. We can express the remainder r as a linear combination of a and b:

$$r = a - qd$$

Substitute our expression for $d = ax_0 + by_0$:

$$r = a - q(ax_0 + by_0)$$

$$r = a - qax_0 - qby_0$$

$$r = a(1 - qx_0) + b(-qy_0)$$

This shows that r is also a linear combination of a and b. Let $x' = (1 - qx_0)$ and $y' = (-qy_0)$. Then $r = ax' + by'$.

Now, consider the properties of r. We know $0 \leq r < d$. If $r > 0$, then r would be a positive linear combination of a and b, which means r would be an element of our set S. However, we also know $r < d$. This would mean r is an element of S that is *smaller* than d. This is a contradiction, as d was defined to be the *least* element of S.

The only way to avoid this contradiction is if our assumption ($r > 0$) was wrong. Therefore, the only possibility is that $r = 0$. If $r = 0$, then $a = qd$, which means $d \mid a$.

By a perfectly symmetrical argument (using $b = q'd + r'$), we can also show that $d \mid b$. Thus, d is a common divisor of a and b.

Second, we prove that d is the *greatest* common divisor. Let c be any arbitrary common divisor of a and b. This means $c \mid a$ and $c \mid b$. By the definition of divisibility, we can write $a = mc$ and $b = nc$ for some integers m and n.

Now, let's look at our expression for d:

$$d = ax_0 + by_0$$

Substitute the expressions for a and b:

$$d = (mc)x_0 + (nc)y_0$$

Factor out c:

$$d = c(mx_0 + ny_0)$$

Since m, x_0, n, and y_0 are all integers, the term $(mx_0 + ny_0)$ is also an integer. This expression $d = c \cdot (\text{integer})$ shows, by definition, that $c \mid d$.

So, we have shown that any other common divisor c must divide d. This implies that $c \leq |d|$. Since d is positive, $c \leq d$. This confirms that d is greater than or equal to every other common divisor, and thus d is the *greatest* common divisor.

We have successfully shown that $d = \gcd(a, b)$. Since we defined d as an element of S, we know there exist integers x_0 and y_0 such that $ax_0 + by_0 = d$. This completes the proof.

8.4 Euclid's Lemma

Proof (By Bezout's)

Euclid's Lemma describes a core property of divisibility and prime numbers. It states that if an integer a divides the product of two integers b and c (i.e., $a|bc$), and a is coprime to one of them (e.g., $\gcd(a, b) = 1$), then a must divide the other integer c (i.e., $a|c$).

Proof Outline The proof of Euclid's Lemma is a direct application of Bézout's Identity.

- Given: $a|bc$ and $\gcd(a, b) = 1$.
- From $a|bc$, we know that $ak = bc$ for some integer k.
- From $\gcd(a, b) = 1$, Bézout's Identity guarantees that there exist integers s and t such that $as + bt = 1$.
- Multiply the entire identity by c:

$$c(as + bt) = c \cdot 1$$

$$cas + cbt = c$$

- Substitute $bc = ak$ into the second term:

$$cas + (ak)t = c$$

- Factor a out of the left-hand side:

$$a(cs + kt) = c$$

- Since a, c, s, k, and t are all integers, the term $(cs+kt)$ is also an integer. By the definition of divisibility, a divides c.

Proving FTA with Euclid's Lemma

The Fundamental Theorem of Arithmetic (FTA) states that every integer greater than 1 are either primes themselves or can be represented uniquely as a product of prime numbers.

Existence of Factorization

- Every integer $n > 1$ is either prime or composite.
- If n is composite, $n = ab$ with $1 < a < n$ and $1 < b < n$.
- By repeatedly factoring composite numbers, we eventually reach only primes. (This argument is formalized using strong induction). Thus, every $n > 1$ has at least one prime factorization.

Uniqueness via Euclid's Lemma

Suppose n has two factorizations into primes:

$$n = p_1 p_2 \cdots p_k = q_1 q_2 \cdots q_m$$

where p_i and q_j are all prime. We must show the set of primes $\{p_i\}$ is identical to the set $\{q_j\}$.

- Consider the prime p_1. Since p_1 divides n, it must also divide the product $q_1 q_2 \cdots q_m$.
- We now apply Euclid's Lemma: If a prime p divides a product ab, then p must divide a or p must divide b.
- By applying Euclid's Lemma repeatedly to the product $q_1 q_2 \cdots q_m$, we find that p_1 must divide at least one of the prime factors q_j for some j.
- Since q_j is prime, its only positive divisors are 1 and q_j. Because p_1 is prime, $p_1 > 1$, so it must be that $p_1 = q_j$.
- We can now cancel p_1 from the left side and q_j from the right side, leaving a smaller product:

$$p_2 \cdots p_k = q_1 \cdots q_{j-1} q_{j+1} \cdots q_m$$

- We repeat this argument (formally, this is a proof by induction) for p_2, p_3, and so on, matching each p_i with a unique q_j.
- This process must terminate, proving that $k = m$ and the two sets of primes are identical.

8.5 Definitions in Group Theory

What is a Group?

A group consists of a set and a binary operation. A binary operation acts on two inputs and produces a single output. A group is generally denoted as $(G, *)$.

In order for something to be considered a group, it must fulfill 4 properties:

1. Closure: If $a, b \in G$, then $a * b \in G$.
2. Associativity: For any arbitrary $a, c, b \in G$, $(a * b) * c = a * (b * c)$.
3. Identity Element: There exists a unique element e in the set, called the identity element, such that $a * e = e * a = a$.
   - Both directions must be shown through computation/known properties.
4. Inverse Element: For every $a \in G$, there exists an element $a^{-1} \in G$ such that $a * a^{-1} = a^{-1} * a = e$.
   - Both directions must be shown.

Socks and Shoes Property

When $a, b \in G$, $(ab)^{-1} = b^{-1}a^{-1}$.

Abelian

Let $(G, \cdot)$ be a group. G is an abelian group if for all elements $a, b \in G$:

$$a \cdot b = b \cdot a$$

In simple terms, it is commutative.

Homomorphism

Let $(G, \cdot)$ and $(H, *)$ be two groups. A function $\phi : G \to H$ is a homomorphism if for all elements $a, b \in G$:

$$\phi(a \cdot b) = \phi(a) * \phi(b)$$

Isomorphism

Bijective + homomorphism

Notation of $(\mathbb{Z}/p\mathbb{Z})^X$

This refers to the set of all integers modulo p which are coprime to p.

More formally, it is the group of units modulo p, and "units" in a ring are the elements with multiplicative inverses, so any such element a must be coprime to p in order to be a "unit".

8.6 Fermat's Little Theorem

Proof of Fermat's Little Theorem

Fermat's Little Theorem states that if p is a prime number, then for any integer a:

$$a^p \equiv a \pmod{p}$$

Equivalently, if p is a prime number and a is an integer not divisible by p (i.e., $\gcd(a, p) = 1$), then:

$$a^{p-1} \equiv 1 \pmod{p}$$

We will prove the second form ($a^{p-1} \equiv 1 \pmod{p}$) first. The proof relies on a key insight: if we take the set of non-zero residues modulo p and multiply each of them by a, we get the *exact same set* back, just in a different order (i.e., permuted). We then equate the product of these two sets to reach our conclusion.

First, let's consider the trivial case for the main theorem, $a^p \equiv a \pmod{p}$:

- If a is a multiple of p, then $a \equiv 0 \pmod{p}$.
- Raising both sides to the power of p: $a^p \equiv 0^p \pmod{p}$, which means $a^p \equiv 0 \pmod{p}$.
- Since $a \equiv 0 \pmod{p}$ and $a^p \equiv 0 \pmod{p}$, we have $a^p \equiv a \pmod{p}$.

This case is proven.

Now, we prove the more complex case where a is not divisible by p. This means $\gcd(a, p) = 1$. Our goal is to prove $a^{p-1} \equiv 1 \pmod{p}$.

1. Define the Set of Residues: Consider the set S of all non-zero least positive residues modulo p: $S = \{1, 2, 3, \dots, p-1\}$ This set has $p-1$ elements.
2. Define the "Multiplied" Set: Let's create a new set, S', by multiplying every element in S by a, and taking the result modulo p: $S' = \{(a \cdot 1) \pmod{p}, (a \cdot 2) \pmod{p}, \dots, (a \cdot (p-1)) \pmod{p}\}$
3. Key Insight: S and S' are the same set. To prove this, we must show that the $p-1$ elements in S' are (1) all non-zero and (2) all distinct. Since S' would then contain $p-1$ distinct, non-zero values, and there are only $p-1$ such values available (the set S), S' must be a permutation of S.
   - (1) All elements of S' are non-zero: An element in S' is of the form $(a \cdot i) \pmod{p}$, where $i \in S$. For an element to be zero, we would need $a \cdot i \equiv 0 \pmod{p}$. Since p is a prime, this implies $p \mid a$ or $p \mid i$ (by Euclid's Lemma). We are in the case where p does not divide a ($\gcd(a, p) = 1$). And by definition of S, i is in the range $\{1, 2, \dots, p-1\}$, so p cannot divide i. Since p divides neither a nor i, it cannot divide their product. Therefore, $a \cdot i \not\equiv 0 \pmod{p}$, and no element in S' is zero.
   - (2) All elements of S' are distinct: Suppose two elements in S' are equal: $(a \cdot i) \equiv (a \cdot j) \pmod{p}$, for some $i, j \in S$. We need to show this implies $i = j$. Since $\gcd(a, p) = 1$, a has a multiplicative inverse a^{-1} modulo p. Multiply both sides of the congruence by a^{-1}: $a^{-1}(a \cdot i) \equiv a^{-1}(a \cdot j) \pmod{p}$ $(a^{-1}a)i \equiv (a^{-1}a)j \pmod{p}$

$1 \cdot i \equiv 1 \cdot j \pmod{p}$ $i \equiv j \pmod{p}$ Since i and j are both from the set $S = \{1, 2, \dots, p-1\}$, they must be equal. Thus, all elements in S' are distinct.

4. The "Rearrangement" Argument: The set S' contains $p-1$ elements, all of which are distinct and non-zero. The only set of $p-1$ distinct, non-zero residues modulo p is the set S itself. Therefore, the sets S and S' are identical, just possibly in a different order. $S = S' \pmod{p}$ $\{1, 2, \dots, p-1\} \equiv \{a \cdot 1, a \cdot 2, \dots, a \cdot (p-1)\} \pmod{p}$
5. Equate the Products: Since the two sets are the same, the product of all elements in S must be congruent to the product of all elements in S' (modulo p).
   Product of S: $1 \cdot 2 \cdot 3 \cdot \cdots \cdot (p-1) = (p-1)!$
   Product of S': $(a \cdot 1) \cdot (a \cdot 2) \cdot \cdots \cdot (a \cdot (p-1))$
   Equating them: $(p-1)! \equiv (a \cdot 1) \cdot (a \cdot 2) \cdot \cdots \cdot (a \cdot (p-1)) \pmod{p}$
6. Solve for a: Let's re-group the terms on the right side: $(p-1)! \equiv (a \cdot a \cdot \cdots \cdot a) \cdot (1 \cdot 2 \cdot \cdots \cdot (p-1)) \pmod{p}$ $(p-1)! \equiv a^{p-1} \cdot (p-1)! \pmod{p}$
7. The Cancellation Step: We want to cancel $(p-1)!$ from both sides. We can do this because $\gcd((p-1)!, p) = 1$. Why? $(p-1)! = 1 \cdot 2 \cdot \cdots \cdot (p-1)$. Since p is prime, its only prime factor is p. None of the numbers from 1 to $p-1$ contain p as a factor. Thus, their product is not divisible by p. Since $(p-1)!$ is coprime to p, it has a multiplicative inverse modulo p. Multiplying both sides by the inverse of $(p-1)!$: $1 \cdot (p-1)! \cdot ((p-1)!)^{-1} \equiv a^{p-1} \cdot (p-1)! \cdot ((p-1)!)^{-1} \pmod{p}$ $1 \equiv a^{p-1} \pmod{p}$
   This proves the second form of the theorem: $a^{p-1} \equiv 1 \pmod{p}$ for a not divisible by p.

We combine the results from Part 1 and Part 2.

- If a is not divisible by p (Part 2): We just proved $a^{p-1} \equiv 1 \pmod{p}$. Multiplying both sides by a gives: $a \cdot a^{p-1} \equiv a \cdot 1 \pmod{p}$ $a^p \equiv a \pmod{p}$
- If a is divisible by p (Part 1): We already showed $a^p \equiv a \pmod{p}$ (since both are $\equiv 0 \pmod{p}$).

In both possible cases, the congruence $a^p \equiv a \pmod{p}$ holds true. This completes the proof.

Simplifying Modular Arithmetic

Often, we may be given the problem of a small number raised to a large power modded by a different small number, such as finding $5^{561} \pmod{7}$.

By Fermat's Little Theorem, we know $a^{p-1} \equiv 1 \pmod{p}$. This means that we want the number to be raised to some *multiple* of $p-1$ (which in this case is 6).

- We only need some multiple of $p-1$ due to the properties of exponents.
- Say that we have $a^{(p-1)k} \equiv (a^{p-1})^k \pmod{p}$. This is no different from $(1)^k \pmod{p}$ by the properties of modular arithmetic and Fermat's Little Theorem.

$$5^{651} \equiv (5^6)^{100} \cdot (5^{51}) \equiv (5^6)^8 \cdot 5^3 \equiv 25 \cdot 5 \equiv 4 \cdot 5 \equiv 20 \pmod{7}$$

The previously-large exponent was reduced down to $20 \pmod{7}$, which is obviously 6.

8.7 Euler's Theorem

The Totient Function

Explanation and Derivation of Euler's Totient Function Formula

Euler's totient function (or phi function), denoted $\phi(n)$, is a central concept in number theory. It counts the number of positive integers less than or equal to n that are relatively prime (or coprime) to n.

1. Statement of the General Formula Let n be a positive integer with the unique prime factorization:

$$n = p_1^{k_1} p_2^{k_2} \cdots p_r^{k_r}$$

where $p_1, p_2, \ldots, p_r$ are the distinct prime factors of n, and $k_1, k_2, \ldots, k_r$ are their respective positive integer exponents.

The general formula for Euler's totient function is:

$$\phi(n) = n\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right)\cdots\left(1 - \frac{1}{p_r}\right)$$

This can be written compactly using product notation:

$$\phi(n) = n \prod_{p|n}\left(1 - \frac{1}{p}\right)$$

where the product is taken over all *distinct* prime factors p of n.

An equivalent form, which we will also derive, is:

$$\phi(n) = (p_1^{k_1} - p_1^{k_1-1})(p_2^{k_2} - p_2^{k_2-1}) \cdots (p_r^{k_r} - p_r^{k_r-1})$$

---

2. Key Concepts and Definitions

- Coprime (Relatively Prime): Two integers a and n are coprime if their greatest common divisor is 1, written as $\gcd(a, n) = 1$. This means they share no common factors other than 1.
- Prime Factorization: Every integer $n > 1$ can be uniquely expressed as a product of prime powers, as shown in the formula statement.
- The Core Idea (Counting Strategy): The function $\phi(n)$ counts the numbers k in the set $\{1, 2, \ldots, n\}$ such that $\gcd(k, n) = 1$. It is often easier to count the *opposite*: the numbers that are not coprime to n. A number k is *not* coprime to n if and only if it shares at least one prime factor with n. So, k is not coprime to n if k is divisible by at least one of the primes $p_1, p_2, \ldots, p_r$. Our strategy will be:

$$\phi(n) = (\text{Total numbers from 1 to } n) - (\text{Numbers from 1 to } n \text{ that are NOT coprime to } n)$$

$$\phi(n) = n - (\text{Count of numbers divisible by at least one } p_i)$$

---

3. Derivation using the Principle of Inclusion-Exclusion (PIE)

This derivation is a direct "sieve" method. We start with all n numbers and "sieve out" the ones that are not coprime.

1. Total Numbers: We start with the set $\{1, 2, \ldots, n\}$. The size is n.
2. The "Bad" Numbers: We want to count and remove any number k that is divisible by at least one of the distinct prime factors $p_1, p_2, \ldots, p_r$.
3. Applying PIE: We will use the Principle of Inclusion-Exclusion to count the size of the set of "bad" numbers.
   - Step A (Subtract multiples): First, we remove all multiples of each prime p_i.
     - The number of multiples of p_1 (up to n) is $\frac{n}{p_1}$.
     - The number of multiples of p_2 (up to n) is $\frac{n}{p_2}$.
     - ... and so on.
   - Step B (Add back double-counts): When we subtracted multiples of p_1 and p_2, we subtracted the multiples of $p_1 p_2$ *twice*. We must add them back.

- The number of multiples of p_1p_2 is $\frac{n}{p_1p_2}$.
- We do this for all pairs (p_i, p_j).

- Step C (Subtract triple-counts): We have now over-corrected. Multiples of $p_1p_2p_3$ were subtracted three times (for p_1, p_2, p_3) and then added back three times (for p_1p_2, p_1p_3, p_2p_3). The net effect is they haven't been removed at all. We must now subtract them.
  - The number of multiples of $p_1p_2p_3$ is $\frac{n}{p_1p_2p_3}$.
- This continues...

4. Formalizing with PIE: The count of numbers *not* coprime to n is:
$$\left(\sum_i \frac{n}{p_i}\right) - \left(\sum_{i<j} \frac{n}{p_ip_j}\right) + \left(\sum_{i<j<k} \frac{n}{p_ip_jp_k}\right) - \cdots$$
5. Calculating $\phi(n)$:
$$\phi(n) = n - \left[\left(\sum \frac{n}{p_i}\right) - \left(\sum \frac{n}{p_ip_j}\right) + \left(\sum \frac{n}{p_ip_jp_k}\right) - \cdots\right]$$
6. Factoring out n:
$$\phi(n) = n\left[1 - \left(\sum \frac{1}{p_i}\right) + \left(\sum \frac{1}{p_ip_j}\right) - \left(\sum \frac{1}{p_ip_jp_k}\right) + \cdots\right]$$
7. The Algebraic Insight: Let's look at the algebraic expansion of the product we are trying to prove:
$$\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right)\cdots\left(1 - \frac{1}{p_r}\right)$$
When you expand this product, you get:
   - The first term is $1 \times 1 \times \cdots \times 1 = 1$.
   - The next terms come from choosing one fraction and $r-1$ ones: $-\frac{1}{p_1} - \frac{1}{p_2} - \cdots = -\left(\sum \frac{1}{p_i}\right)$.
   - The next terms come from choosing two fractions and $r-2$ ones: $\left(-\frac{1}{p_1}\right)\left(-\frac{1}{p_2}\right) + \cdots = +\left(\sum \frac{1}{p_ip_j}\right)$.
   - The next terms come from choosing three fractions: $-\left(\sum \frac{1}{p_ip_jp_k}\right)$.
   - This continues until the last term, $(-1)^r \frac{1}{p_1p_2\cdots p_r}$.
   - The expansion is: $1 - \left(\sum \frac{1}{p_i}\right) + \left(\sum \frac{1}{p_ip_j}\right) - \left(\sum \frac{1}{p_ip_jp_k}\right) + \cdots$
8. Conclusion: The expression in the brackets [...] from step 6 is *exactly* the algebraic expansion of our target product. Therefore, we can replace the brackets with the product form:
$$\phi(n) = n\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right)\cdots\left(1 - \frac{1}{p_r}\right) = n\prod_{p|n}\left(1 - \frac{1}{p}\right)$$
This completes the derivation.

---

4. Alternative Derivation (Using Multiplicativity)

This method relies on two key properties.

Property 1: Formula for $\phi(p^k)$ Let's find the totient of a single prime power, $n = p^k$.

- Total numbers: $\{1, 2, \ldots, p^k\}$. Count is p^k.
- Numbers *not* coprime to p^k: These are the numbers that share a factor with p^k. The only prime factor is p. So, we just need to count the multiples of p.

- The multiples of p are: $\{1p, 2p, 3p, \dots, (p^{k-1})p\}$.
- The last multiple is $p^{k-1} \times p = p^k$. So, there are exactly p^{k-1} such multiples.
- $\phi(p^k) = (\text{Total}) - (\text{Not coprime}) = p^k - p^{k-1}$.
- We can factor this: $\phi(p^k) = p^k\left(1 - \frac{p^{k-1}}{p^k}\right) = p^k\left(1 - \frac{1}{p}\right)$. This matches the general formula for a single prime factor.

Property 2: $\phi(n)$ is a Multiplicative Function A function f is multiplicative if for any two coprime integers m and n (i.e., $\gcd(m, n) = 1$), we have $f(mn) = f(m)f(n)$. Using the Chinese Remainder Theorem, it can be proven that $\phi(n)$ is a multiplicative function.

Combining the Properties:

1. Start with the prime factorization of n: $n = p_1^{k_1} p_2^{k_2} \cdots p_r^{k_r}$.
2. Each term $p_i^{k_i}$ is coprime to every other term $p_j^{k_j}$ (since they are powers of different primes).
3. We can apply the multiplicative property repeatedly:
$$\phi(n) = \phi(p_1^{k_1} p_2^{k_2} \cdots p_r^{k_r}) = \phi(p_1^{k_1}) \times \phi(p_2^{k_2}) \times \cdots \times \phi(p_r^{k_r})$$
4. Now, substitute the formula from Property 1 ($\phi(p^k) = p^k - p^{k-1}$) for each term:
$$\phi(n) = (p_1^{k_1} - p_1^{k_1-1})(p_2^{k_2} - p_2^{k_2-1}) \cdots (p_r^{k_r} - p_r^{k_r-1})$$
(This is the second form of the formula.)
5. Now, factor each of these terms as we did in Property 1:
$$\phi(n) = \left[p_1^{k_1}\left(1 - \frac{1}{p_1}\right)\right]\left[p_2^{k_2}\left(1 - \frac{1}{p_2}\right)\right] \cdots \left[p_r^{k_r}\left(1 - \frac{1}{p_r}\right)\right]$$
6. Group all the $p_i^{k_i}$ terms together and all the $\left(1 - \frac{1}{p_i}\right)$ terms together:
$$\phi(n) = \left(p_1^{k_1} p_2^{k_2} \cdots p_r^{k_r}\right) \times \left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right) \cdots \left(1 - \frac{1}{p_r}\right)$$
7. The first large group ($p_1^{k_1} p_2^{k_2} \cdots p_r^{k_r}$) is just the prime factorization of n itself.
$$\phi(n) = n\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right) \cdots \left(1 - \frac{1}{p_r}\right)$$
This completes the alternative derivation.

---

5. Worked Example

Let's calculate $\phi(12)$.

- Step 1: Prime Factorization $n = 12 = 4 \times 3 = 2^2 \times 3^1$. The distinct prime factors are $p_1 = 2$ and $p_2 = 3$. The exponents are $k_1 = 2$ and $k_2 = 1$.
- Step 2: Apply the Product Formula
$$\phi(12) = 12\left(1 - \frac{1}{2}\right)\left(1 - \frac{1}{3}\right)$$
$$\phi(12) = 12\left(\frac{1}{2}\right)\left(\frac{2}{3}\right)$$
$$\phi(12) = 12\left(\frac{2}{6}\right) = 12\left(\frac{1}{3}\right)$$
$$\phi(12) = 4$$

- Alternative Formula: Using $\phi(n) = \phi(p_1^{k_1})\phi(p_2^{k_2})$, we get:

$$\phi(12) = \phi(2^2) \times \phi(3^1)$$
$$\phi(12) = (2^2 - 2^{2-1}) \times (3^1 - 3^{1-1})$$
$$\phi(12) = (2^2 - 2^1) \times (3^1 - 3^0)$$
$$\phi(12) = (4 - 2) \times (3 - 1)$$
$$\phi(12) = 2 \times 2 = 4$$

- Verification (Manual Counting): Let's list the numbers from 1 to 12 and find the ones coprime to 12 (i.e., not divisible by 2 or 3). Set: $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$
  - $\gcd(1, 12) = 1$ (Coprime)
  - $\gcd(2, 12) = 2$
  - $\gcd(3, 12) = 3$
  - $\gcd(4, 12) = 4$
  - $\gcd(5, 12) = 1$ (Coprime)
  - $\gcd(6, 12) = 6$
  - $\gcd(7, 12) = 1$ (Coprime)
  - $\gcd(8, 12) = 4$
  - $\gcd(9, 12) = 3$
  - $\gcd(10, 12) = 2$
  - $\gcd(11, 12) = 1$ (Coprime)
  - $\gcd(12, 12) = 12$ The set of coprime numbers is $\{1, 5, 7, 11\}$. The count is 4. All three methods agree.

A Note on the Properties of the Totient Function

1. If $gcd(n, m) = 1$, then:
$$\phi(mn) = \phi(m)\phi(n)$$
2. If p is a prime, then $\phi(p) = p - 1$.

The above two combined is a powerful method of calculating the value of a totient function for some large n which is the product of a few primes.

Simplifying Modular Arithmetic

Just as Euler's Theorem is a generalization of Fermat's Little Theorem, while Fermat's Little Theorem is helpful for simplifying modular arithmetic with prime moduli, Euler's Theorem can be used with composite moduli.

Consider finding $7^{821} \pmod{15}$. Euler's Theorem is able to work with composite moduli like 15, so we can claim that $7^{\phi(15)} \equiv 1 \pmod{15}$.

Computing $\phi(15)$, we can say $\phi(15) = \phi(3)\phi(5) = (3 - 1)(5 - 1) = 8$. Then $7^8 \equiv 1 \pmod{15}$.

- This gives us great leverage in modular arithmetic to simplify math.

$$7^{821} \equiv (7^8)^{102}(7^5) \equiv 1 \cdot (49)^2 \cdot 7 \equiv 4^2 \cdot 7 \equiv 16 \cdot 7 \equiv 1 \cdot 7 \pmod{15}$$

Proof of Euler's Theorem

1. Statement of the Proof Let a and n be positive integers such that $\gcd(a, n) = 1$ (meaning a and n are coprime, or share no common factors other than 1). Let $\phi(n)$ be Euler's totient function.

Then, Euler's Theorem states:

$$a^{\phi(n)} \equiv 1 \pmod{n}$$

- If n is a prime, then this becomes Fermat's Little Theorem: $a^{p-1} \equiv 1 \pmod{p}$.

---

2. Necessary Context

- Euler's Totient Function $\phi(n)$: The totient function $\phi(n)$ (also called Euler's phi function) counts the number of positive integers less than or equal to n that are relatively prime (coprime) to n.
  - Example: For $n = 10$, the numbers less than or equal to 10 are $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. The numbers coprime to 10 are those that do not share a factor of 2 or 5. These are $\{1, 3, 7, 9\}$.
  - Therefore, $\phi(10) = 4$.
- Reduced Residue System (RRS) Modulo n: This is the set of all integers r such that $1 \leq r < n$ and $\gcd(r, n) = 1$.
  - Intuitive Note: This is simply the set of numbers that the totient function $\phi(n)$ counts. The size of this set is, by definition, $\phi(n)$.
  - Example (revisited): The RRS for $n = 10$ is the set $R = \{1, 3, 7, 9\}$. The size of this set is $|R| = \phi(10) = 4$.

---

3. The Core Lemma: Multiplication Permutes the RRS

This is the central insight of the proof.

Lemma: Let $R = \{r_1, r_2, \ldots, r_{\phi(n)}\}$ be the reduced residue system modulo n. If $\gcd(a, n) = 1$, then the set $aR = \{a \cdot r_1 \pmod{n}, a \cdot r_2 \pmod{n}, \ldots, a \cdot r_{\phi(n)} \pmod{n}\}$ is simply a *permutation* (a reordering) of the original set R.

Intuitive Note: This lemma says that if you take the set of all numbers coprime to n (our RRS) and multiply each of them by a (which is also coprime to n), you just get the same set back, but with the elements "shuffled" into a different order.

Proof of the Lemma: To prove this, we must show two things:

1. Every element $a \cdot r_i$ in the new set aR is *also* an element of R.
2. All the elements in the new set aR are *distinct* (i.e., no two different elements r_i and r_j from R "collide" and become the same element in aR).

- Part 1 (Closure): We need to show that $a \cdot r_i$ is also coprime to n. We are given $\gcd(a, n) = 1$ and $\gcd(r_i, n) = 1$ (because r_i is in the RRS). A fundamental property of the greatest common divisor is that if a and n are coprime, and r_i and n are coprime, then their product $(a \cdot r_i)$ is also coprime to n. Thus, $\gcd(a \cdot r_i, n) = 1$. This means $a \cdot r_i \pmod{n}$ is, by definition, an element of the reduced residue system R.
- Part 2 (Distinctness): We need to show that if r_i and r_j are two *different* elements of R, then $a \cdot r_i \pmod{n}$ and $a \cdot r_j \pmod{n}$ are also different. Let's prove this by contradiction. Assume they are *not* distinct, i.e., they "collide." Suppose $a \cdot r_i \equiv a \cdot r_j \pmod{n}$ for two elements $r_i, r_j \in R$. This means $a \cdot r_i - a \cdot r_j \equiv 0 \pmod{n}$. Factoring out a, we get $a(r_i - r_j) \equiv 0 \pmod{n}$. This means n divides the product $a(r_i - r_j)$. However, we are given that $\gcd(a, n) = 1$. Since n shares no factors with a, it *must* divide the other part of the product. Therefore, n must divide $(r_i - r_j)$, which means $r_i \equiv r_j \pmod{n}$. Since r_i and r_j are both in the RRS (meaning $1 \leq r < n$), the only way they can be congruent modulo n is if they are the same number: $r_i = r_j$. This completes our proof by contradiction. We have shown that if $r_i \neq r_j$, then $a \cdot r_i \not\equiv a \cdot r_j \pmod{n}$.

Since the set aR contains $\phi(n)$ distinct elements, all of which are in R, and R itself only has $\phi(n)$ elements, the set aR must be the *same set* as R, just in a different order.

---

4. The Main Proof of Euler's Theorem

Now we assemble the pieces.

1. Let $R = \{r_1, r_2, \dots, r_{\phi(n)}\}$ be the reduced residue system modulo n.
2. From our Lemma, we know the set $aR = \{a \cdot r_1, a \cdot r_2, \dots, a \cdot r_{\phi(n)}\}$ (all $\pmod n$) is a permutation of R.
3. Since the two sets R and aR contain the exact same elements (just shuffled), the *product* of all elements in R must be congruent to the *product* of all elements in aR, modulo n.

$$(r_1 \cdot r_2 \cdot \dots \cdot r_{\phi(n)}) \equiv (a \cdot r_1) \cdot (a \cdot r_2) \cdot \dots \cdot (a \cdot r_{\phi(n)}) \pmod n$$

4. Let P be the product of all elements in R: $P = \prod_{i=1}^{\phi(n)} r_i$. We can rewrite the congruence from step 3 as:

$$P \equiv \left(a^{\phi(n)}\right) \cdot (r_1 \cdot r_2 \cdot \dots \cdot r_{\phi(n)}) \pmod n$$

$$P \equiv a^{\phi(n)} \cdot P \pmod n$$

5. Now, we need to "cancel" P from both sides. We can't just "divide" in modular arithmetic. Instead, we must multiply by the modular multiplicative inverse. We can only do this if P has an inverse, which is true if and only if $\gcd(P, n) = 1$.
   - Justification: P is the product of all elements r_i in the RRS. By definition, *every* r_i is coprime to n (i.e., $\gcd(r_i, n) = 1$). Since P is a product of numbers that are all coprime to n, P itself must be coprime to n.
   - Therefore, $\gcd(P, n) = 1$, and P has a multiplicative inverse P^{-1} modulo n.
6. Multiply both sides of our congruence $P \equiv a^{\phi(n)} \cdot P \pmod n$ by P^{-1}:

$$P^{-1} \cdot P \equiv P^{-1} \cdot a^{\phi(n)} \cdot P \pmod n$$

$$1 \equiv a^{\phi(n)} \cdot 1 \pmod n$$

$$1 \equiv a^{\phi(n)} \pmod n$$

7. This can be rewritten as:

$$a^{\phi(n)} \equiv 1 \pmod n$$

This completes the proof.

---

5. Conclusion and Relation to Fermat's Little Theorem

We have rigorously shown that for any integer a coprime to n, $a^{\phi(n)} \equiv 1 \pmod n$.

Special Case (Fermat's Little Theorem): This theorem is a generalization of Fermat's Little Theorem.

- Let $n = p$, where p is a prime number.
- By definition, $\phi(p)$ is the count of numbers from 1 to $p - 1$ that are coprime to p.
- Since p is prime, *all* numbers $\{1, 2, \dots, p - 1\}$ are coprime to p.
- Therefore, $\phi(p) = p - 1$.
- Substituting $n = p$ and $\phi(n) = p - 1$ into Euler's Theorem, we get:

$$a^{p-1} \equiv 1 \pmod p$$

...which is precisely Fermat's Little Theorem (for any a not divisible by p, i.e., $\gcd(a, p) = 1$).

8.8 Chinese Remainder Theorem

Statement

Let $n_1, n_2, \dots, n_k$ be a set of positive integers that are pairwise coprime, meaning $\gcd(n_i, n_j) = 1$ for all $i \neq j$. Then, for any set of arbitrary integers $a_1, a_2, \dots, a_k$, the system of simultaneous congruences:

$$\begin{aligned} x &\equiv a_1 \pmod{n_1} \\ x &\equiv a_2 \pmod{n_2} \\ &\vdots \\ x &\equiv a_k \pmod{n_k} \end{aligned}$$

has a solution. Furthermore, any two solutions, x_0 and x_1, are congruent modulo N, where $N = n_1 \cdot n_2 \cdots n_k$. This means the system has a unique solution modulo N.

Proof of CRT

Let $n_1, n_2, \dots, n_k$ be integers that are pairwise coprime (i.e., $gcd(n_i, n_j) = 1$ for any $i \neq j$). Then for any sequence of integers $a_1, a_2, \dots, a_k$, the system of congruences:

$$\begin{cases} x \equiv a_1 \pmod{n_1} \\ x \equiv a_2 \pmod{n_2} \\ \vdots \\ x \equiv a_k \pmod{n_k} \end{cases}$$

has a solution x. Furthermore, this solution is unique modulo N, where $N = n_1 \cdot n_2 \cdot \dots \cdot n_k$.

Intuition The core idea is to find k special numbers, let's call them $e_1, e_2, \dots, e_k$. We will design these numbers to act like "switches" or "basis vectors".

We want each e_i to have two properties:

1. It is congruent to 1 modulo n_i. ($e_i \equiv 1 \pmod{n_i}$)
2. It is congruent to 0 modulo all *other* moduli n_j. ($e_i \equiv 0 \pmod{n_j}$ for all $j \neq i$)

If we can find these k numbers, constructing the final solution x is easy. We just "dial in" the required a_i for each congruence: $x = a_1e_1 + a_2e_2 + \dots + a_ke_k$

Why does this work? Let's check this proposed x against one of the moduli, say n_1: $x \equiv (a_1e_1 + a_2e_2 + \dots + a_ke_k) \pmod{n_1}$

By our design, $e_2, e_3, \dots, e_k$ are all $0 \pmod{n_1}$. And e_1 is $1 \pmod{n_1}$. So the equation simplifies:

$$\begin{aligned} x &\equiv (a_1 \cdot 1 + a_2 \cdot 0 + \dots + a_k \cdot 0) \pmod{n_1} \\ x &\equiv a_1 \pmod{n_1} \end{aligned}$$

This works for all n_i! So, the entire proof boils down to one task: how to build e_i.

Let's focus on building one of these special numbers, say e_i.

1. Satisfy the "0" condition: We need $e_i \equiv 0 \pmod{n_j}$ for all $j \neq i$. This means e_i must be a multiple of $n_1, n_2, \dots$ (skipping n_i), $\dots, n_k$. Since all moduli are pairwise coprime, e_i must be a multiple of their product. Let's define $N = n_1 \cdot n_2 \cdot \dots \cdot n_k$. Let's define $N_i = N/n_i = n_1 \cdot n_2 \cdot \dots \cdot n_{i-1} \cdot n_{i+1} \cdot \dots \cdot n_k$. So, e_i must be of the form $e_i = c \cdot N_i$ for some integer c. This guarantees it is 0 modulo all other n_j.

2. Satisfy the "1" condition: Now, we just need to find the c that makes $e_i \equiv 1 \pmod{n_i}$. $c \cdot N_i \equiv 1 \pmod{n_i}$ This means c is the modular inverse of N_i (modulo n_i). Let's call this inverse y_i. So, $c = y_i = N_i^{-1} \pmod{n_i}$.
3. Does this inverse y_i always exist? Yes. A modular inverse $N_i^{-1} \pmod{n_i}$ exists if and only if $gcd(N_i, n_i) = 1$.
   - N_i is the product of all n_j where $j \neq i$.
   - The theorem's main condition is that n_i is coprime to *all* other n_j.
   - Since n_i shares no prime factors with any of the n_j in the product N_i, it cannot share any prime factors with N_i itself.
   - Therefore, $gcd(N_i, n_i) = 1$, and the inverse $y_i \equiv N_i^{-1} \pmod{n_i}$ is guaranteed to exist.
4. Assemble e_i and the final solution x: We found $c = y_i$, so $e_i = y_i \cdot N_i$. Plugging this back into our formula for x:
   $x = \sum_{i=1}^{k} a_i e_i = \sum_{i=1}^{k} a_i y_i N_i$

This x is a solution. This completes the constructive proof.

We've shown *a* solution exists. Now we show it's the *only* one (modulo N).

Suppose x and x' are two different solutions to the system. This means for all i: $x \equiv a_i \pmod{n_i}$ $x' \equiv a_i \pmod{n_i}$

Let's look at their difference, $d = x - x'$: $d = x - x' \equiv a_i - a_i \equiv 0 \pmod{n_i}$

This is true for *all* i from 1 to k.

- $d \equiv 0 \pmod{n_1}$ means d is a multiple of n_1.
- $d \equiv 0 \pmod{n_2}$ means d is a multiple of n_2.
- ...
- $d \equiv 0 \pmod{n_k}$ means d is a multiple of n_k.

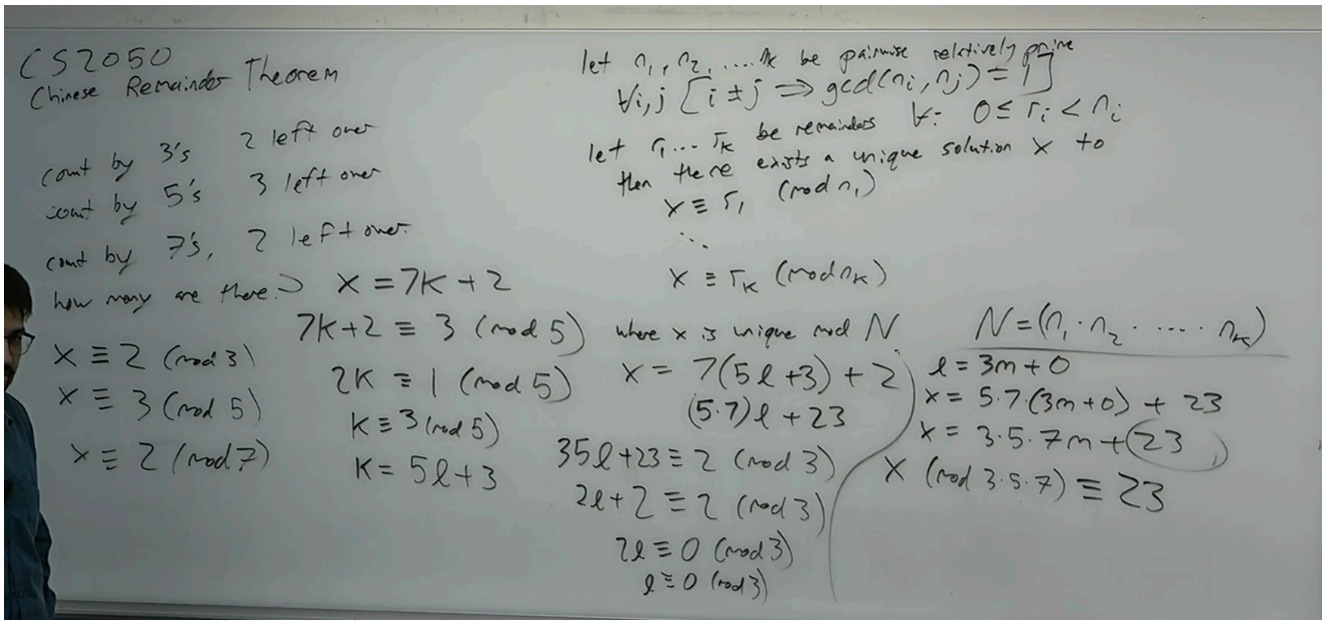
Since d is a common multiple of all n_i, and all n_i are pairwise coprime, d must be a multiple of their least common multiple. The LCM of pairwise coprime numbers is simply their product: $lcm(n_1, n_2, \dots, n_k) = n_1 \cdot n_2 \cdot \dots \cdot n_k = N$.

So, d must be a multiple of N. $d \equiv 0 \pmod{N}$ $x - x' \equiv 0 \pmod{N}$ $x \equiv x' \pmod{N}$

This proves that any two solutions x and x' are congruent modulo N. Therefore, the solution is unique modulo N.

Solving Modular Systems with CRT

1. Check whether all of the moduli are pairwise coprime.
2. Rewrite the congruence expression with the largest modulus into $x = n_i k + a_i$.
3. Plug this value of x into the equation with the next-largest modulus. Simplify using modular arithmetic until we have $k = n_{i-1} l + a_{i-1}$. Now rewrite x in terms of this "smaller variable" l.
4. Proceed to do this over and over until the last congruence expression is reached (with the smallest modulus). At that point, we can find a value for the "smallest variable" (let's call it m). Use that value to find x.
   - We must proceed to do this repeatedly because before reaching the last iteration, any value that we find for any variable (e.g., k, l) is just one that "happens to work".
   - If we decide to plug back in to find x using any intermediate variable, x is NOT guaranteed to fulfill the remaining equations in the system.



Simplifying Modular Arithmetic

First, we had Fermat's Little Theorem, which worked well with moduli which were small and prime. Then, Euler's Theorem allowed for moduli which were composite, but still relatively small.

HOWEVER, there will be some problems, where the moduli is a *large composite number*.

CRT allows conversion of modular arithmetic from a composite modulus to prime power moduli. When you have a large composite modulus $n = p_1^{k_1} \cdot p_2^{k_2} \cdot \ldots \cdot p_m^{k_m}$, CRT enables solving the congruence:

$$x \equiv a \pmod{n}$$

as an equivalent system:

$$x \equiv a_1 \pmod{p_1^{k_1}}, \quad x \equiv a_2 \pmod{p_2^{k_2}}, \quad \ldots, \quad x \equiv a_m \pmod{p_m^{k_m}}$$

where the moduli are pairwise coprime. This decomposition is crucial because working modulo prime powers is significantly simpler than working modulo the original composite number.

Critically, once decomposed into prime power moduli, Euler's Theorem can be applied to each component individually (where coprimality is more likely to hold), and the exponent can be reduced by taking it modulo $\phi(p_i^{k_i})$ for each prime power.

8.9 RSA Cryptography

RSA is an asymmetric encryption algorithm whose security relies on the computational difficulty of factoring large numbers.

The key space determines the difficulty of brute-force attacks—if the key space has size k, an adversary must perform up to 2^k operations to guess the key correctly.

- A cryptographic system is considered secure when the best strategy available to an attacker is to perform brute-force key guessing.

Symmetric vs Asymmetric Cryptography

Symmetric cryptography uses the same key for both encryption and decryption. The fundamental challenge is key distribution: if parties A and B need to communicate securely, A must somehow share the key with B without allowing an adversary to intercept it.

Asymmetric cryptography solves this problem using a tuple of algorithms `(G, E, D)`.

- The key generation algorithm `G` uses randomness to produce a public key `Pk` and a secret key `Sk`.
- To send a message to B, any party must encrypt using B's public key.
- B can receive encrypted messages from anyone, but only B's secret key (which is never shared) can decrypt them.

Mathematical Foundations

The existence of public-key cryptography assumes that P ≠ NP. RSA's security depends on two key computational problems from number theory:

- Primality Testing: Determining whether a number n is prime can be done efficiently. Fermat's test states that if a^(n-1) ≡ 1 (mod n) and gcd(a,n) = 1, then n is likely prime.
- Factoring Hardness: While primality testing is efficient, factoring large composite numbers into their prime factors remains computationally hard (though this hardness is an unproven assumption in number theory).

If N = pq where p and q are large primes, computing Euler's totient function φ(N) = (p-1)(q-1) is efficient only when p and q are known.

RSA Key Generation

1. Find two large prime numbers p and q
2. Compute N = pq and φ(N) = (p-1)(q-1)
3. Choose an encryption exponent e such that gcd(e, φ(N)) = 1 (typically e = 17 is chosen)
4. Compute the decryption exponent d such that d ≡ e^(-1) (mod φ(N))
5. Public key: (N, e); Secret key: d

Encryption and Decryption

Encryption transforms a message m (represented as a number) into ciphertext: c ≡ m^e (mod N)

Decryption recovers the original message: m ≡ c^d (mod N)

The security equivalence states that if factoring N is hard, then RSA is secure. This is because knowing φ(N) = (p-1)(q-1) allows computation of d ≡ e^(-1) (mod φ(N)), which breaks the system. Therefore, factoring N into p and q would compromise RSA's security.

8.10 Diffie-Hellman Key Exchange

Diffie-Hellman is a symmetric key-sharing system that allows two parties to establish a shared secret over an insecure channel. It operates within a mathematical group structure, typically the multiplicative group (Z/pZ)* for a prime p. A generator g is an element whose powers produce all elements of the group.

Discrete Logarithm Problem

The security of Diffie-Hellman relies on the discrete logarithm problem: given g, p, and g^x (mod p), finding x is computationally hard. This problem forms the foundation of the system's security.

Protocol Steps

1. Parties A and B publicly agree on a generator g and prime p
2. A chooses a secret value x and sends g^x (mod p) to B
3. B chooses a secret value y and sends g^y (mod p) to A
4. Both parties compute the shared secret: A computes (g^{y})x (mod p) and B computes (g^{x})y (mod p)
5. Both computations yield g^(xy) (mod p), which becomes their shared secret key

The elegance of Diffie-Hellman is that an eavesdropper sees only g^x and g^y but cannot efficiently compute g^(xy) without solving the discrete logarithm problem to find either x or y.

9 Combinatorics

9.1 Introduction to Combinatorics

Combinatorics is about the art of counting the number of ways to do something, or the number of ways an event can occur.

Sum and Product Rules

The Sum and Product rules are often very intuitive, but must also be defined precisely to avoid misuse.

The Product Rule

If there are m ways to do one independent event and n ways to do another, then to do the two actions there are a total of $m \times n$ ways. Here, an independent event points to the condition that the outcome of one event does not affect the probability of another. That is, one does not influence the number of ways to do the other.

Consider it to be when you can decompose an action into smaller, potentially-ordered steps. Order helps to intuitively find the smaller steps, but is not necessary due to commutativity of addition (it would work just as well going backwards in the steps).

The Sum Rule

If there are disjoint sets of choices, say one with m options and one with n options, then the total number of ways to choose one option from either set is $m + n$.

- This appears in set theory as: $|A \cup B| = |A| + |B|$.
- An alternative way of phrasing this is if there are a few different disjoint cases for ways to reach a target outcome, then the number of ways to achieve each disjoint and exhaustive case can be summed to obtain all of the ways to reach the target outcome.

Combinatorial Proofs

Combinatorial proofs provide a method to solve counting problems by establishing a direct correspondence between two sets. Most importantly, a bijection must be established between the two things being counted. This way, finding a one-to-one mapping between a hard-to-count problem set (A) and an easily countable set (B) proves that their sizes are equal, denoted as $|A| = |B|$.

In simpler terms, combinatorial analysis allows us to prove if two sides of an equation are equal by describing the same scenario but using different methods of counting. One may be much simpler/direct, while the other may require more setup and a specific path to think through.

Note: Combinatorics is not very concerned with finding out the exact values or methods of achieving an event, but rather the existence and countability of solutions. Typically, the bijection is established through some analogy which is intuitive for the reader of the proof and the bijection is not formally/rigorously defined, at least at this level.

Example

Prove that $\sum_{i=0}^{n} \binom{n}{i} 2^i = 3^n$.

Consider a ternary string composed of 0, 1, and 2. We prove equality by showing that the LHS and RHS are two different ways to count the same thing: the number different of ternary strings able to be formed of length n.

First, consider the RHS. At each index in the ternary string of length n, we can choose 1 of 3 options independently, so by the Product Rule the total number of ways to form a ternary string of length n is 3^n.

Then, consider the LHS. Let i be the number of nonzero numbers in the ternary string. Suppose that we fixed i between 0 and n, so that forming such a ternary string would involve 1) Selecting i indices from a set of n indices

to be nonzero values, and then 2) For each index of a nonzero value, we have 2 independent options (either 1 or 2), so 2^i possible choices. By the Product Rule, for a fixed i, we have $\binom{n}{i}2^i$. Now by the Sum Rule, we can add the number of ways to form such a ternary string for each $0 \leq i \leq n$, as each scenario is disjoint and exhaustive. Then we obtain $\sum_{i=0}^{n} \binom{n}{i}2^i$.

9.2 Counting and Distinguishability

The Factorial

The factorial determines how to compute the number of unique ways to arrange n distinct objects.

- Objects should be distinct, and when fixing one object at an index/position, it is removed from the set of possible objects that can be placed at a different posiiton.

Example

The number of ways to rearrange the letters "ABC" is 3!. At the first index, there are three possibilities. At the second, because one letter has been taken up, there are two possibilities. And at the last there is only one remaining possible letter to use.

Permutations

Let S be a finite set containing n distinct items. A k-permutation of S is an ordered sequence of k distinct elements from S, where $0 \leq k \leq n$, fulfilling the following:

- $\forall i, x_i \in S$
- $\forall i \neq j, x_i \neq x_j$

The number of such sequences able to be formed is denoted as $P(n, k)$, or ${}_nP_k$, is calculated as:

$$P(n, k) = \frac{n!}{(n-k)!}$$

The denominator comes about as a result for accounting the overcounting of sequences. Since only k objects need to be ordered/considered, we do not want to include the arrangements of the last $(n - k)$ objects where the first k (the ones we care about) are the same/indistinguishable.

Distinct Permutations of a Multiset

A multiset is an ordered collection of items where each item may appear more than once.

The concept of distinguishability is crucial when permuting items that are not unique. If we are permuting the letters of the word "SUCCESS," we cannot simply use 7! because the 'S's and 'C's are indistinguishable.

For the sake of overcounting and later correction though, just consider all n (in this case, 7) letters to be distinct. Then clearly we have 7! arrangements. Now disregard all other letters temporarily, and consider only the 2 C's and the 2 S's. There are 2! ways to permute them respectively, which are identical to the outside observer. That means that we have overcounted by a factor of 2!. For each letter, we divide to correct.

Generally, if there are n total items with n_1 items of type 1 through n_k items of type k, the number of distinguishable permutations is:

$$\frac{n!}{n_1! \ldots n_k!}.$$

Combinations

Let S be a finite set containing n distinct items. A k-combination of S is a subset of S containing exactly k elements, where $0 \leq k \leq n$.

Combinations, denoted as $C(n, k)$ or $\binom{n}{k}$, count the number of ways to group k elements from a set of n elements where order does not matter.

$$C(n, k) = \binom{n}{k} = \frac{n!}{k!(n-k)!}$$

The formula is derived by taking the permutation formula and further dividing by $k!$ to correct for "overcounting" the arrangements of the chosen items.

Note: A key symmetry in combinations is that choosing k items to include is mathematically identical to choosing $n - k$ items to exclude, meaning $\binom{n}{k} = \binom{n}{n-k}$.

Stars and Bars

The core of "Stars and Bars" is a commonly-used bijection (derived from a combinatorial proof) regarding the number of ways to distribute *indistinguishable* objects into *distinguishable* bins (e.g., how to split $10, which is indistinguishable, between 3 people, who are distinguishable).

Statement: Suppose you have n indistinguishable items, represented as stars (*) to distribute into k distinguishable bins. To separate these stars into k groups, you need $k - 1$ dividers (bars/bin edges).

We can then simplify the problem by defining a bijection with a string of symbols consisting of n stars (representing items) and $k-1$ bars (representing dividers between bins). The total number of arrangements is the number of ways to choose positions for the bars (or stars) within the total length of the string, expressed as $\binom{n+k-1}{k-1}$ or equivalently $\binom{n+k-1}{n}$.

- Either choose the indices for bars in the string $(k - 1)$ from the set of total indices or the indices of the stars (n).
  - At this point, each bar is indistinguishable, and thus we use combination since a set of indices of (1, 3, 5) is identical of (5, 3, 1).

Case 1: Non-Negative Integers (Empty Bins Allowed) In the standard form of the theorem, it is acceptable for a bin to be empty (e.g., for two dividers to be immediately next to one another in terms of indices). This uses the formula defined above.

Case 2: Positive Integers Only (No Empty Bins) In the case that no bin is allowed to be empty, the problem's bijection is slightly adjusted: Count the number of ways to insert $(k - 1)$ bars in the available $(n - 1)$ gaps between stars.

For example, with 5 stars, we have 4 gaps in between for potential slots. Thus, consider the set of indices to choose where to place bars to be of size $n - 1 = 4$.

$$* * * * *$$

Now let us create k non-empty bins. To do so, we need $(k - 1)$ bars, each of which can be inserted at one gap.

Then we have:

$$\binom{n-1}{k-1}$$

Example (TODO)

9.3 The Binomial Theorem

The Binomial Theorem states that $(x+y)^n = \sum_{i=0}^{n} \binom{n}{i} x^{n-i} y^i$.

Consider expanding $(x+y)^n$, which involves multiplying $(x+y)$ terms together. Then terms are formed b multiplying one variable chosen from each of the n factors, where each term is a product of n variables (some x's and some y's). Disregard the commutativity of multiplication temporarily, and consider all of the ordered sequences of $xx \ldots xx$, $xx \ldots xy$, $xx \ldots yx$, and so on.

The coefficient of any $x^{n-i}y^i$ will be the number of ways to form a combination of $n-i$ x's and i y's after summing them together by commutativity of multiplication. Now considering that having i y's means having $n-i$ x's, the number of ways to choose i from n is exactly $\binom{n}{i}$, being the coefficient for each term/combination of x/y.

Pascal's Triangle / Pascal's Identity

Consider Pascal's triangle, where each entry on the k-th row of the triangle corresponds to the coefficients of terms in $(x+y)^k$.

Exponent	Pascal's Triangle	Binomial Expansion
0	1	$(a+b)^0 = 1$
1	1 1	$(a+b)^1 = 1a + 1b$
2	1 2 1	$(a+b)^2 = 1a^2 + 2ab + 1b^2$
3	1 3 3 1	$(a+b)^3 = 1a^3 + 3a^2b + 3ab^2 + 1b^3$
4	1 4 6 4 1	$(a+b)^4 = 1a^4 + 4a^3b + 6a^2b^2 + 4ab^3 + 1b^4$
5	1 5 10 10 5 1	$(a+b)^5 = 1a^5 + 5a^4b + 10a^3b^2 + 10a^2b^3 + 5ab^4 + 1b^5$
6		

Most importantly, each term in the subsequent row can be determined by summing the two entries above it, so long as it is not a 1. This is stated as Pascal's Identity:

$$\binom{n}{k-1} + \binom{n}{k} = \binom{n+1}{k}$$

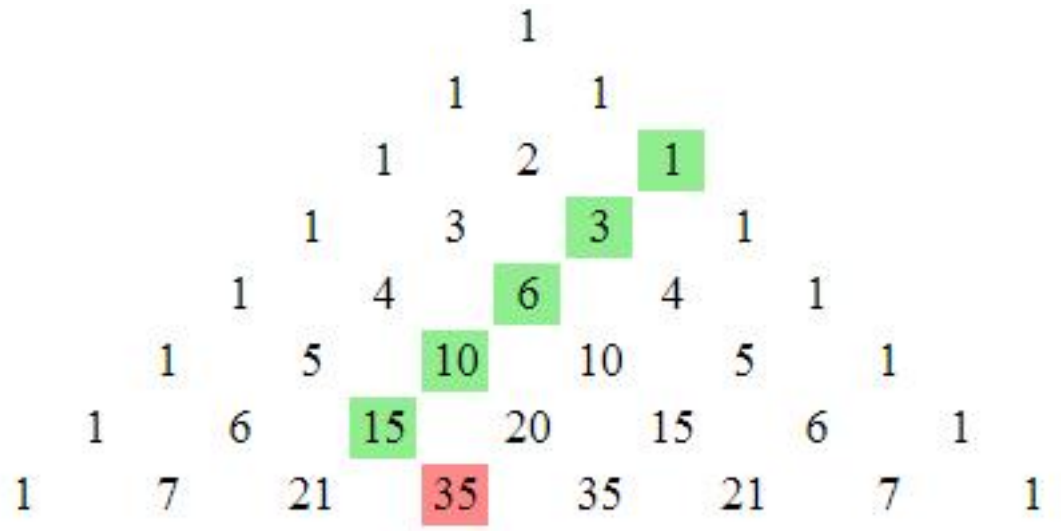
Consider a group of n students and 1 teacher. To form a group of size k from these people, there are two disjoint and exhaustive cases: 1) the teacher is in the group, 2) the teacher is not in the group.

If the teacher is in the group, then we can choose $k-1$ students from n total, being $\binom{n}{k-1}$. If the teacher is not in the group, then we can choose k students from n, so $\binom{n}{k}$. By the sum rule, the total number of ways to form a group of size k is $\binom{n}{k-1} + \binom{n}{k} = \binom{n+1}{k}$.

Hockey-Stick Identity

The hockey-stick identity is a variation of Pascal's Identity and another pattern in Pascal's Triangle. Suppose we fix some r. Then to find the $r+1$-th entry on the $n+1$-th row, we use the formula:

$$\sum_{i=r}^{n} \binom{i}{r} = \binom{r}{r} + \binom{r+1}{r} + \cdots + \binom{n}{r} = \binom{n+1}{r+1}$$



Consider $n+1$ distinct children lined up to receive $r+1$ identical candies so that exactly $r+1$ children receive one candy each (no child gets more than 1). This is essentially choosing a group of $r+1$ children where each person in that group gets a candy, so $\binom{n+1}{r+1}$.

Now consider a different method of counting the same scenario. Since the children are all distinct, suppose that their distinguishing factor is their heights. Then by the Well-Ordering Principle, when we choose a group of $r+1$ children, they form a non-empty set and there must be one shortest child. Let this shortest child receive the last of the $r+1$ candies (distributed within the group from tallest to shortest).

Then fixing this shortest child, there are $\binom{n}{r}$ ways to distribute the remaining r candies to the other taller children. If we fix a different child to be the shortest in the group, then the group must be different and excludes the previous child, so the number of ways to distribute the candies to the remaining children becomes $\binom{n-1}{r-1}$, and so on until we have excluded $n-(r+1)$ shortest children and have $r+1$ children left for $r+1$ candies (being $\binom{r}{r}$). By the sum rule, the total number of ways to distribute $r+1$ candies to $n+1$ children is their sum, matching the LHS of the equation.

Vandermonde's Identity

Suppose that there are a total of $m+n$ people at a company: m developers and n designers. To form a team of size r, you may choose any number of developers and designers.

- *Note:* Here, even though there are two "types" of objects, within each type objects are still distinguishable from one another. For example, two developers likely have different names from one another.

Vandermonde's Identity states:

$$\binom{m+n}{r} = \sum_{k=0}^{r} \binom{m}{k}\binom{n}{r-k}$$

The LHS offers a straightforward count of the ways to form a group of size r from $m+n$ people: disregard the type and just choose r.

The RHS is a different way of counting. Suppose that you predetermine the number of developers on the team to be k. Then there must be $r-k$ designers. So the total ways to form a team with k developers is $\binom{m}{k} \times \binom{n}{r-k}$ by the product rule (i.e., first choose k developers, and choose $r-k$ designers).

Iterating through all possible number of developers, by the sum rule, we have $\sum_{k=0}^{r} \binom{m}{k}\binom{n}{r-k}$, which matches the RHS.

9.4 The Pigeonhole Principle

Statement: If there are n pigeons and k holes with $n > k$, then there exists a hole with more than 1 pigeon in it (or, there exists a hole with at least 2 pigeons).

More formally: Let A and B be finite sets with $|A| = n$ and $|B| = m$. If $n > m$, then for any function $f : A \to B$, there exists at least two distinct elements $a_1, a_2 \in A$ such that $f(a_1) = f(a_2)$.

- In other words, no function from a larger set to a smaller set can be injective.

The Existence/Threshold Problem

By applying the Pigeonhole Principle, we can prove that a "collision" or duplicate of sorts occurs. For example, we can prove that in any set of $n + 1$ integers, there must exist two whose difference is divisible by n.

It is also best to think about the Pigeonhold Principle adversarially. For example, consider a 3x3 pigeon box, able to contain 9 pigeons total (1 in each hole). If the target is to show that there exists a hole with at least 2 pigeons in it, then the "worst" way to place the pigeons to avoid having a hole with 2 pigeons in it is to place one in each hole 9 times. The tipping point is then 10 pigeons.

Example

Consider an 8x8 chessboard filled with some number of k kings, all wishing to attack each other. What is the maximum k before at least 1 king is able to attack another in a single move?

Generalized Pigeonhole Principle

The Generalized Pigeonhole Principle states that if n items (pigeons) are placed into k containers (holes), there must be at least one container holding at least $\lceil n/k \rceil$ items. This principle does not provide a constructive method for finding the arrangement but guarantees the existence of such a bound.

It is entirely plausible for why we take the ceiling. Consider the most adversarial placement of 9 pigeons, one in each hole, in a 3x3 box. Then $n = 9$ and $k = 9$ and we guarantee that at least 1 container holds at least 1 pigeon. In fact, even if there were fewer pigeons, as long as it is nonzero there will be at least 1 container with at least 1 pigeon.

The Magnitude Problem

Using the Generalized Pigeonhold Principle, we can go further to claim that there exists a container with $\geq k$ objects.

If you pick out 100 integers (need not be distinct) from the set {1,2,3,...,1000}, what is the smallest value of k such that you can be sure that at least k out of the 100 integers have the same remainder when divided by 7?

This principle has applications in Group Theory. A group is defined by properties such as Identity, Associativity, and the existence of an Inverse. For example, the movements of a Rubik's cube form a finite group. We can prove that for any element a in a finite group G, there exists an integer $k \geq 1$ such that $a^k = e$ (the identity). By listing the sequence $a^1, a^2, \ldots, a^{n+1}$ where $n = |G|$, the Pigeonhole Principle guarantees that two distinct powers must be equal (e.g., $a^x = a^y$), implying $a^{x-y} = e$.

Algorithmic Application: Coin Weighing

Combinatorics helps analyze algorithmic complexity. In the problem of finding a single heavy coin among n coins using a balance scale, the optimal algorithm splits the coins into three even groups. Because each weighing provides three possible outcomes (less than, equal, greater than), the search space is reduced by a factor of 3 at each step. The Generalized Pigeonhole Principle can be used to prove that the minimum number of weighings required is $\lceil \log_3 n \rceil$.

9.5 Probabilistic Method

The Probabilistic Method is a non-constructive proof technique that demonstrates the existence of a mathematical object by showing that a randomly chosen object from an appropriate probability space has a positive probability of satisfying desired properties. Rather than explicitly constructing an object with certain properties, this method proves existence by showing that the probability of such an object existing is greater than zero.

The fundamental principle states: if $\Pr[\text{object exists}] > 0$, then the object must exist. This approach leverages the linearity of expectation and probability theory to prove existence results that might be difficult or impossible to establish constructively.

9.6 Linearity of Expectation

For any random variable X, the expected value is calculated as $E[X] = \sum x \cdot \Pr[X = x]$, representing the weighted sum of all possible observed values multiplied by their respective probabilities. A crucial property of expectation is its linearity: if X and Y are random variables, then $E[X + Y] = E[X] + E[Y]$. This linearity holds regardless of whether the random variables are independent, making it an exceptionally powerful tool in probabilistic arguments.

The key insight for applying the Probabilistic Method is recognizing that if $E[X] \geq a$, then there must exist some outcome where $X \geq a$. Similarly, if $E[X] \leq b$, there exists an outcome where $X \leq b$.

Application: Largest Bipartite Subgraph

Any graph G with $|E|$ edges contains a bipartite subgraph with at least $\frac{|E|}{2}$ edges. This result illustrates the Probabilistic Method by using random coloring rather than explicit construction.

The proof proceeds by randomly coloring each vertex either White or Black with probability $\frac{1}{2}$ independently. Define E' as the set of edges whose endpoints have opposite colors, and for each edge $e \in E$, let A_e be an indicator random variable where $A_e = 1$ if edge e has endpoints of opposite colors and $A_e = 0$ otherwise. Among the four possible colorings for any edge (WW, WB, BW, BB), exactly two result in opposite colors, so $\Pr[A_e = 1] = \frac{1}{2}$.

Applying linearity of expectation: $E[|E'|] = E[\sum_{e \in E} A_e] = \sum_{e \in E} E[A_e] = \sum_{e \in E} \frac{1}{2} = \frac{|E|}{2}$. Since the expected number of edges with opposite-colored endpoints is $\frac{|E|}{2}$, there must exist at least one coloring that achieves this value or greater (probability of it occurring is strictly greater than 0), guaranteeing a bipartite subgraph of the desired size.

9.7 Example: Ramsey Theory

Ramsey Theory captures the principle that within any sufficiently large chaotic or random system, there must exist a certain amount of order or structure. This concept formalizes the idea that complete disorder is impossible at scale.

A classic example demonstrates that among any 6 people, there must exist either 3 who all know each other (forming a red triangle in graph terminology) or 3 who are all strangers to each other (forming a blue triangle). This can be modeled using a complete graph K_n, where every vertex is connected to all other vertices, with edges colored to represent relationships.

Ramsey Number Lower Bound Proof

This set of notes covers the probabilistic proof for the lower bound of the Ramsey Number $R(t, t)$. The goal is to show that $R(t, t) > n$ for some n, which implies there exists a graph of size n with no monochromatic clique of size t. We use the Probabilistic Method to demonstrate that such a coloring exists by showing the probability of a "bad" event (existence of a monochromatic clique) is strictly less than 1.

Random Coloring Setup

Consider a complete graph K_n on n vertices. We color every edge of K_n either red or blue uniformly at random. This means that for any edge e, $\Pr[e \text{ is red}] = \Pr[e \text{ is blue}] = \frac{1}{2}$, and the color choices for all edges are independent. We focus on finding the probability that this random coloring produces a monochromatic K_t (a clique of size t where all edges are the same color).

Probability for a Single Set

Let S be a subset of vertices with size t (i.e., $|S| = t$). We define the event A_S as the event that the subgraph induced by S is monochromatic (either all red or all blue). The total number of edges in S is $\binom{t}{2}$. For S to be all red, every one of these edges must be chosen as red.

$$\Pr[S \text{ is red}] = \left(\frac{1}{2}\right)^{\binom{t}{2}} = 2^{-\binom{t}{2}}$$

Since S can be either all red or all blue, and these are mutually exclusive events (assuming $t \geq 2$), we sum the probabilities:

$$\Pr[A_S] = \Pr[S \text{ is red}] + \Pr[S \text{ is blue}] = 2^{-\binom{t}{2}} + 2^{-\binom{t}{2}} = 2 \cdot 2^{-\binom{t}{2}} = 2^{1-\binom{t}{2}}$$

Union Bound Application

We now consider the probability that *any* subset of size t forms a monochromatic clique. Let E be the event that K_n contains a red K_t or a blue K_t. This event is the union of the events A_S over all possible subsets S of size t. By the Union Bound (Boole's Inequality), the probability of a union of events is at most the sum of their individual probabilities:

$$\Pr[E] = \Pr\left[\bigcup_{|S|=t} A_S\right] \leq \sum_{|S|=t} \Pr[A_S]$$

Since there are $\binom{n}{t}$ such subsets, and each has the same probability $2^{1-\binom{t}{2}}$, the total probability is bounded by:

$$\Pr[E] \leq \binom{n}{t} 2^{1-\binom{t}{2}}$$

Establishing the Bound

For the probabilistic method to work, we need to show that $\Pr[E] < 1$. If the probability that a "bad" coloring occurs is strictly less than 1, then the probability that a "good" coloring exists (one with no monochromatic K_t) is strictly greater than 0.

$$\binom{n}{t} 2^{1-\binom{t}{2}} < 1$$

Using the approximation $\binom{n}{t} \leq \frac{n^t}{t!}$ and simplifying the exponent term $2^{1-\binom{t}{2}} = 2^{1-\frac{t(t-1)}{2}}$, we can analyze the condition.

- The board notes simplify this condition to derive a specific bound for n.
- Specifically, if we pick $n = 2^{t/2}$, we can verify if the inequality holds.
- Through substitution and approximation (detailed in classical proofs), it can be shown that if $n \leq 2^{t/2}$, the probability sum is strictly less than 1.

Conclusion

Since $\Pr[\text{no monochromatic } K_t] > 0$ when $n \leq 2^{t/2}$, there must exist at least one coloring of K_n that has no monochromatic clique of size t. By the definition of Ramsey numbers, $R(t,t)$ is the minimum size where a monochromatic clique is *guaranteed*. Since we found a graph of size $n = 2^{t/2}$ where it is *not* guaranteed (in fact, we showed one exists without it), the Ramsey number must be larger than this n.

$$R(t,t) > 2^{t/2}$$