Appendix A

Mathematical Background

This appendix summarizes aspects of language and mathematics that are not directly part of graph theory but provide useful background for learning graph theory. Where appropriate, we mention examples in the context of graphs, so it is best to read this appendix in conjunction with Chapter 1. This presentation is modeled on material in the first half of *Mathematical Thinking*, by John P. D'Angelo and Douglas B. West (Prentice–Hall, second edition, 2000).

SETS

Our most primitive mathematical notion is that of a **set**. It is so fundamental that we cannot define it in terms of simpler concepts. We think of a set as a collection of distinct objects with a precise description that provides a way of deciding (in principle) whether a given object is in it.

A.1. Definition. The objects in a set are its **elements** or **members**. When x is an element of A, we write $x \in A$ and say "x **belongs to** A". When x is not in A, we write $x \notin A$. If every element of a set B belongs to A, then B is a **subset** of A, and A **contains** B; we write $B \subseteq A$ or $A \supseteq B$.

For example, we may speak of the set A of graphs with n vertices. When we impose an additional restriction, such as requiring that the graphs also be connected, we obtain a subset of A.

When we list the elements of a set explicitly, we put braces around the list; "$A = \{-1, 1\}$" specifies the set A consisting of the elements -1 and 1. Writing the elements in a different order does not change a set. We write $x, y \in S$ to mean that both x and y are elements of S.

A.2. Example. We use the characters $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, and $\mathbb{R}$ to name the sets of **natural numbers, integers, rational numbers**, and **real numbers**, respectively. Each set in this list is contained in the next, so we write $\mathbb{N} \subseteq \mathbb{Z} \subseteq \mathbb{Q} \subseteq \mathbb{R}$.

We treat these sets and their elements as familiar objects. By convention, 0 is not a natural number, so $\mathbb{N} = \{1, 2, 3, \ldots\}$. The set of integers is $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$. The set $\mathbb{Q}$ of rational numbers is the set of real numbers expressible as a/b with $a, b \in \mathbb{Z}$ and $b \neq 0$.

We also take as familiar the elementary arithmetic properties of these number systems. These include the rules that permit algebraic manipulation of expressions, equalities, and inequalities. They also include elementary properties about divisibility of integers. ∎

A.3. Definition. Sets A and B are **equal**, written $A = B$, if they have the same elements. The **empty set**, written $\emptyset$, is the unique set with no elements. A **proper subset** of a set A is a subset that is not A itself.

The empty set is a subset of every set, and every set is a subset of itself. The definition of subgraph (Definition 1.1.16) is similar. Every graph is a subgraph of itself, but something must be discarded to obtain a proper subgraph.

"Solving a mathematical problem" often means describing a given set more simply. We must show that the set of objects satisfying the new description is equal to the given set.

A.4. Remark. *Equality of sets.* To prove that $A = B$, we prove that every element of A is in B and that every element of B is in A; in other words, $A \subseteq B$ and $B \subseteq A$. It also suffices to turn the description of one set into the description of the other by operations that do not change membership.

This book proves many characterization theorems for classes of graphs. Such a theorem states that two sets are the same (example: the set of bipartite graphs is equal to the set of graphs without odd cycles—Theorem 1.2.18).

Often, a mathematical model defines a set S of *solutions*; these are the objects that satisfy the conditions of the problem. We want to list or describe the solutions explicitly; this specifies a set T. The problem is to show that $S = T$. Proving $S \subseteq T$ means showing that every solution belongs to T. Proving $T \subseteq S$ means showing that every member of T is a solution. ∎

A.5. Remark. *Specifying a set.* Given a set A, we may want to specify a subset S consisting of the elements of A that satisfy a given condition. To do so, we write "$S = \{x \in A$: condition$(x)\}$". We read this as "S is the set of elements x in A such that x satisfies 'condition' ". For example, the expression $\{n \in \mathbb{N} : n^2 \leq 25\}$ is another way to name the set $\{1, 2, 3, 4, 5\}$.

In this format, the set A is the **universe** for x; we can drop this part of the notation when the context makes it clear. For example, $\{n^2 \colon n \in \mathbb{N}\}$ is the set of positive integers squares. ∎

Many special sets have common names and/or notation.

A.6. Definition. When $a, b \in \mathbb{Z}$, we write $\{a, \ldots, b\}$ for $\{i \in \mathbb{Z} \colon a \leq i \leq b\}$. When $n \in \mathbb{N}$, we write $[n]$ for $\{1, \ldots, n\}$; also $[0] = \emptyset$. The set of **even numbers**

is $\{2k: k \in \mathbb{Z}\}$. The set of **odd numbers** is $\{2k+1: k \in \mathbb{Z}\}$. The **parity** of an integer states whether it is even or odd.

Note that 0 is an even number. We say "even" and "odd" for numbers *only* when discussing integers. Every integer is even or odd; none is both.

A.7. Definition. A **partition** of a set A is a list $A_1, \ldots, A_k$ of subsets of A such that each element of A appears in exactly one subset in the list.

The set of even numbers and the set of odd numbers partition $\mathbb{Z}$. In a partition of A into $A_1, \ldots, A_k$, the sets $A_1, \ldots, A_k$ in the list are called "blocks" or "classes" or "parts" or "partite sets". The use of "blocks" is common in combinatorics, but graph theory has another definition for the word "block", so we usually use "classes" or "sets". "Partite sets" is used only for the sets in a partition of the vertex set of a graph into independent sets.

A.8. Remark. *Conventions about universes.* When we write "$[n]$", it is understood that n is a nonnegative integer. When we speak of n as the number of vertices in a graph, by context we know that n is a natural number. When we say only that a number is positive without specifying the number system containing it, we mean that it is a positive real number. Thus, "consider $x > 0$" means "let x be a positive real number", but in "For $n \geq 2$, let G be a n-vertex graph" our convention is that $n \in \mathbb{N}$. ∎

A.9. Definition. A set A is **finite** if there is a one-to-one correspondence between A and $[n]$ for some $n \in \mathbb{N} \cup \{0\}$. This n is the **size** of A, written $|A|$.

Another elementary property of number systems is that a set A cannot be in one-to-one correspondence with both $[m]$ and $[n]$ when $m \neq n$. Thus the size of a finite set is a well-defined integer. **Counting** a set means determining its size.

A.10. Remark. *"If" in definitions.* It is a common convention in definining mathematical properties to say that an object has a certain property **if** it satisfies a certain condition. Subsequently, the condition can be substituted for the property and vice versa, so the "if" really means "if and only if". This conventional usage in definitions reflects the notion that the concept being defined does not exist until the definition is complete. ∎

There are several natural ways to obtain new sets from old sets.

A.11. Definition. Let A and B be sets. Their **union** $A \cup B$ consists of all elements in A or in B (or both). Their **intersection** $A \cap B$ consists of all elements in both A and B. Their **difference** $A - B$ consists of the elements of A that are not in B. Their **symmetric difference** $A \triangle B$ is the set of elements belonging to exactly one of A and B.

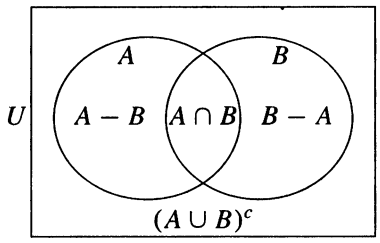
Two sets are **disjoint** if their intersection is the empty set $\emptyset$. If a set A

is contained in some universe U under discussion, then the **complement** $\overline{A}$ of A is the set of elements of U *not* in A.

When we speak of taking the "complement" of a simple graph, we are keeping the vertex set unchanged and taking the complement of the edge set (viewed as pairs of vertices) within the universe of vertex pairs. Other times we speak of the complement $\overline{S}$ of a set of vertices S in G; in this case we mean $\overline{S} = V(G) - S$.

A.12. Remark. In a **Venn diagram**, an outer box represents the universe under consideration, and regions within the box correspond to sets. Non-overlapping regions correspond to disjoint sets. The four regions in the Venn diagram for two sets A and B represent $A \cap B$, $\overline{(A \cup B)}$, $A - B$, and $B - A$. Note that $A \triangle B = (A - B) \cup (B - A)$,

Since $A - B$ consists of the elements in A and not in B, we have $A - B = A \cap \overline{B}$. Similarly, the diagram suggests that $\overline{B}$ is the union of $A - B$ and $\overline{(A \cup B)}$, which are disjoint. It also suggests that the symmetric difference $A \triangle B$ is obtained from the union by deleting the intersection. ∎



A.13. Remark. *When A and B are sets,* $A \triangle B = (A \cup B) - (A \cap B)$. The union starts with all elements in at least one of A and B; we delete those in both.

When A and B are finite sets, $|A \cup B| + |A \cap B| = |A| + |B|$. Each element of the intersection is counted twice on both sides, each element of the symmetric difference is counted once on both sides, and no other elements are counted. ∎

A.14. Definition. A **list** with entries in A consists of elements of A in a specified order, with repetition allowed. A **k-tuple** is a list with k entries. We write A^k for the set of k-tuples with entries in A. When $A = \{0, 1\}$, A^k is the set of **binary k-tuples**.

An **ordered pair** (x, y) is a list with two entries. The **cartesian product** of sets S and T, written $S \times T$, is the set $\{(x, y): x \in S, y \in T\}$.

Note that $A^2 = A \times A$ and $A^k = \{(x_1, \ldots, x_k): x_i \in A\}$. We read "$x_i$" as "$x$ sub i". When $S = T = \mathbb{Z}$, the cartesian product $S \times T$ is the **integer lattice**, the set of points in the plane with integer coordinates.

QUANTIFIERS AND PROOFS

Roughly speaking, a mathematical statement is a statement that can be determined to be true or false. This requires correct mathematical grammar, and it requires that variables be "quantified".

For example, the sentence $x^2 - 4 = 0$ cannot be determined to be true or false because we do not know the value of x. It becomes a mathematical statement if we precede it with "When $x = 3$," or "For $x \in \{2, -2\}$," or "For some integer x,".

If a sentence $P(x)$ becomes a mathematical statement whenever the variable x takes a value in the set S, then the two sentences below are mathematical statements.

"For all x in S, the sentence $P(x)$ is true."
"For some x in S, the sentence $P(x)$ is true."

A.15. Definition. In the statement "For all x in S, $P(x)$ is true", the variable x is **universally quantified**. We write this as $(\forall x \in S)P(x)$ and say that $\forall$ is a **universal quantifier**. In "For some x in S, $P(x)$ is true", the variable x is **existentially quantified**. We write this as $(\exists x \in S)P(x)$ and say that $\exists$ is an **existential quantifier**. The set of allowed values for a variable is its **universe**.

A.16. Remark. *English words that express quantification.* Typically, "every" and "for all" represent universal quantifiers, while "some" and "there is" represent existential quantifiers. We can also express universal quantification by referring to an arbitrary element of the universe, as in "Let x be an integer", or "A student failing the exam will fail the course". Below we list common indicators of quantification.

Universal ($\forall$)	(helpers)	Existential ($\exists$)	(helpers)
for [all], for every		for some	
if	then	there exists	such that
whenever, for, given		at least one	for which
every, any	satisfies	some	satisfies
a, arbitrary	must, is	has a	such that
let	be		

The "helpers" may be absent. Consider "The square of a real number is nonnegative". This means $x^2 \geq 0$ for *every* $x \in \mathbb{R}$; it is not a statement about one real number and cannot be verified by an example. When we write "A bipartite graph has no odd cycle", we mean "in every bipartite graph there is no odd cycle". When we write "Let G be a bipartite graph", we mean that every bipartite graph is under consideration. When we take an "arbitrary" vertex in a graph, we are considering each one individually. When we discus an "arbitrary" pair of vertices in a graph, we are considering each pair, one at a time.

The difference between "for every G" and "for every graph G" is that the latter specifies the universe for the universally quantified variable G. ∎

Existential quantifiers state lower bounds; "there is a" and "there are two" mean "at least one" and "at least two". Phrases like "there is a unique" and "there are exactly two" indicate equality. Sometimes equality is clear from context, but it does not hurt to make it explicit when it is intended.

A statement may have more than one quantifier. Consider the sentence "There are triangle-free graphs with arbitrarily large chromatic number". Phrased using explicit quantifiers, this means "For every $n \in \mathbb{N}$, there exists a triangle-free graph with chromatic number at least n". The expression "arbitrarily large" often conveys an implicit universal quantifier in this way.

In contrast, the expression "sufficiently large" imposes an implicit existential quantifier. The statement "$2^n > n^{1000}$ when n is sufficiently large" means "There exists $N \in \mathbb{N}$ such that for all $n \geq N$, the inequality $2^n > n^{1000}$ holds".

A.17. Remark. The meaning of a statement with more than one quantifier depends on their order. Compare these two sentences:

"For every graph G, there exists $m \in \mathbb{N}$ such that every $v \in V(G)$ has degree at most m"
"There exists $m \in \mathbb{N}$ such that for every graph G, every $v \in V(G)$ has degree at most m"

The first statement is true; the second is false. Every (finite) graph has a maximum degree, but there is no maximum over all graphs. We write the two sentences in logical notation as

$$(\forall G)(\exists m \in \mathbb{N})(\forall v \in V(G))(d_G(v) \leq m).$$
$$(\exists m \in \mathbb{N})(\forall G)(\forall v \in V(G))(d_G(v) \leq m).$$

In English, quantifiers often appear at the ends of sentence to enhance readability, as in "I feel happy every time I learn something new." In sentences with abstract concepts and more than one quantifier, we adopt conventions about order to avoid confusion. Quantifiers apply in the order in which they are stated. In particular, a variable is chosen in terms of the preceding variables.

For example, in $(\forall G)(\exists m \in \mathbb{N})P(G, m)$, we have the freedom to choose m after knowing what G is. In $(\exists m \in \mathbb{N})(\forall G)P(G, m)$, we must choose a single m that works for all G. ■

A.18. Remark. *Negation of quantified statements.* The logical symbol for negation is $\neg$. If it is false that all $x \in S$ make $P(x)$ true, then there must be some $x \in S$ such that $P(x)$ is false. Similarly, negating an existentially quantified statement yields a universally quantified negation. In notation,

$$\neg[(\forall x \in S)P(x)] \text{ has the same meaning as } (\exists x \in S)(\neg P(x)).$$
$$\neg[(\exists x \in S)P(x)] \text{ has the same meaning as } (\forall x \in S)(\neg P(x)).$$

The universe of quantification *does not change* when the statement is negated.

For example, the false statement in Remark A.17 was

$$(\exists m \in \mathbb{N})(\forall G)(\forall v \in V(G))(d_G(v) \leq m).$$

Its negation is the same as $(\forall m \in \mathbb{N})(\exists G)[\neg((\forall v \in V(G))(d_G(v) \leq m))]$, which we further simplify to $(\forall m \in \mathbb{N})(\exists G)(\exists v \in V(G))(d_G(v) > m)$. This statement is

"for every natural number m, there is some graph having a vertex with degree greater than m", which is true. ∎

Logical connectives permit us to build compound statements.

A.19. Definition. *Logical connectives.* In the following table, we define the operations named in the first column by the truth values specified in the last column.

Name	Symbol	Meaning	Condition for truth
Negation	$\neg P$	not P	P false
Conjunction	$P \wedge Q$	P and Q	both true
Disjunction	$P \vee Q$	P or Q	at least one true
Biconditional	$P \Leftrightarrow Q$	P if & only if Q	same truth value
Conditional	$P \Rightarrow Q$	P implies Q	Q true whenever P true

A.20. Remark. Conjunction and disjunction are quantifiers over the truth of their component statements. A conjunction ("and") is true precisely when all of its component statements are true. A disjunction ("or") is true precisely when there exists a true statement among its components. Our understanding of negation thus yields logical equivalence between $\neg(P \wedge Q)$ and $(\neg P) \vee (\neg Q)$ and between $\neg(P \vee Q)$ and $(\neg P) \wedge (\neg Q)$. ∎

A.21. Definition. In the conditional statement $P \Rightarrow Q$, we call P the **hypothesis** and Q the **conclusion**. The statement $Q \Rightarrow P$ is the **converse** of $P \Rightarrow Q$.

A.22. Remark. *Conditionals.* Conditional statements are the only type in Definition A.19 whose meaning changes when P and Q are interchanged. There is no general implication between $P \Rightarrow Q$ and its converse $Q \Rightarrow P$. Consider these three statements about a graph G: P is "G is a path", Q is "G is bipartite", and R is "G has no odd cycles". Here $P \Rightarrow Q$ is true but $Q \Rightarrow P$ is false. On the other hand, both $Q \Rightarrow R$ and $R \Rightarrow Q$ are true.

Note that here G is a variable. We have dropped G from the notation for the statements because the context is clear. The precise meaning of $P \Rightarrow Q$ using G is $(\forall G)(P(G) \Rightarrow Q(G))$.

A conditional statement is false when and only when the hypothesis is true and the conclusion is false. Thus the meaning of $P \Rightarrow Q$ is $(\neg P) \vee Q$; the two are logically equivalent. Every conditional statement with a false hypothesis is true, regardless of whether the conclusion is true. The meaning of $\neg(P \Rightarrow Q)$ is $P \wedge (\neg Q)$.

Below we list ways to say $P \Rightarrow Q$ in English. ∎

If P (is true), then Q (is true).	P is true only if Q is true.
Q is true whenever P is true.	P is a sufficient condition for Q.
Q is true if P is true.	Q is a necessary condition for P.

The business of mathematics is proving implications. Note that universally quantified statements can be interpreted as conditional statements. The statement "$(\forall G \in \mathbf{G})(P(G))$" has the same meaning as "If $G \in \mathbf{G}$, then $P(G)$" (consider the two statements when $\mathbf{G}$ is the family of bipartite graphs and $P(G)$ is the assertion that G has no odd cycles).

The basic proof methods come from the meaning of conditional statements.

A.23. Remark. *Proving implications.* The **direct method** of proving $P \Rightarrow Q$ is to assume that P is true and then to apply mathematical reasoning to deduce that Q is true. When P is "$x \in A$" and Q is "$Q(x)$", the direct method considers an *arbitrary* $x \in A$ and deduces $Q(x)$. There is no "proof by example". The proof must apply to every member of A as a possible instance of x.

The **contrapositive** of $P \Rightarrow Q$ is $\neg Q \Rightarrow \neg P$. Each of these statements fails only when P is true and Q is false. Thus they are equivalent; we can prove $P \Rightarrow Q$ by proving $\neg Q \Rightarrow \neg P$. This is the **contrapositive method**.

We have observed that $(P \Rightarrow Q) \Leftrightarrow \neg [P \wedge (\neg Q)]$. Hence we can prove $P \Rightarrow Q$ by proving that P and $\neg Q$ cannot both be true. We do this by obtaining a contradiction after assuming both P and $\neg Q$. This is the **method of contradiction**.

The two latter methods are **indirect proof**. When the direct method for $P \Rightarrow Q$ doesn't seem to work, we say "Well, suppose not". At that point we are starting from the assumption $\neg Q$. We need not know in advance whether we are seeking to derive $\neg P$ (contrapositive method) or seeking to use P and $\neg Q$ to obtain a contradiction. ∎

Examples of each of these methods appear in the text. Indirect proof is promising when the negation of the conclusion provides useful information. This approach may be easier than finding a direct proof, because both the hypothesis and the negation of the conclusion can be used. If the contradiction we obtain is the impossibility of our original assumption $\neg Q$, then usually we can rewrite the proof in simpler language as a direct proof. If instead we obtain $\neg P$, then we have proved the contrapositive.

A.24. Remark. *Biconditional statements.* The biconditional statement "$P \Leftrightarrow Q$" has the same meaning as "$(P \Rightarrow Q) \wedge (Q \Rightarrow P)$". We read it as "$P$ if and only if Q", where "$Q \Rightarrow P$" is "P if Q", and "$P \Rightarrow Q$" is "P only if Q".

Although sometimes we can prove a biconditional statement by a chain of equivalences, usually we prove a conditional statement and its converse; the latter is also a conditional statement. For each we have the three fundamental methods above. To prove $P \Leftrightarrow Q$, we must prove one statement in each column in the table below. The lines are the direct method, the contrapositive method, and the method of contradiction, respectively. Proving two statements in the same column would amount to proving the same statement twice. ∎

$P \Rightarrow Q$	$Q \Rightarrow P$
$\neg Q \Rightarrow \neg P$	$\neg P \Rightarrow \neg Q$
$\neg(P \wedge \neg Q)$	$\neg(Q \wedge \neg P)$

Students sometimes wonder about the precise meanings of words like "theorem", "lemma", and "corollary" that are used to designate mathematical results. In Greek, *lemma* means "premise" and *theorema* means "thesis to be proved". Thus a theorem is a major result requiring some effort. A lemma is a lesser statement, usually proved in order to help prove other statements. A proposition is something "proposed" to be proved; typically this takes less effort than a theorem. The word *corollary* comes from Latin, as a modification of a word meaning "gift"; a corollary follows easily from a theorem or proposition, without much additional work.

INDUCTION AND RECURRENCE

Many statements having a natural number as a variable can be proved using the technique of induction. In Theorem 1.2.1, we describe the strong version of induction. Here we review the ordinary version that most students learn when they first encounter induction. It involves the Well Ordering Property for the natural numbers, which states that every nonempty subset of $\mathbb{N}$ has a least element. We take this as an axiom, as part of our intuitive understanding of what $\mathbb{N}$ is. Although we then state the Principle of Induction as a Theorem, in reality it is equivalent to the Well Ordering Property for $\mathbb{N}$.

A.25. Theorem. (Principle of Induction) For each natural number n, let $P(n)$ be a mathematical statement. If properties (a) and (b) below hold, then for each $n \in \mathbb{N}$ the statement $P(n)$ is true.
a) $P(1)$ is true.
b) For $k \in \mathbb{N}$, if $P(k)$ is true, then $P(k+1)$ is true.

Proof: If $P(n)$ is not true for all n, then the set of natural numbers where it fails is nonempty. By the Well Ordering Property, there is a least natural number in this set. By (a), this number cannot be 1. By (b), it cannot be bigger than 1. The contradiction implies that $P(n)$ is true for all n. ∎

When applying the method of induction, we prove statement (a) in Theorem A.25 as the **basis step** and statement (b) as the **induction step**. Statement (b) is a conditional statement, and its hypothesis ("$P(k)$" is true) is the **induction hypothesis**. We present one example in rather formal language.

A.26. Proposition. If S is a set of n lines in the plane such that every two have exactly one common point and no three have a common point, then S cuts the plane into $1 + n(n+1)/2$ regions.

Proof: We use induction on n to prove the claim for all $n \in \mathbb{N}$. Let $P(n)$ be the statement that the claim holds for all such sets of n lines.

Basis step ($P(1)$). With one line the number of regions is 2, which equals $1 + 1(1+1)/2$.

Induction step ($P(k) \Rightarrow P(k+1)$). The statement $P(k)$ is the induction hypothesis. Let S be a set of $k+1$ lines meeting the conditions. Select a line L

in S (the dashed line in the figure), and let S' be the set of k lines obtained by deleting L from S.

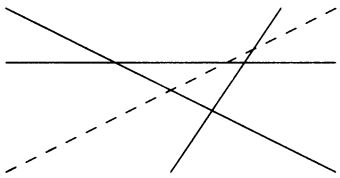
Since S' meets the conditions, the induction hypothesis states that S' cuts the plane into $1 + k(k+1)/2$ regions. When we replace L, some regions are cut. The increase in the number of regions is the number of regions that L cuts. It moves from one of these regions to another each time it crosses a line in S'. Since L crosses each line in S' once, the lines in S' cut L into $k+1$ pieces. Each piece corresponds to a region that L cuts.

Thus the number of regions formed by S is $k+1$ more than the number of regions formed by S'. The number of regions formed by S is

$$1 + k(k+1)/2 + (k+1) = 1 + (k+1)(k+2)/2.$$

We have proved that $P(k)$ implies $P(k+1)$.

By the principle of induction, the claim holds for every $n \in \mathbb{N}$. ∎



A.27. Remark. The discussion of Proposition A.26 suggests several comments about proof by induction. Note first that we could also have used $n = 0$ as the basis step to prove the statement for all nonnegative n.

It is not immediately obvious from the statement of the problem that the number of regions is the same for all sets of n lines, but this follows because we proved a formula for this number that depends only on n.

In the proof of the induction step, we began with L, an instance of the larger-sized problem. This approach ensures that we have considered all such instances; we return to this point shortly.

We proved $P(k+1)$ from $P(k)$ as suggested by statement (b) of Theorem A.25. In most examples in this book, we use a different phrasing that is more consistent with strong induction as introduced in Section 1.2. To prove $P(n)$ for all $n \in \mathbb{N}$, in this example we would write "Basis step: $n = 1$. . . ." and then "Induction step: $n > 1$. . . .". In the proof of the induction step, we would consider an arbitrary set S of n lines and apply the induction hypothesis to the set S' obtained by deleting one line L.

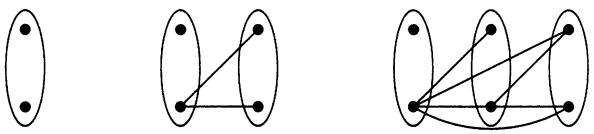
The content of the proof is the same in both phrasings. The phrasing that we have just described emphasizes the item about which the claim is proved. The basis step directly verifies the claim for the smallest value of the induction parameter. When the parameter has a larger value, the claim about the item is proved using the hypothesis that it holds for an earlier item; this is the induction step. Invoking it (repeatedly) yields the claim for each subsequent value of the parameter. ∎

When learning to use induction in graph theory, many students have trouble with two particular aspects. One is when the statement $P(n)$ being proved by induction is itself a conditional statement $A(n) \Rightarrow B(n)$. The induction hypothesis is the statement $A(n-1) \Rightarrow B(n-1)$. A template for the induction step in this situation is presented in Remark 1.3.25, and there are examples of this throughout Chapter 1.

The other pitfall we call the "induction trap", discussed at length in Example 1.3.26. Here we provide another example, using the language of proving $P(n+1)$ from $P(n)$ that sometimes leads students into the trap.

A.28. Example. *The Handshake Problem.* Let a **handshake party** of order n (henceforth "n-party") be a party with n married couples where no spouses shake hands with each other and the $2n-1$ people other than the host shake hands with different numbers of people. We use induction on n to prove that in every n-party, the hostess shakes with exactly $n-1$ people.

We model the party using a simple graph in which the vertices are the people at the party and the edges are the pairs who shake hands. The degree of a vertex is its number of handshaking partners. If no one shakes with his or her spouse, then each degree is between 0 and $2n-2$. The condition that the $2n-1$ numbers other than the host's are distinct implies that the degrees are 0 through $2n-2$. The figure below shows for $n \in \{1, 2, 3\}$ the graph that is forced; each circled pair of vertices indicates a married couple, with host and hostess rightmost in each graph.



Basis step: If $n = 1$, then the hostess shakes with 0 (which equals $n-1$), because the host and hostess don't shake.

Induction step (**INVALID**): The induction hypothesis is that the claim holds for n-parties. Consider such a party. By the induction hypothesis, the degree of the hostess is $n-1$. By our earlier discussion, the degrees of vertices other than the host are $0, \ldots, 2n-2$. We form an $(n+1)$-party by adding one more couple. Let one member of the new couple shake with everyone in the first n couples; the other shakes with no one. This increases the degree of each of the earlier vertices by 1, so those degrees other than the host are now $1, \ldots, 2n-1$, and the new couple have degrees 0 and $2n$. Hence the larger configuration is an $(n+1)$-party. The degree of the hostess has increased by 1, so it is n.

Induction step (**VALID**): The induction hypothesis is that the claim holds for n-parties. Consider an $(n+1)$-party. By our earlier discussion, the degrees other than the host are $0, \ldots, 2n$. Let p_i denote the person of degree i among these. Since p_{2n} shakes with all but one person, the person p_0 who shakes with no one must be the only person missed by p_{2n}. Hence p_0 is the spouse of

p_{2n}. Furthermore, this married couple $S = \{p_0, p_{2n}\}$ is not the host and hostess, since the host is not in $\{p_0, \ldots, p_{2n}\}$.

Everyone not in S shakes with exactly one person in S, namely p_{2n}. If we delete S to obtain a smaller party, then we have n couples remaining (including the host and hostess), no person shakes with a spouse, and each person shakes with one fewer person than in the full party. Hence in the smaller party the people other than the host shake hands with different numbers of people.

By deleting the set S, we thus obtain an n-party (deleting the leftmost couple in the picture for $n = 3$ yields the picture for $n = 2$). Applying the induction hypothesis to this n-party tells us that, outside of the couple S, the hostess shakes with $n - 1$ people. Since she also shakes with $p_{2n} \in S$, in the full $(n + 1)$-party she shakes with n people. ∎

The first argument in Example A.28 falls into the induction trap, because it does not consider all possible $(n + 1)$-parties. It considers only those obtained by adding a couple to an n-party in a certain way, without proving that every $(n + 1)$-party is obtained in this way.

Starting with an arbitrary $(n + 1)$-party forces us to prove that every $(n + 1)$-party arises in this way in order to obtain a configuration where we can apply the induction hypothesis. We cannot discard just any married couple to obtain the smaller party. We must find a couple S such that everyone outside S shakes with exactly one person in S. Only then will the smaller party satisfy the hypotheses needed to be an n-party.

The need to show that our smaller object satisfies the conditions in the induction hypothesis replaces the need to prove that all objects of the larger size were generated by growing from an object of the smaller size.

Sometimes the proof of the induction step uses more than one earlier instance. If we always use both $P(n - 2)$ and $P(n - 1)$ to prove $P(n)$, then we must verify both $P(1)$ and $P(2)$ to get started. The proof of the induction step is not valid for $n = 2$, since there is no $P(0)$ to use.

A.29. Example. Let $a_1, a_2, \ldots$ be defined by $a_1 = 2$, $a_2 = 8$, and $a_n = 4(a_{n-1} - a_{n-2})$ for $n \geq 3$. We seek a formula for a_n in terms of n.

We may try to guess a formula that fits the data. The definition yields $a_3 = 24$, $a_4 = 64$, and $a_5 = 160$. All these satisfy $a_n = n2^n$. Having guessed this as a possible formula for a_n, we can try to use induction to prove it.

When $n = 1$, we have $a_1 = 2 = 1 \cdot 2^1$. When $n = 2$, we have $a_2 = 8 = 2 \cdot 2^2$. In both cases, the formula is correct.

In the induction step, we prove that the desired formula is correct for $n \geq 3$. We use the hypothesis that the formula is correct for the preceding instances $n - 1$ and $n - 2$. This allows us to compute a_n using its expression in terms of earlier values:

$$a_n = 4(a_{n-1} - a_{n-2}) = 4[(n - 1)2^{n-1} - (n - 2)2^{n-2}] = (2n - 2)2^n - (n - 2)2^n = n2^n.$$

The validity of the formula for a_n follows from its validity for a_{n-1} and a_{n-2}, which completes the proof. ∎

In this proof, we must verify the formula for $n = 1$ and $n = 2$ in the basis step; the proof of the induction step is not valid when $n = 2$. Example A.29 specifies $a_1, a_2, \ldots$ by a **recurrence relation**. The general term a_n is specified using earlier terms. Similarly, the proof of Proposition A.26 yields a recurrence for the number r_n of regions formed by n lines; $r_n = r_{n-1} + n$, with $r_1 = 2$.

If the recurrence relation uses k earlier terms to compute a_n, then we must provide k initial values in order to specify the terms exactly; this is a recurrence of **order** k. Statements proved by induction about recurrences of order k typically require verification of k instances in the basis step. Standard techniques from enumerative combinatorics yield solutions to many recurrence relations without guessing formulas or directly using induction.

We also sometimes use recursive computation in graph theory. We may have a value for each graph instead of just one for each "size" as in a sequence. If we can express the value for a graph G as a formula in terms of graphs with fewer edges (and specify the values for graphs with no edges), then again we have a recurrence. We use this technique to count spanning trees (Section 2.2) and proper colorings (Section 5.3).

FUNCTIONS

A function transforms elements of one set into elements of another.

A.30. Definition. A **function** f *from* a set A *to* a set B assigns to each $a \in A$ a single element $f(a)$ in B, called the **image** of a under f. For a function f from A to B (written $f\colon A \to B$), the set A is the **domain** and the set B is the **target**. The **image** of a function f with domain A is $\{f(a)\colon a \in A\}$.

We take many elementary functions as familiar, such as the absolute value function and polynomials (both defined on $\mathbb{R}$). "Size" is a function whose domain is the set of finite sets and whose target is $\mathbb{N} \cup \{0\}$.

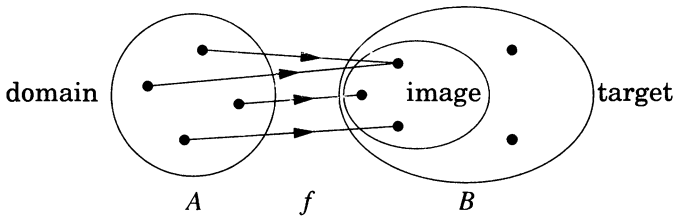
A.31. Definition. For $x \in \mathbb{R}$, the **floor** $\lfloor x \rfloor$ is the greatest integer that is at most x. The **ceiling** $\lceil x \rceil$ is the smallest integer that is at least x. A **sequence** is a function f whose domain is $\mathbb{N}$.

The floor function and ceiling function map $\mathbb{R}$ to $\mathbb{Z}$. When the target of a sequence is A, we have a sequence of elements in A, and we express the sequence as $a_1, a_2, a_3, \ldots$, where $a_n = f(n)$. We have used induction to prove sequences of statements and to prove formulas specifying sequences of numbers.

We may want to know how fast a function from $\mathbb{R}$ to $\mathbb{R}$ grows, particularly when analyzing algorithms. For example, we say that the growth of a function g is (at most) **quadratic** if it is bounded by a quadratic polynomial for all sufficiently large inputs. A more precise discussion of growth rates of functions appears in Appendix B.

A.32. Remark. *Schematic representation.* A function $f\colon A \to B$ is **defined on** A and **maps** A into B. To visualize a function $f\colon A \to B$, we draw a region representing A and a region representing B, and from each $x \in A$ we draw an arrow to $f(x)$ in B. In digraph language, this produces an orientation of a bipartite graph with partite sets A and B in which every element of A is the tail of exactly one edge.

The image of a function is contained in its target. Thus we draw the region for the image inside the region for the target. ■



To describe a function, we must specify $f(a)$ for each $a \in A$. We can list the pairs $(a, f(a))$, provide a formula for computing $f(a)$ from a, or describe the rule for obtaining $f(a)$ from a in words.

A.33. Definition. A function $f\colon A \to B$ is a **bijection** if for every $b \in B$ there is exactly one $a \in A$ such that $f(a) = b$.

Under a bijection, each element of the target is the image of exactly one element of the domain. Thus when a bijection is represented as in Remark A.32, every element of the target is the head of exactly one edge.

A.34. Example. *Pairing spouses.* Let M be the set of men at a party, and let W be the set of women. If the attendees consist entirely of married couples, then we can define a function $f\colon M \to W$ by letting $f(x)$ be the spouse of x. For each woman $w \in W$, there is exactly one $x \in M$ such that $f(x) = w$. Hence f is a bijection from M to W. ■

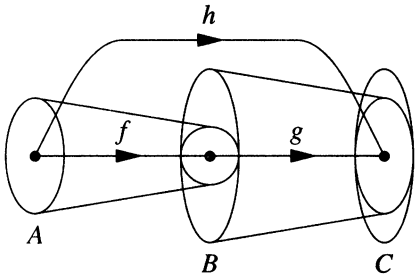
Bijections pair up elements from different sets. Thus we also describe a bijection from A to B as a **one-to-one correspondence** between A and B. Occasionally in the text we say informally that elements of one set "correspond" to elements of another; by this we mean that there is a natural one-to-one correspondence between the two sets.

When A has n elements, listing them as $a_1, \ldots, a_n$ defines a bijection from $[n]$ to A. Viewing the correspondence in the other direction defines a bijection from A to $[n]$. All bijections can be "inverted".

A.35. Definition. If f is a bijection from A to B, then the **inverse** of f is the function $g\colon B \to A$ such that, for each $b \in B$, $g(b)$ is the unique element $x \in A$ such that $f(x) = b$. We write f^{-1} for the function g.

When the target of a function is the domain of a second function, we can create a new function by applying the first and then the second. This yields a function from the domain of the first function into the target of the second.

A.36. Definition. If f: $A \to B$ and g: $B \to C$, then the **composition** of g with f is a function h: $A \to C$ defined by $h(x) = g(f(x))$ for $x \in A$. When h is the composition of g with f, we write $h = g \circ f$.



From the definitions, it is easy to verify that the composition of two bijections is a bijection. We use this in Proposition 1.1.24 in verifying for graphs that a composition of isomorphisms is an isomorphism.

COUNTING AND BINOMIAL COEFFICIENTS

A discussion of counting quickly leads to summations and products. These can be written concisely using appropriate notation.

A.37. Remark. We express summation using $\sum$, the uppercase Greek letter "sigma". When a and b are integers, the value of $\sum_{i=a}^{b} f(i)$ is the sum of the numbers $f(i)$ over the integers i satisfying $a \le i \le b$. Here i is the **index of summation**, and the formula $f(i)$ is the **summand**.

We write $\sum_{j \in S} f(j)$ to sum a real-valued function f over the elements of a set S in its domain. When no subset is specified, as in $\sum_j x_j$, we sum over the entire domain. When the summand has only one symbol that can vary, we may omit the subscript on the summation symbol, as in $\sum x_i$.

Similar comments apply to indexed products using $\prod$, which is the uppercase Greek letter "pi". ■

Two simple rules help organizing the counting of finite sets by breaking problems into subproblems. These rules follow from the definition of size and properties of bijections.

A.38. Definition. The **rule of sum** states that if A is a finite set and $B_1, \ldots, B_m$ is a partition of A, then $|A| = \sum_{i=1}^{m} |B_i|$:

Let T be a set whose elements can be described using a procedure

involving steps $S_1, \ldots, S_k$ such that step S_i can be performed in r_i ways, regardless of *how* steps $S_1, \ldots, S_{i-1}$ are performed. The **rule of product** states that $|T| = \prod_{i=1}^{k} r_i$.

For example, there are q^k lists of length k from a set of size q. There are q choices for each position, regardless of the choices in other positions. By the product rule, there are q^k ways to form the k-tuple.

A.39. Definition. A **permutation** of a finite set S is a bijection from S to S. The **word form** of a permutation f of $[n]$ is the list $f(1), \ldots, f(n)$ in that order. An **arrangement** of elements from a set S is a list of elements of S (in order). We write $n!$ (read as "n **factorial**") to mean $\prod_{i=1}^{n} i$, with the convention that $0! = 1$.

The word form of a permutation of $[n]$ includes the full description of the permutation. For counting purposes we refer to the word form *as* the permutation; thus 614325 is a permutation of [6]. With this viewpoint, a permutation of $[n]$ is an arrangement of all the elements of $[n]$.

A.40. Theorem. An n-element set has $n!$ permutations (arrangements without repetition). In general, the number of arrangements of k distinct elements from a set of size n is $n(n-1)\cdots(n-k+1)$.

Proof: We count the lists of k distinct elements from a set S of size n. There is no such list when $k > n$, which agrees with the formula. We construct the lists one element at a time, specifying the element in position $i+1$ after specifying the elements in earlier positions.

There are n ways to choose the image of 1. For each way we do this, there are $n-1$ ways to choose the image of 2. In general, after we have chosen the first i images, avoiding them leaves $n-i$ ways to choose the next image, no matter how we made the first i choices. The rule of product yields $\prod_{i=0}^{k-1}(n-i)$ for the number of arrangements. ∎

Often the order of elements in a list is unimportant.

A.41. Definition. A **selection** of k elements from $[n]$ is a k-element subset of $[n]$. The number of such selections is "n choose k", written as $\binom{n}{k}$.

If $k < 0$ or $k > n$, then $\binom{n}{k} = 0$; in these cases there are no selections of k elements from $[n]$. When $0 \le k \le n$, we obtain a simple formula.

A.42. Theorem. For integers n, k with $0 \le k \le n$, $\binom{n}{k} = \frac{1}{k!}\prod_{i=0}^{k-1}(n-i)$.

Proof: We relate selections to arrangements. We count the arrangements of k elements from $[n]$ in two ways. Picking elements for positions as in Theorem A.40 yields $n(n-1)\cdot(n-k+1)$ as the number of arrangements.

Alternatively, we can select the k-element subset first and then write it in some order. Since by definition there are $\binom{n}{k}$ selections, the product rule yields $\binom{n}{k}k!$ for the number of arrangements.

In each case, we are counting the set of arrangements, so we conclude that $n(n-1)\cdots(n-k+1) = \binom{n}{k}k!$. Dividing by $k!$ completes the proof. ∎

The formula for $\binom{n}{k}$ can be written as $\frac{n!}{k!(n-k)!}$, but the form in the statement of Theorem A.42 tends to be more useful, especially when k is small. For example, $\binom{n}{2} = n(n-1)/2$ and $\binom{n}{3} = n(n-1)(n-2)/6$, the former being the number of edges in a complete graph with n vertices. This form more directly reflects the counting argument and cancels the $(n-k)!$ appearing in both the numerator and denominator.

The numbers $\binom{n}{k}$ are called the **binomial coefficients** due to their appearance as coefficients in the nth power of a sum of two terms.

A.43. Theorem. (Binomial Theorem) For $n \in \mathbb{N}$, $(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$.

Proof: The proof interprets the process of multiplying out the factors in the product $(x+y)(x+y)\cdots(x+y)$. To form a term in the product, we must choose x or y from each factor. The number of factors that contribute x is some integer k in $\{0, \ldots, n\}$, and the remaining $n-k$ factors contribute y. The number of terms of the form $x^k y^{n-k}$ is the number of ways to choose k of the factors to contribute x. Summing over k accounts for all the terms. ∎

Using the definition of size and the composition of bijections, it follows that finite sets A and B have the same size if and only if there is a bijection from A to B. Thus we can compute the size of a set by establishing a bijection from it to a set of known size.

Simple examples include the statements that a complete graph has $\binom{n}{2}$ edges and that therefore there are $2^{\binom{n}{2}}$ simple graphs with vertex set $[n]$. Proposition 1.3.10 uses a bijection to count 6-cycles in the Petersen graph. Exercise 1.3.32 uses a bijection to count graphs with vertex set $[n]$ and even vertex degrees. Theorem 2.2.3 uses a bijection to count trees with vertex set $[n]$.

A.44. Lemma. For $n \in \mathbb{N}$, the number of subsets of $[n]$ with even size equals the number of subsets of $[n]$ with odd size.

Proof: *Proof 1* (bijection). For each subset with even size, delete the element n if it appears, and add n if it does not appear. This always changes the size by 1 and produces a subset with odd size. The map is a bijection, since each odd subset containing n arises only from one even subset omitting n, and each odd subset omitting n arises only from one even subset containing n.

Proof 2 (binomial theorem). Setting $x = -1$ and $y = 1$ in Theorem A.43. yields $\sum_{k=0}^{n} \binom{n}{k}(-1)^k = (-1+1)^n = 0$. (Note that we proved Theorem A.43 using bijections.) ∎

We prove a few identities involving binomial coefficients to illustrate combinatorial arguments involving bijections and the idea of counting a set in two ways. We can prove an equality by showing that both sides count the same set.

A.45. Lemma. $\binom{n}{k} = \binom{n}{n-k}$.

Proof: *Proof 1* (counting two ways). By definition, $[n]$ has $\binom{n}{k}$ subsets of size k. Another way to count selections of k elements is to count selections of $n-k$ elements to omit, and there are $\binom{n}{n-k}$ of these.

Proof 2 (bijections). The left side counts the k-element subsets of $[n]$, the right side counts the $n-k$-element subsets, and the operation of "complementation" establishes a bijection between the two collections. ∎

Often, "counting two ways" means grouping the elements in two ways. Sometimes one of the counts only gives a bound on the size of the set. In this case the counting argument proves an inequality; there are several instances of this phenomenon in Chapter 3 (see also Exercise 1.3.31). Here we stick to equalities.

A.46. Lemma. (The Chairperson Identity) $k\binom{n}{k} = n\binom{n-1}{k-1}$.

Proof: Each side counts the k-person committees with a designated chairperson that can be formed from a set of n people. On the left, we select the committee and then select the chair from it; on the right, we select the chair first and then fill out the rest of the committee. ∎

Many students see the next formula as the first application of induction, but it also is easily proved by counting a set in two ways.

A.47. Lemma. $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$.

Proof: The right side is $\binom{n+1}{2}$; we can view this as counting the nontrivial intervals with endpoints in the set $\{1, \ldots, n+1\}$. On the other hand, we can group the intervals by length; there is one interval with length n, two with length $n-1$, and so on up to n intervals with length 1. ∎

Lemma A.47 generalizes to $\sum_{i=k}^{n} \binom{i}{k} = \binom{n+1}{k+1}$. To prove this by counting in two ways, partition the set of $k+1$-element subsets of $[n+1]$ into groups so that the size of the ith group will be $\binom{i}{k}$.

Finally, a recursive computation for the binomial coefficients.

A.48. Lemma. (Pascal's Formula) If $n \geq 1$, then $\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$.

Proof: We count the k-sets in $[n]$. There are $\binom{n-1}{k}$ such sets not containing n and $\binom{n-1}{k-1}$ such sets containing n. ∎

Given the initial conditions for $n = 0$, which are $\binom{0}{0} = 1$ and $\binom{0}{k} = 0$ for $k \neq 0$, Pascal's Formula can be used to give inductive proofs of many statements about binomial coefficients, including Theorems A.42–A.43.

A.49. Remark. *Multinomial coefficients.* Binomial coefficients and the binomial theorem generalize to multinomials. When $\sum n_i = n$, the **multinomial coefficient** $\binom{n}{n_1,\ldots,n_k}$ is the coefficient of $\prod x_i^{n_i}$ in the expansion of $(\sum_{i=1}^{k} x_i)^n$. It has the value $n!/\prod n_i!$. Terms of the form $\prod x_i^{n_i}$ arise in the expansion only when $\sum n_i = n$. Otherwise, there is nothing to count, and we say that $\binom{n}{n_1,\ldots,n_k} = 0$ when $\sum n_i \neq n$.

The contributions to this coefficient correspond to n-tuples that are arrangements of n objects, using n_i copies of object i for each i. Having a copy of object i in position j corresponds to choosing the term x_i from the jth factor $(x_1 + \cdots + x_k)$.

The formula $n!/\prod n_i!$ is derived by counting these arrangements. There are $n!$ arrangements of n distinct items. If we view these objects as distinct, then we count each arrangement $\prod n_i!$ times, since permuting the copies of a single object does not change the arrangement.

In Corollary 2.2.4, these arrangements correspond to trees with vertex set $[n]$ and specified vertex degrees. When we set $x_i = 1$ for all i, we obtain the total number of n-tuples formed from k types of letters over all multiplicities of repetition; the result is k^n. ∎

RELATIONS

Given two objects s and t, not necessarily of the same type, we may ask whether they satisfy a given relationship. Let S denote the set of objects of the first type, and let T denote the set of objects of the second type. Some of the ordered pairs (s, t) may satisfy the relationship, and some may not. The next definition makes this notion precise.

A.50. Definition. When S and T are sets, a **relation** between S and T is a subset of the product $S \times T$. A **relation on** S is a subset of $S \times S$.

We usually specify a relation by a condition on pairs. In Section 1.1, we define several relations associated with a graph G. The *incidence relation* between $S = V(G)$ and $T = E(G)$ is the set of ordered pairs (v, e) such that $v \in V(G)$, $e \in E(G)$, and v is an endpoint of edge e. The *adjacency relation* on the set $V(G)$ is the set of ordered pairs (x, y) of vertices such that x and y are the endpoints of an edge.

A.51. Remark. Let R be a relation defined on a set S. When discussing several items from S, we use the adjective **pairwise** to specify that each pair among these items satisfies R. Thus we can talk about a family of pairwise disjoint sets, or a family of pairwise isomorphic graphs. An independent set in a graph is a set of pairwise nonadjacent vertices. A set of **distinct** objects is a set of pairwise unequal objects.

We need the term "pairwise" because the relation is defined for pairs. For the same reason, we don't use "pairwise" when discussing only two objects.

When two graphs are isomorphic, we don't say they are pairwise isomorphic. Similarly, we say that the endpoints of an edge are adjacent, not pairwise adjacent; the adjacency relation is satisfied by certain pairs of vertices. ∎

To specify a relation between S and T, we can list the ordered pairs satisfying it. Usually it is more convenient to let S index the rows and T the columns of a grid of positions called a **matrix**. We can then specify the relation by recording, in the position for row s and column t, a 1 if (s, t) satisfies the relation and a 0 if (s, t) does not satisfy the relation. Thus the adjacency and incidence matrices of a graph are the matrices recording the adjacency and incidence relations (see Definition 1.1.17).

The condition "have the same parity" defines a relation on $\mathbb{Z}$. If x, y are both even or both odd, then (x, y) satisfies this relation; otherwise it does not. The key properties of parity lead us to an important class of relations.

A.52. Definition. An **equivalence relation** on a set S is a relation R on S such that for all choices of distinct $x, y, z \in S$,
a) $(x, x) \in R$ (**reflexive property**).
b) $(x, y) \in R$ implies $(y, x) \in R$ (**symmetric property**).
c) $(x, y) \in R$ and $(y, z) \in R$ imply $(x, z) \in R$ (**transitive property**).

For every set S, the **equality relation** $R = \{(x, x): x \in S\}$ is an equivalence relation on S. In Proposition 1.1.24 we show that the isomorphism relation is an equivalence relation on graphs. The notation $G \cong H$ for this relation suggests "equal in some sense".

A.53. Definition. Given an equivalence relation on S, the set of elements equivalent to $x \in S$ is the **equivalence class** containing x.

The equivalence classes of an equivalence relation on S form a partition of S; elements x and y belong to the same class if and only if (x, y) satisfies the relation. The converse assertion also holds. If $A_1, \ldots, A_k$ is a partition of S, then the condition "x and y are in the same set in the partition" defines an equivalence relation on S.

Parity partitions the integers into two equivalence classes by their remainder upon division by 2. This notion generalizes to any natural number.

A.54. Definition. Given a natural number n, the integers x and y are **congruent modulo** n if $x - y$ is divisible by n. We write this as $x \equiv y \bmod n$. The number n is the **modulus**.

A.55. Theorem. For $n \in \mathbb{N}$, congruence mod n is an equivalence relation on $\mathbb{Z}$.

Proof: Reflexive property: $x - x$ equals 0, which is divisible by n.

Symmetric property: If $x \equiv y \bmod n$, then by definition $n|(x - y)$. Since $y - x = -(x - y)$, and since n divides $-m$ if and only if n divides m, we also have $n|(y - x)$, and hence $y \equiv x \bmod n$.

Transitive property: If $n|(x-y)$ and $n|(y-z)$, then integers a, b exist such that $x-y=an$ and $y-z=bn$. Adding these equations yields $x-z=an+bn=(a+b)n$, so $n|(x-z)$. Thus the relation is transitive. ∎

A.56. Definition. The equivalence classes of the relation "congruence modulo n" on $\mathbb{Z}$ are the **remainder classes** or **congruence classes** modulo n. The set of congruence classes is written as $\mathbb{Z}_n$ or $\mathbb{Z}/n\mathbb{Z}$.

There are n remainder classes modulo n. For $0 \leq r < n$, the rth class in $\mathbb{Z}_n$ is $\{kn+r: k \in \mathbb{Z}\}$. Numbers a and b lie in the rth class if and only if they both have remainder r upon division by n. Thus "$m \equiv r \mod n$" has the same meaning as "m is r more than a multiple of n".

THE PIGEONHOLE PRINCIPLE

The pigeonhole principle is a simple notion that leads to elegant proofs and can reduce case analysis. In every set of numbers, the average is between the minimum and the maximum. When dealing with integers, the pigeonhole principle allows us to take the ceiling or floor of the average in the desired direction.

A.57. Lemma. (Pigeonhole Principle) If a set consisting of more than kn objects is partitioned into n classes, then some class receives more than k objects.

Proof: The contrapositive states that if every class receives at most k objects, then in total there are at most kn objects. ∎

The pigeonhole principle can reduce case analysis by allowing us to use additional information about an extreme element of a set. This simple idea can crop up unexpectedly, but its use can be quite effective. When we find that we need the pigeonhole principle, there is no trouble applying it: we need a sufficiently big value in our set, and the pigeonhole principle provides it.

Some applications of the pigeonhole principle are rather subtle. Section 8.3 presents several of these. The subtlety arises when it is unclear how to define the objects and the classes so that the pigeonhole principle will apply.

Proposition 1.3.15 proves the next proposition using Remark A.13. Here we use the pigeonhole principle instead.

A.58. Proposition. If G is a simple n-vertex graph with $\delta(G) \geq (n-1)/2$, then G is connected.

Proof: Choose $u, v \in V(G)$. If $u \not\to v$, then at least $n-1$ edges join $\{u, v\}$ to the remaining vertices, since $\delta(G) \geq (n-1)/2$. There are $n-2$ other vertices, so the pigeonhole principle implies that one of them receives two of these edges. Since G is simple, this vertex is a common neighbor of u and v.

For every two vertices $u, v \in V(G)$, we have proved that u and v are adjacent or have a common neighbor. Thus G is connected. ∎

The pigeonhole principle can also be useful in statements about trees, where the number of vertices is one more than the number of edges. If each vertex selects an edge in some way, then some edge must be selected twice. The idea is to design the selection so that when an edge is selected twice, the desired outcome occurs. Applications of this idea occur in Lemma 8.1.10 and Theorem 8.3.2.

The pigeonhole principle is the discrete version of the statement that the average of a set of numbers is between the minimum and the maximum. This statement is made explicit for vertex degrees in Corollary 1.3.4. Other applications are sprinkled throughout the book.