Chapter 1

Logic and Set Theory

> To criticize mathematics for its abstraction is to miss the point entirely. Abstraction is what makes mathematics work. If you concentrate too closely on too limited an application of a mathematical idea, you rob the mathematician of his most important tools: analogy, generality, and simplicity.
>
> – *Ian Stewart*
> Does God play dice? The mathematics of chaos

In mathematics, a **proof** is a demonstration that, assuming certain axioms, some statement is necessarily true. That is, a proof is a logical argument, not an empirical one. One must demonstrate that a proposition is true in all cases before it is considered a theorem of mathematics. An unproven proposition for which there is some sort of empirical evidence is known as a **conjecture**. Mathematical logic is the framework upon which rigorous proofs are built. It is the study of the principles and criteria of valid inference and demonstrations.

Logicians have analyzed set theory in great details, formulating a collection of axioms that affords a broad enough and strong enough foundation to mathematical reasoning. The standard form of axiomatic set theory is denoted ZFC and it consists of the Zermelo-Fraenkel (ZF) axioms combined with the axiom of choice (C). Each of the axioms included in this theory expresses a property of sets that is widely accepted by mathematicians. It is unfortunately true that careless use of set theory can lead to contradictions. Avoiding such contradictions was one of the original motivations for the axiomatization of set theory.

A rigorous analysis of set theory belongs to the foundations of mathematics and mathematical logic. The study of these topics is, in itself, a formidable task. For our purposes, it will suffice to approach basic logical concepts informally. That is, we adopt a naive point of view regarding set theory and assume that the meaning of a set as a collection of objects is intuitively clear. While informal logic is not itself rigorous, it provides the underpinning for rigorous proofs. The rules we follow in dealing with sets are derived from established axioms. At some point of your academic career, you may wish to study set theory and logic in greater detail. Our main purpose here is to learn how to state mathematical results clearly and how to prove them.

1.1 Statements

A proof in mathematics demonstrates the truth of certain **statement**. It is therefore natural to begin with a brief discussion of statements. A statement, or **proposition**, is the content of an assertion. It is either true or false, but cannot be both true and false at the same time. For example, the expression "There are no classes at Texas A&M University today" is a statement since it is either true or false. The expression "Do not cheat and do not tolerate those who do" is not a statement. Note that an expression being a statement does not depend on whether we personally can verify its validity. The expression "The base of the natural logarithm, denoted e, is an irrational number" is a statement that most of us cannot prove.

Statements on their own are fairly uninteresting. What brings value to logic is the fact that there are a number of ways to form new statements from old ones. In this section, we present five ways to form new statements from old ones. They correspond to the English expressions: and; or; not; if, then; if and only if. In the discussion below, P and Q represent two abstract statements.

A logical **conjunction** is an operation on two logical propositions that produces a value of true if both statements are true, and is false otherwise. The conjunction (or logical AND) of P and Q, denoted by $P \wedge Q$, is precisely defined by

P	Q	$P \wedge Q$
T	T	T
T	F	F
F	T	F
F	F	F

.

Similarly, a logical **disjunction** is an operator on two logical propositions that is true if either statement is true or both are true, and is false otherwise. The disjunction (or logical OR) of P and Q, denoted $P \vee Q$, is defined by

P	Q	$P \vee Q$
T	T	T
T	F	T
F	T	T
F	F	F

.

In mathematics, a **negation** is an operator on the logical value of a proposition that sends true to false and false to true. The negation (or logical NOT) of P, denoted $\neg P$, is given by

P	$\neg P$
T	F
F	T

.

The next method of combining mathematical statements is slightly more subtle than the preceding ones. The **conditional connective** $P \to Q$ is a logical statement that is read "if P then Q" and defined by the truth table

P	Q	$P \to Q$
T	T	T
T	F	F
F	T	T
F	F	T

.

In this statement, P is called the **antecedent** and Q is called the **consequent**. The truth table should match your intuition when P is true. When P is false, students often think the resulting truth value should be undefined. Although the given definition may seem strange at first glance, this truth table is universally accepted by mathematicians.

To motivate this definition, one can think of $P \to Q$ as a promise that Q is true whenever P is true. When P is false, the promise is kept by default. For example, suppose your friend promises "if it is sunny tomorrow, I will ride my bike". We will call this a true statement if they keep their promise. If it rains and they don't ride their bike, most people would agree that they have still kept their promise. Therefore, this definition allows one to combine many statements together and detect broken promises without being distracted by uninformative statements.

Logicians draw a firm distinction between the **conditional connective** and the **implication relation**. They use the phrase "if P then Q" for the conditional connective and the phrase "P implies Q" for the implication relation. They explain the difference between these two forms by saying that the conditional is the contemplated relation, while the implication is the asserted relation. We will discuss this distinction in the Section 1.2, where we formally study relations between statements. The importance and soundness of the conditional form $P \to Q$ will become clearer then.

The logical **biconditional** is an operator connecting two logical propositions that is true if the statements are both true or both false, and it is false otherwise. The biconditional from P to Q, denoted $P \leftrightarrow Q$, is precisely defined by

P	Q	$P \leftrightarrow Q$
T	T	T
T	F	F
F	T	F
F	F	T

.

We read $P \leftrightarrow Q$ as "P if and only if Q." The phrase "if and only if" is often abbreviated as "iff".

Using the five basic operations defined above, it is possible to form more complicated compound statements. We sometimes need parentheses to avoid ambiguity in writing compound statements. We use the convention that $\neg$ takes precedence over the other four operations, but none of these operations takes precedence over the others. For example, let P, Q and R be three propositions. We wish to make a truth table for the following statement,

$$(P \to R) \wedge (Q \vee \neg R). \tag{1.1}$$

We can form the true table for this statement, using simple steps, as follows

P	Q	R	$(P$	$\to$	$R)$	$\wedge$	$(Q$	$\vee$	$\neg R)$
T	T	T	T	T	T	T	T	T	F
T	T	F	T	F	F	F	T	T	T
T	F	T	T	T	T	F	F	F	F
T	F	F	T	F	F	F	F	T	T
F	T	T	F	T	T	T	T	T	F
F	T	F	F	T	F	T	T	T	T
F	F	T	F	T	T	F	F	F	F
F	F	F	F	T	F	T	F	T	T
			1	5	2	7	3	6	4

.

We conclude this section with a brief mention of two important concepts. A **tautology** is a statement that is true in every valuation of its propositional variables, independent of the truth values assigned to these variables. The proverbial tautology is $P \vee \neg P$,

P	P	$\vee$	$\neg P$
T	T	T	F
F	F	T	T
	1	3	2

.

For instance, the statement "The Aggies won their last football game or the Aggies did not win their last football game" is true regardless of whether the Aggies actually defeated their latest opponent.

The negation of a tautology is a **contradiction**, a statement that is necessarily false regardless of the truth values of its propositional variables. The statement $P \wedge \neg P$ is a contradiction, and its truth table is

P	P	$\wedge$	$\neg P$
T	T	F	F
F	F	F	T
	1	3	2

.

Of course, most statements we encounter are neither tautologies nor contradictions. For example, (1.1) is not necessarily either true or false. Its truth value depends on the values of P, Q and R. Try to see whether the statement

$$((P \wedge Q) \to R) \to (P \to (Q \to R))$$

is a tautology, a contradiction, or neither.

1.2 Relations between Statements

Strictly speaking, relations between statements are not formal statements themselves. They are *meta-statements* about some propositions. We study two types of relations between statements, *implication* and *equivalence*. An example of an implication meta-statement is the observation that "if the statement 'Robert graduated from Texas A&M University' is true, then it implies that the statement 'Robert is an Aggie' is also true." Another example of a meta-statement is "the statement 'Fred is an Aggie and Fred is honest' being true is equivalent to the statement 'Fred is honest and Fred is an Aggie' being true." These two examples illustrate how meta-statements describe the relationship between statements. It is also instructive to note that implications and equivalences are the meta-statement analogs of conditionals and biconditionals.

Consider two compound statements P and Q that depend on other logical statements (e.g., $P = (R \rightarrow S) \wedge (S \rightarrow T)$ and $Q = R \rightarrow T$). A **logical implication** from P to Q, read as "P implies Q", asserts that Q must be true whenever P is true (i.e., for all possible truth values of the dependent statements R, S, T). Necessity is the key aspect of this sentence; the fact that P and Q both happen to be true cannot be coincidental. To state that P implies Q, denoted by $P \Rightarrow Q$, one needs the conditional $P \rightarrow Q$ to be true under all possible circumstances.

Meta-statements, such as "P implies Q", can be defined formally only when P and Q are both logical functions of other propositions. For example, consider $P = R \wedge (R \rightarrow S)$ and $Q = S$. Then, the truth of the statement $P \rightarrow Q$ depends only on the truth of external propositions R and S.

The notion of implication can be rigorously defined as follows, P implies Q if the statement $P \rightarrow Q$ is a tautology. We abbreviate P implies Q by writing $P \Rightarrow Q$. It is important to understand the difference between "$P \rightarrow Q$" and "$P \Rightarrow Q$." The former, $P \rightarrow Q$, is a compound statement that may or may not be true. On the other hand, $P \Rightarrow Q$ is a relation stating that the compound statement $P \rightarrow Q$ is true under all instances of the external propositions.

While the distinction between implication and conditional may seem extraneous, we will soon see that meta-statements become extremely useful in building valid arguments. In particular, the following implications are used extensively in constructing proofs.

Fact 1.2.1. *Let P, Q, R and S be statements.*

1. $(P \to Q) \wedge P \Rightarrow Q$.
2. $(P \to Q) \wedge \neg Q \Rightarrow \neg P$.
3. $P \wedge Q \Rightarrow P$.
4. $(P \vee Q) \wedge \neg P \Rightarrow Q$.
5. $P \leftrightarrow Q \Rightarrow P \to Q$.
6. $(P \to Q) \wedge (Q \to P) \Rightarrow P \to Q$.
7. $(P \to Q) \wedge (Q \to R) \Rightarrow P \to R$
8. $(P \to Q) \wedge (R \to S) \wedge (P \vee R) \Rightarrow Q \vee S$.

As an illustrative example, we show that $(P \to Q) \wedge (Q \to R)$ implies $P \to R$. To demonstrate this assertion, we need to show that

$$((P \to Q) \wedge (Q \to R)) \to (P \to R) \tag{1.2}$$

is a tautology. This is accomplished in the truth table below

P	Q	R	$((P$	$\to$	$Q)$	$\wedge$	$(Q$	$\to$	$R))$	$\to$	$(P$	$\to$	$R)$
T	T	T	T	T	T	T	T	T	T	T	T	T	T
T	T	F	T	T	T	F	T	F	F	T	T	F	F
T	F	T	T	F	F	F	F	T	T	T	T	T	T
T	F	F	T	F	F	F	F	T	F	T	T	F	F
F	T	T	F	T	T	T	T	T	T	T	F	T	T
F	T	F	F	T	T	F	T	F	F	T	F	T	F
F	F	T	F	T	F	T	F	T	T	T	F	T	T
F	F	F	F	T	F	T	F	T	F	T	F	T	F
			1	7	2	10	3	8	4	11	5	9	6

.

Column 11 has the truth values for statement (1.2). Since (1.2) is true under all circumstances, it is a tautology and the implication holds. Showing that the other relations are valid is left to the reader as an exercise.

Reversing the arrow in a conditional statement gives the **converse** of that statement. For example, the statement $Q \to P$ is the converse of $P \to Q$. This reversal may not preserve the truth of the statement though and therefore logical implications are not always reversible. For instance, although $(P \to Q) \wedge (Q \to R)$ implies $P \to R$, the converse is not always true. It can easily be seen from columns 9 & 10 above that

$$(P \to R) \to ((P \to Q) \wedge (Q \to R))$$

is not a tautology. That is, $P \to R$ certainly does not imply $(P \to Q) \wedge (Q \to R)$.

A logical implication that is reversible is called a **logical equivalence**. More precisely, P is equivalent to Q if the statement $P \leftrightarrow Q$ is a tautology. We denote the sentence "P is equivalent to Q" by simply writing "$P \Leftrightarrow Q$." The meta-statement $P \Leftrightarrow Q$ holds if and only if $P \Rightarrow Q$ and $Q \Rightarrow P$ are both true. Being able to recognize that two statements are equivalent will become handy. It is sometime possible to demonstrate a result by finding an alternative, equivalent form of the statement that is easier to prove than the original form. A list of important equivalences appears below.

Fact 1.2.2. *Let P, Q and R be statements.*

1. $\neg(\neg P) \Leftrightarrow P$.
2. $P \vee Q \Leftrightarrow Q \vee P$.
3. $P \wedge Q \Leftrightarrow Q \wedge P$.
4. $(P \vee Q) \vee R \Leftrightarrow P \vee (Q \vee R)$.
5. $(P \wedge Q) \wedge R \Leftrightarrow P \wedge (Q \wedge R)$.
6. $P \wedge (Q \vee R) \Leftrightarrow (P \wedge Q) \vee (P \wedge R)$.
7. $P \vee (Q \wedge R) \Leftrightarrow (P \vee Q) \wedge (P \vee R)$.
8. $P \to Q \Leftrightarrow \neg P \vee Q$.
9. $P \to Q \Leftrightarrow \neg Q \to \neg P$ *(Contrapositive).*
10. $P \leftrightarrow Q \Leftrightarrow (P \to Q) \wedge (Q \to P)$.
11. $\neg(P \wedge Q) \Leftrightarrow \neg P \vee \neg Q$ *(De Morgan's Law).*
12. $\neg(P \vee Q) \Leftrightarrow \neg P \wedge \neg Q$ *(De Morgan's Law).*

Given a conditional statement of the form $P \rightarrow Q$, we call $\neg Q \rightarrow \neg P$ the **contrapositive** of the original statement. The equivalence $P \rightarrow Q \Leftrightarrow \neg Q \rightarrow \neg P$ noted above is used extensively in constructing mathematical proofs.

One must be careful not to allow contradictions in logical arguments because, starting from a contradiction, anything can be proven true. For example, one can verify that $P \wedge \neg P \Rightarrow Q$ is a valid logical equivalence. But, Q doesn't appear on the LHS. Thus, a contradiction in your assumptions can lead to a "correct" proof for an arbitrary statement.

Fortunately, propositional logic has an axiomatic formulation that is consistent, complete, and decidable. In this context, the term **consistent** means that the logical implications generated by the axioms do not contain a contradiction, the term **complete** means that any valid logical implication can be generated by applying the axioms, and the term **decidable** means there is a terminating method that always determines whether a postulated implication is valid or invalid.

1.2.1 Fallacious Arguments

A **fallacy** is a component of an argument that is demonstrably flawed in its logic or form, thus rendering the argument invalid. Recognizing fallacies in mathematical proofs may be difficult since arguments are often structured using convoluted patterns that obscure the logical connections between assertions. We give below examples for three types of fallacies that are often found in attempted mathematical proofs.

Affirming the Consequent: If the Indian cricket team wins a test match, then all the players will drink tea together. All the players drank tea together. Therefore the Indian cricket team won a test match.

Denying the Antecedent: If Diego Maradona drinks coffee, then he will be fidgety. Diego Maradona did not drink coffee. Therefore, he is not fidgety.

Unwarranted Assumptions: If Yao Ming gets close to the basket, then he scores a lot of points. Therefore, Yao Ming scores a lot of points.

1.2.2 Quantifiers

Consider the statements “Socrates is a person” and “Every person is mortal”. In propositional logic, there is no formal way to combine these statements to deduce that “Socrates is mortal”. In the first statement, the noun “Socrates” is called the subject and the phrase “is a person” is called the **predicate**. Likewise, in predicate logic, the statement $P(x) =$ “x is a person” is called a predicate and x is called a **free variable** because its value is not fixed in the statement $P(x)$.

Let U be a specific collection of elements and let $P(x)$ be a statement that can be applied to any $x \in U$. In first-order predicate logic, quantifiers are applied to predicates in order to make statements about collections of elements. Later, we will see that quantifiers are of paramount importance in rigorous proofs.

The **universal quantifier** is typically denoted by $\forall$ and it is informally read “for all.” It follows that the statement “$\forall x \in U, P(x)$” is true if $P(x)$ is true for all values of x in U. It can be seen as shorthand for an iterated conjunction because

$$\forall x \in U, P(x) \Leftrightarrow \bigwedge_{x \in U} P(x),$$

where $\Leftrightarrow$ indicates that these statements are equivalent for all sets U and predicates P. If $U = \emptyset$ is the empty set, then $\forall x \in U, P(x)$ is vacuously true by convention because there are no elements in U to test with $P(x)$.

Returning to the motivating example, let us also define $Q(x) =$“x is mortal”. With these definitions, we can write the statement “Every person is mortal” as $\forall x, (P(x) \rightarrow Q(x))$. In logic, this usage implies that x ranges over the universal set. In engineering mathematics, however, the range of free variables is typically stated explicitly.

The other type of quantifier often seen in mathematical proofs is the **existential quantifier**, denoted $\exists$. The statement “$\exists x \in U, P(x)$” is true if $P(x)$ is true for at least one value of x in U. It can be seen as shorthand for an iterated disjunction because

$$\exists x \in U, P(x) \Leftrightarrow \bigvee_{x \in U} P(x),$$

From these definitions, it follows naturally that $\forall x \in U, P(x) \Rightarrow \exists x \in U, P(x)$. If $U = \emptyset$ is the empty set, then $\exists x \in U, P(x)$ is false by convention because there are no elements in U.

Based on the meaning of these quantifiers, one can infer the logical implications

$$\begin{aligned}\neg\left(\forall x \in U, P(x)\right) &\Leftrightarrow \exists x \in U, \neg P(x)\\ \neg\left(\exists x \in U, P(x)\right) &\Leftrightarrow \forall x \in U, \neg P(x).\end{aligned}$$

Using the connection to conjunction and disjunction, these rules are actually equivalent to De Morgan's law for iterated conjunctions and disjunctions.

One can also define predicates with multiple free variables such as $P(x, y) =$"x contains y". Once again, these statements are assumed to be true or false for every choice of x, y. There are 8 possible quantifiers for a 2-variable predicate and they can be arranged according to their natural implications:

$$\begin{tikzcd}
\forall x, \forall y, P(x,y) \arrow[r, Rightarrow] \arrow[d, Leftrightarrow] & \exists x, \forall y, P(x,y) \arrow[r, Rightarrow] & \forall y, \exists x, P(x,y) \arrow[r, Rightarrow] & \exists y, \exists x, P(x,y) \arrow[d, Leftrightarrow] \\
\forall y, \forall x, P(x,y) \arrow[r, Rightarrow] & \exists y, \forall x, P(x,y) \arrow[r, Rightarrow] & \forall x, \exists y, P(x,y) \arrow[r, Rightarrow] & \exists x, \exists y, P(x,y)
\end{tikzcd}$$

All of these implications follow from $\forall x \forall y = \forall y \forall x$, $\exists x \exists y = \exists y \exists x$, and the single variable inference rule $\forall x, P(x) \Rightarrow \exists x, P(x)$ except for two: $\exists x, \forall y, P(x,y) \Rightarrow \forall y, \exists x, P(x,y)$ and its symmetric pair.

To understand this last implication, consider an example where x is in a set I of images and y is in a set C of colors. Then, $\exists x, \forall y, P(x,y)$ means "there is an image that contains all the colors" (e.g., an image of a rainbow) and $\forall y, \exists x, P(x,y)$ means "for each color there is an image containing that color". The first statement implies the second because, in the second, the rainbow image satisfies the $\exists x$ quantifier for all y. To see that the implication is not an equivalence, consider a set of pictures where each image contains exactly one color and there is one such image for each color. In this case, it is true that "for each color there is an image containing that color" but it is not true that 'there is an image that contains all the colors".

In quantified statements, such as $\exists x \in U, P(x)$, the variable x is called a **bound variable** because its value cannot be chosen freely. Similarly, in the statement $\exists y \in U, P(x,y)$, x is a free variable and y is a bound variable.

Finally, we note that first-order predicate logic has an axiomatic formulation that is consistent, complete, and semidecidable. In this context, **semidecidable** means that there is an algorithm that, if it terminates, correctly determines the truth of any postulated implication. But, it is only guaranteed to terminate for true postulates.

1.3 Strategies for Proofs

The relation between intuition and formal rigor is not a trivial matter. Intuition tells us what is important, what might be true, and what mathematical tools may be used to prove it. Rigorous proofs are used to verify that a given statement which appears intuitively true is indeed true. Ultimately, a mathematical proof is a convincing argument that starts from some premises, and logically deduces the desired conclusion. Most proofs do not mention the logical rules of inference used in the derivation. Rather, they focus on the mathematical justification of each step, leaving to the reader the task of filling the logical gaps. The mathematics is the major issue. Yet, it is essential that you understand the underlying logic behind the derivation as to not get confused while reading or writing a proof.

True statements in mathematics have different names. They can be called theorems, propositions, lemmas, corollaries and exercises. A **theorem** is a statement that can be proved on the basis of explicitly stated or previously agreed assumptions. A **proposition** is a statement not associated with any particular theorem; this term sometimes connotes a statement with a simple proof. A **lemma** is a proven proposition which is used as a stepping stone to a larger result rather than an independent statement in itself. A **corollary** is a mathematical statement which follows easily from a previously proven statement, typically a mathematical theorem. The distinction between these names and their definitions is somewhat arbitrary. Ultimately, they are all synonymous to a true statement.

A proof should be written in grammatically correct English. Complete sentences should be used, with full punctuation. In particular, every sentence should end with a period, even if the sentence ends in a displayed equation. Mathematical formulas and symbols are parts of sentences, and are treated no differently than words. One way to learn to construct proofs is to read a lot of well written proofs, to write progressively more difficult proofs, and to get detailed feedback on the proofs you write.

Direct Proof: The simplest form of proof for a statement of the form $P \to Q$ is the **direct proof**. First assume that P is true. Produce a series of steps, each one following from the previous ones, that eventually leads to conclusion Q. It warrants the name “direct proof” only to distinguish it from other, more intricate, methods of proof.

Proof by Contrapositive: A proof by contrapositive takes advantage of the mathematical equivalence $P \to Q \Leftrightarrow \neg Q \to \neg P$. That is, a proof by contrapositive begins by assuming that Q is false (i.e., $\neg Q$ is true). It then produces a series of direct implications leading to the conclusion that P is false (i.e., $\neg P$ is true). It follows that Q cannot be false when P is true, so $P \to Q$.

Proof by Contradiction: A proof by contradiction is based on the mathematical equivalence $\neg(P \to Q) \Leftrightarrow P \wedge \neg Q$. In a proof by contradiction, one starts by assuming that both P and $\neg Q$ are true. Then, a series of direct implications are given that lead to a logical contradiction. Hence, $P \wedge \neg Q$ cannot be true and $P \to Q$.

Example 1.3.1. *We wish to show that $\sqrt{2}$ is an irrational number.*

First, suppose that $\sqrt{2}$ is a rational number. This would imply that there exist integers p and q with $q \neq 0$ such that $p/q = \sqrt{2}$. In fact, we can further assume that the fraction p/q is irreducible. That is, p and q are coprime integers (they have no common factor greater than 1). From $p/q = \sqrt{2}$, it follows that $p = \sqrt{2}q$, and so $p^2 = 2q^2$. Thus p^2 is an even number, which implies that p itself is even (only even numbers have even squares). Because p is even, there exists an integer r satisfying $p = 2r$. We then obtain the equation $(2r)^2 = 2q^2$, which is equivalent to $2r^2 = q^2$ after simplification. Because $2r^2$ is even, it follows that q^2 is even, which means that q is also even. We conclude that p and q are both even. This contradicts the fact that p/q is irreducible. Hence, the initial assumption that $\sqrt{2}$ is a rational number must be false. That is to say, $\sqrt{2}$ is irrational.

Example 1.3.2. *Consider the following statement, which is related to Example 1.3.1. "If $\sqrt{2}$ is rational, then $\sqrt{2}$ can be expressed as an irreducible fraction." The contrapositive of this statement is "If $\sqrt{2}$ cannot be expressed as an irreducible fraction, then $\sqrt{2}$ is not rational." Above, we proved that $\sqrt{2}$ cannot be expressed as an irreducible fraction and therefore $\sqrt{2}$ is not a rational number.*

The final proof strategy we discuss is finite induction.

Definition 1.3.3. *Let $P(n)$ be a logical statement for each $n \in \mathbb{N}$. The principle of* ***mathematical induction*** *states that $P(n)$ is true all $n \in \mathbb{N}$ if:*

1. *$P(1)$ is true, and*

2. *$P(n) \to P(n+1)$ for all $n \in \mathbb{N}$.*

From a foundational perspective, this statement is essentially equivalent to the existence and uniqueness of the natural numbers. It is taken as an axiom in the Peano axiomatic formulation of arithmetic. In contrast, the ZF axiomatic formulation of set theory defines the natural numbers as the smallest inductive set and the existence of an inductive set is taken as an axiom.

Example 1.3.4. *Let $S_n = \sum_{i=1}^{n} i$. We wish to show that the statement $P(n) =$ "$S_n = \frac{n^2+n}{2}$" is true for all $n \in \mathbb{N}$. For $n = 1$, this is true because both expressions equal 1. For $P(n+1)$, we are given $P(n)$ and can write*

$$S_{n+1} = S_n + (n+1) = \frac{n^2+n}{2} + n + 1 = \frac{n^2+3n+2}{2} = \frac{(n+1)^2+(n+1)}{2}.$$

Thus, the result follows from mathematical induction.

More general forms of finite induction are also quite common but they can reduced to the original form. For example, let $Q(m)$ be a predicate for $m \in \mathbb{N}$ and define $P(n) =$"$\forall m \in S_n, Q(m)$" for a sequence nested finite sets $S_1 \subset S_2 \subset \cdots \subseteq \mathbb{N}$. Defining $S_\infty = \cup_{n\in\mathbb{N}} S_n$, we see that "$\forall n \in \mathbb{N}, P(n)$"$\Leftrightarrow$"$\forall m \in S_\infty, Q(m)$" follows from $P(1) =$"$\forall m \in S_1, Q(m)$" and "$P(n) \to P(n+1)$"$\Leftrightarrow$"$\forall m \in S_n, Q(m) \to \forall m \in S_{n+1}, Q(m)$".

1.4 Set Theory

Set theory is generally considered to be the foundation of all modern mathematics. This means that most mathematical objects (numbers, relations, functions, etc.) are defined in terms of sets. Unfortunately for engineers, set theory is not quite as simple as it seems. It turns out that simple approaches to set theory include paradoxes (e.g., statements which are both true and false). These paradoxes can be resolved by putting set theory in a firm axiomatic framework, but that exercise is rather unproductive for engineers. Instead, we adopt what is called **naive set theory** which rigorously defines the operations of set theory without worrying about possible contradictions. This approach is sufficient for most of mathematics and also acts as a stepping-stone to more formal treatments.

A **set** is taken to be any collection of objects, mathematical or otherwise. For example, one can think of “the set of all books published in 2007”. The objects in a set are referred to as **elements** or members of the set. The logical statement “a is a member of the set A” is written

$$a \in A.$$

Likewise, its logical negation “a is not a member of the set A” is written $a \notin A$. Therefore, exactly one of these two statements is true. In naive set theory, one assumes the existence of any set that can be described in words. Later, we will see that this can be problematic when one considers objects like the “set of all sets”.

One may present a set by listing its elements. For example, $A = \{a, e, i, o, u\}$ is the set of standard English vowels. It is important to note that the order elements are presented is irrelevant and the set $\{i, o, u, a, e\}$ is the same as A. Likewise, repeated elements have no effect and the set $\{a, e, i, o, u, e, o\}$ is the same as A. A **singleton** set is a set containing exactly one element such as $\{a\}$.

There are a number of standard sets worth mentioning: the **integers** $\mathbb{Z}$, the **real numbers** $\mathbb{R}$, and the **complex numbers** $\mathbb{C}$. It is possible to construct these sets in a rigorous manner, but instead we will assume their meaning is intuitively clear. New sets can be defined in terms of old sets using **set-builder notation**. Let $P(x)$ be a logical statement about objects x in the set X, then the “set of elements in X such that $P(x)$ is true” is denoted by

$$\{x \in X | P(x)\}.$$

For example, the set of even integers is given by

$$\{x \in \mathbb{Z} | \text{“}x \text{ is even”}\} = \{\ldots, -4, -2, 0, 2, 4, \ldots\}.$$

If no element $x \in X$ satisfies the condition, then the result is the **empty set** which is denoted $\emptyset$. Using set-builder notation, we can also recreate the **natural numbers** $\mathbb{N}$ and the **rational numbers** $\mathbb{Q}$ with

$$\begin{aligned}\mathbb{N} &= \{n \in \mathbb{Z} | n \geq 1\}\\ \mathbb{Q} &= \{q \in \mathbb{R} | q = a/b, a \in \mathbb{Z}, b \in \mathbb{N}\}.\end{aligned}$$

The following standard notation is used for interval subsets of the real numbers:

$$\begin{aligned}\text{Open interval: } (a,b) &\triangleq \{x \in \mathbb{R} | a < x < b\}\\ \text{Closed interval: } [a,b] &\triangleq \{x \in \mathbb{R} | a \le x \le b\}\\ \text{Half-open intervals: } (a,b] &\triangleq \{x \in \mathbb{R} | a < x \le b\}\\ [a,b) &\triangleq \{x \in \mathbb{R} | a \le x < b\}\end{aligned}$$

Definition 1.4.1. *For a finite set A, the **cardinality** $|A|$ equals the number of elements in A. If there is a bjiective mapping between the set A and the natural numbers $\mathbb{N}$, then $|A| = \infty$ and the set is called **countably infinite**. If $|A| = \infty$ and the set is not countably infinite, then A is called **uncountably infinite**.*

Example 1.4.2. *The set of rational numbers is countably infinite while the set of real numbers is uncountably infinite.*

Example 1.4.3 (Russell's Paradox). *Let R be the set of all sets that do not contain themselves or $R = \{S | S \notin S\}$. Such a set is said to exist in naive set theory (though it may empty) simply because it can be described in words. The paradox arises from the fact that the definition leads to the logical contradiction $R \in R \leftrightarrow R \notin R$.*

What this proves is that *naive set theory is not consistent* because it allows constructions that lead to contradictions. Axiomatic set theory eliminates this paradox by disallowing self-referential and other problematic constructions. Thus, another reasonable conclusion is that Russell's paradox shows that the set R cannot exist in any consistent theory of sets.

Another common question is whether there are sets that contains themselves. In naive set theory, the answer is yes and some examples are the "set of all sets" and the "set of all abstract ideas". On the other hand, in the ZF axiomatic formulation of set theory, it is a theorem that no set contains itself.

There are a few standard relationships defined between any two sets A, B.

Definition 1.4.4. *We say that A **equals** B (denoted $A = B$) if, for all x, $x \in A$ iff $x \in B$. This means that*

$$A = B \Leftrightarrow \forall x \left((x \in A) \leftrightarrow (x \in B) \right).$$

Definition 1.4.5. *We say that A is a **subset** of B (denoted $A \subseteq B$) if, for all x, if $x \in A$ then $x \in B$. This means that*

$$A \subseteq B \Leftrightarrow \forall x \left((x \in A) \to (x \in B) \right).$$

*It is a **proper subset** (denoted $A \subset B$) if $A \subseteq B$ and $A \neq B$.*

There are also a number of operations between sets. Let A, B be any two sets.

Definition 1.4.6. *The **union** of A and B (denoted $A \cup B$) is the set of elements in either A or B. This means that $A \cup B = \{x \in A \text{ or } x \in B\}$ is also defined by*

$$x \in A \cup B \Leftrightarrow (x \in A) \vee (x \in B).$$

Definition 1.4.7. *The **intersection** of A and B (denoted $A \cap B$) is the set of elements in both A and B. This means that $A \cap B = \{x \in A | x \in B\}$ is also defined by*

$$x \in A \cap B \Leftrightarrow (x \in A) \wedge (x \in B).$$

*Two sets are said to be **disjoint** if $A \cap B = \emptyset$.*

Definition 1.4.8. *The **set difference** between A and B (denoted $A - B$ or $A \backslash B$) is the set of elements in A but not in B. This means that*

$$x \in A - B \Leftrightarrow (x \in A) \wedge (x \notin B).$$

*If there is some implied universal set U, then the **complement** (denoted A^c) is defined by $A^c = U - A$*

One can apply De Morgan's Law in set theory to verify that

$$\begin{aligned}(A \cup B)^c &= A^c \cap B^c \\ (A \cap B)^c &= A^c \cup B^c,\end{aligned}$$

which allows us to interchange union or intersection with set difference.

We can also form the union or the intersection of arbitrarily many sets. This is defined in a straightforward way,

$$\begin{aligned}\bigcup_{\alpha \in I} S_\alpha &= \{x | x \in S_\alpha \text{ for some } \alpha \in I\} \\ \bigcap_{\alpha \in I} S_\alpha &= \{x | x \in S_\alpha \text{ for all } \alpha \in I\}.\end{aligned}$$

It is worth noting that the definitions apply whether the index set is finite, countably infinite, or even uncountably infinite.

Another way to build sets is by grouping elements into pairs, triples, and vectors.

Definition 1.4.9. *The **Cartesian Product**, denoted $A \times B$, of two sets is the set of ordered pairs $\{(a,b)|a \in A, b \in B\}$. For n-tuples taken from the same set, the notation A^n denotes the n-fold product $A \times A \times \cdots \times A$.*

Example 1.4.10. *If $A = \{a,b\}$, then the set of all 3-tuples from A is given by*

$$A^3 = \{(a,a,a),(a,a,b),(a,b,a),(a,b,b),(b,a,a),(b,a,b),(b,b,a),(b,b,b)\}.$$

The countably infinite product of X, denoted X^ω, is the set of infinite sequences $(x_1, x_2, x_3, \ldots)$ where $x_n \in X$ is arbitrary for $n \in \mathbb{N}$. If the sequences are restricted to have only a finite number of non-zero terms, then the set is usually denoted X^∞.

One can also formalize relationships between elements of a set. A **relation** $\sim$ between elements of the set A is defined by the pairs $(x,y) \in A \times A$ for which the relation holds. Specifically, the relation is defined by the subset of ordered pairs $E \subseteq A \times A$ where the relation $a \sim b$ holds; so $x \sim y$ if and only if $(x,y) \in E$. A relation on A is said to be:

1. Reflexive if $x \sim x$ holds for all $x \in A$

2. Symmetric if $x \sim y$ implies $y \sim x$ for all $x, y \in A$

3. Transitive if $x \sim y$ and $y \sim z$, then $x \sim z$ for all $x, y, z \in A$

A relation is called an **equivalence relation** if it is reflexive, symmetric, and transitive. For example, let A be a set of people and $P(x,y)$ be the statement "x has the same birthday (month and day) as y." Then, we can define $\sim$ such that $a \sim b$ holds if and only if $P(x,y)$ is true. In this case, the set E is given by $E = \{(x,y) \in A \times A | P(x,y)\}$. One can verify that this is an equivalence relation by checking that it is reflexive, symmetric, and transitive.

One important characteristic of an equivalence relation is that it partitions the entire set A into disjoint **equivalence classes**. The equivalence class associated with $a \in A$ is given by $[a] = \{x \in A | x \sim a\}$. In the birthday example, there is a natural equivalence class associated with each day of the year. The set of all equivalence classes is called the **quotient set** and is denoted $A/{\sim} = \{[a] | a \in A\}$.

In fact, there is a natural equivalence relation defined by any disjoint partition of a set. For example, let $A_{i,j}$ be the set of people in A whose birthday was on the j-th day of the i-th month. It follows that $x \sim y$ if and only if there exists a unique pair i, j such that $x, y \in A_{i,j}$. In this case, the days of year are used as equivalence classes to define the equivalence relation.

Example 1.4.11. *Consider the set* $\mathbb{N} \times \mathbb{N} = \{(a,b) | a, b \in \mathbb{N}\}$ *of ordered pairs of natural numbers. If one associates the element* (a,b) *with the fraction* a/b*, then the entire set is associated with the set of (possibly reducible) fractions. Now, consider the equivalence relation* $(a,b) \sim (c,d)$ *if* $ad = bc$*. In this case, two ordered pairs are equivalent if their associated fractions evaluate to the same real number. The quotient set* $\mathbb{N}/\!\sim$ *can therefore be associated with the set of reduced fractions.*

Unfortunately, this section will not end on a happy note by saying that the ZFC axiomatic formulation of set theory is consistent. Instead, we observe that Kurt Gödel's Incompleteness Theorems imply that, if ZFC is consistent, then this cannot be proven using statements in ZFC and, moreover, it cannot be complete. On the other hand, if ZFC is inconsistent, then it contains a paradox and one can prove anything using statements in ZFC. Since ZFC manages to avoid all known paradoxes and no contradictions have been so far, it is still the most popular formal system in which to define mathematics.

1.5 Functions

In elementary mathematics, functions are typically described in terms of graphs and formulas. The drawback of this approach is that one tends to picture only "nice" functions. In fact, Cauchy himself published in 1821 an incorrect proof of the false assertion that "a sequence of continuous functions that converges everywhere has a continuous limit function." Nowadays, every teacher warns their students that one must be careful because the world is filled with "not so nice" functions.

The modern approach to defining functions is based on set theory. A **function** $f \colon X \to Y$ is a rule that assigns a single value $f(x) \in Y$ to each element $x \in X$. The notation $f \colon X \to Y$ is used to emphasize the role of the **domain** X and the **codomain** Y. The **range** of f is the subset of Y which is actually achieved by f, $\{f(x) \in Y | x \in X\}$. Since the term codomain is somewhat uncommon, people often use the term range instead of codomain either intentionally (for simplicity) or unintentionally (due to confusion).

Definition 1.5.1. *Formally, a* ***function*** $f\colon X \to Y$ *from* X *to* Y *is defined by a subset* $F \subset X \times Y$ *such that* $A_x = \{y \in Y | (x, y) \in F\}$ *has exactly one element for each* $x \in X$*. The* ***value*** *of* f *at* $x \in X$*, denoted* $f(x)$*, is the unique element of* Y *contained in* A_x*.*

Two functions are said to be equal if they have the same domain, codomain, and value for all elements of the domain. A function f is called:

1. **one-to-one** or **injective** if, for all $x, x' \in X$, if $f(x) = f(x')$ then $x = x'$;

2. **onto** or **surjective** if its range $\{f(x)|x \in X\}$ equals Y;

3. a **one-to-one correspondence** or **bijective** if it is both one-to-one and onto.

A bijective function $f\colon X \to Y$ has a unique **inverse function** $f^{-1}\colon Y \to X$ such that $f^{-1}(f(x)) = x$ for all $x \in X$ and $f(f^{-1}(y)) = y$ for all $y \in Y$. In fact, any one-to-one function $f\colon X \to Y$ can be transformed into a bijective function $g\colon X \to R$ with $g(x) = f(x)$ by restricting its codomain Y to its range R.

Functions can also be applied to sets in a natural way. For a function $f\colon X \to Y$ and subset $A \subseteq X$, the **image** of A under f is

$$f(A) \triangleq \{y \in Y | \exists x \in A \text{ s.t. } f(x) = y\} = \{f(x)|x \in A\}.$$

Using this definition, we see that the range of f is simply $f(X)$. One benefit of allowing functions to have set-valued images is that a set-valued inverse function always exists. The **inverse image** or **preimage** of a subset $B \subseteq Y$ is

$$f^{-1}(B) \triangleq \{x \in X | f(x) \in B\}.$$

For a one-to-one function f, the inverse image of any singleton set $\{f(x)\}$ is the singleton set $\{x\}$. It is worth noting that the notation $f^{-1}(B)$ for the preimage of B can be somewhat misleading because, in some cases, $f^{-1}(f(A)) \neq A$. In general, a function gives rise to the following property, $f(f^{-1}(B)) \subseteq B$ and $f^{-1}(f(A)) \supseteq A$.

Example 1.5.2. *Let the function* $f\colon \mathbb{R} \to \mathbb{R}$ *be defined by* $f(x) = x^2$*. Let* $A = [1, 2]$ *and notice that* $B = f(A) = [1, 4]$*. Then,*

$$f^{-1}(B) = f^{-1}([1, 4]) = [-2, -1] \cup [1, 2] \supseteq A.$$

Example 1.5.3. *Let the function* $f\colon \mathbb{R} \to \mathbb{R}$ *be defined by* $f(x) = x^2 + 1$. *Let* $B = [0, 2]$ *and notice that* $A = f^{-1}(B) = [-1, 1]$. *Then,*

$$f(A) = f([-1, 1]) = [1, 2] \subseteq B.$$

Problem 1.5.4. *For all* $f : X \to Y$*,* $A \subseteq X$*, and* $B \subseteq Y$*, we have the rules:*

$$(a)\ x \in A \Rightarrow f(x) \in f(A) \qquad (b)\ y \in f(A) \Rightarrow \exists x \in A \text{ s.t. } f(x) = y$$
$$(c)\ x \in f^{-1}(B) \Rightarrow f(x) \in B \qquad (d)\ f(x) \in B \Rightarrow x \in f^{-1}(B).$$

Use these rules to show that $f^{-1}(f(A)) \supseteq A$ *and* $f(f^{-1}(B)) \subseteq B$.

Solution 1.5.4. The first result follows from

$$x \in A \overset{(a)}{\Rightarrow} f(x) \in f(A) \overset{(d)}{\Rightarrow} x \in f^{-1}(f(A)),$$

and the definition of subset. The second result follows from

$$y \in f(f^{-1}(B)) \overset{(b)}{\Rightarrow} \exists x \in f^{-1}(B) \text{ s.t. } f(x) = y \overset{(c)}{\Rightarrow} y \in B,$$

and the definition of subset.

Problem 1.5.5. *Let* $f\colon X \to Y$*,* $A_i \subseteq X$ *for all* $i \in I$*, and* $B_i \subseteq Y$ *for all* $i \in I$*. Show that the following expressions hold:*

$$(1)\quad f\left(\bigcup_{i\in I} A_i\right) = \bigcup_{i\in I} f\left(A_i\right) \qquad (2)\quad f\left(\bigcap_{i\in I} A_i\right) \subseteq \bigcap_{i\in I} f\left(A_i\right)$$
$$(3)\ f^{-1}\left(\bigcup_{i\in I} B_i\right) = \bigcup_{i\in I} f^{-1}\left(B_i\right) \qquad (4)\ f^{-1}\left(\bigcap_{i\in I} B_i\right) = \bigcap_{i\in I} f^{-1}\left(B_i\right).$$

Advanced Counting Techniques

Many counting problems cannot be solved by the previous counting techniques.

Example: How many bit strings of length n do not contain 2 consecutive 0's?

Answer: $a_n = a_{n-1} + a_{n-2}$, $a_1 = 2$, $a_2 = 3$.

The answer is a recurrence relation.

Example: Compound interest at 7%.

$$P_n = P_{n-1} + 0.07P_{n-1} = 1.07P_{n-1}$$
$$P_n = (1.07)^n P_0$$

The Tower of Hanoi: The problem of moving n disks from one peg to another peg, one at a time, via a third peg in such a way that no disk is on top of a smaller one. Let H_n be a minimum number of moves needed to solve the problem.
We can summarize the solution as follows:

move $n-1$ top disks from peg 1 to 2
move the largest disk from peg 1 to peg 3
move $n-1$ disks from peg 2 to 3

Thus we have

$$\begin{aligned}
&H_n = H_{n-1} + 1 + H_{n-1} \\
&H_n = 2H_{n-1} + 1 \\
&H_n = 2(2H_{n-2} + 1) + 1 = 2^2 H_{n-2} + 2 + 1 \\
&\quad \ldots\ldots\ldots\ldots \\
&H_n = 2^{n-1} H_1 + 2^{n-2} + 2^{n-3} + \ldots. + 1 \\
&H_n = \textstyle\sum_{i=0}^{n-1} 2^i = 2^n - 1
\end{aligned}$$

Example: How many bit strings of length n do not contain 2 consecutive 0's?

Answer: Denote by a_n the number of such strings. We will try to relate a_n with a_{n-1} and a_{n-2}.

Case 1: If the string begins with a 1 then it can be followed by a_{n-1} strings that do not contain 2 consecutive 0's.

Case 2: If the string begins with a 0 then the next bit must be 1, and it can be followed by a_{n-2} strings that do not contain 2 consecutive 0's.
Therefore: $a_n = a_{n-1} + a_{n-2}$, the initial conditions are easily found: $a_1 = 2$ and $a_2 = 3$.

Example: How many strings of n decimal digits (0-9) contain an even number of 0's?

Answer: Let a_n denote the number of such strings. Hence

$a_1 = 9$.

$a_2 = 9 \cdot 9 + 1 \ = \ 82$

Case 1: Take a valid string length $n-1$ and append a digit $\neq 0$ (there are 9):

There are: $9a_{n-1}$ such strings.

Case 2: Take a non-valid string length $n-1$ and append a 0:

There are: $(10^{n-1} - a_{n-1})$ such strings.

The total is: $a_n = 10^{n-1} - a_{n-1} + 9a_{n-1} = 8a_{n-1} + 10^{n-1}$.

Problems:

1. How many bit strings of length n do not contain 00?

2. How many bit strings of length n contain 00?

3. How many bit strings of length 7 either begin with 00 or (inclusive or) end with 111?

4. How many bit strings of length 10 either have 5 consecutive $0's$ or 5 consecutive $1's$?

Solving Linear Recurrences

Linear homogeneous recurrences

Linear homogeneous recurrence relation of degree k with constant coefficients:

$$a_n = c_1 a_{n-1} + \ldots + c_k a_{n-k}$$

where $c_1, \ldots, c_k$ are real numbers with $c_k \neq 0$.

Characteristic equation:

$$r^k - c_1 r^{k-1} - \ldots - c_{k-1} r - c_k = 0.$$

For example, solve $a_n - 5a_{n-1} + 8a_{n-2} - 4a_{n-3} = 0$ with proper initial conditions.

Method:

1. Find the characteristic equation for the homogeneous recurrence.

2. Solve the characteristic equation for the roots $r_1, r_2, ..., r_k$.

There are two cases:

Case 1 If all the roots are distinct, then the solution is of the form:

$$C_1 r_1^n + C_2 r_2^n + ... + C_k r_k^n$$

Case 2 If some roots are the same,
for example, three roots with $r_1 = r_2 = r_3 = r$
then the solution is $C_1 r^n + C_2 n r^n + C_3 n^2 r^n$.
If the three roots are $r_1, r_2 = r_3 = r$
then the solution is $C_1 r_1^n + C_2 r^n + C_3 n r^n$.
If the three roots are $r_1 = r_2 = r, r_3$
then the solution is $C_1 r^n + C_2 n r^n + C_3 r_3^n$.

Examples

Example 1. Solve $a_n - 3a_{n-1} - 4a_{n-2} = 0, n \geq 2$, $a_0 = 0, a_1 = 1$

Solution. The characteristic equation is $r^2 - 3r - 4 = 0$ which has roots $r_1 = -1, r_2 = 4$ which are distinct, so use Case 1:

$a_n = C_1(-1)^n + C_2(4)^n$.

Using the initial conditions we obtain

$$\begin{aligned} a_0 &= 0 = C_1 + C_2 \\ a_1 &= 1 = -C_1 + 4C_2. \end{aligned}$$

Solving the system above yields $C_1 = -1/5$ and $C_2 = 1/5$, thus

$$a_n = \frac{(-1)^{n+1} + 4^n}{5}.$$

Example 2. Solve
$a_n - 5a_{n-1} + 8a_{n-2} - 4a_{n-3} = 0$,
$n \geq 3, a_0 = 0, a_1 = 1, a_2 = 2$.

Solution. Characteristic equation: $r^3 - 5r^2 + 8r - 4 = 0$. We find one root $r = 1$. We factor the equation $r^3 - 5r^2 + 8r - 4 = 0$ by dividing the left side by $r - 1$ $\Rightarrow (r-1)(r-2)^2 = 0 \Rightarrow r_1 = 1 (multiplicity = 1), r_2 = 2 (multiplicity = 2)$. It's Case 2.
Therefore, $a_n = C_1 1^n + C_2 2^n + C_3 n 2^n$.
With initial conditions $a_0 = 0, a_1 = 1, a_2 = 2$
we find $C_1 = -2, C_2 = 2, C_3 = -1/2$
Therefore

$$a_n = -2 + 2(2^n) - \frac{1}{2}n(2^n).$$

Linear nonhomogeneous recurrences

Linear nonhomogeneous recurrence relation of degree k with constant coefficients:

$$a_n = c_1 a_{n-1} + \ldots + c_k a_{n-k} + F(n)$$

where $c_1, \ldots, c_k$ are real numbers with $c_k \neq 0$ and $F(n)$ is the function not identically zero depending only on n.

For example, solve $a_n - 5a_{n-1} + 6a_{n-2} = 7^n$ with proper initial conditions. Denote $F(n) =$ the nonhomogeneous part, i.e. $F(n) = 7^n$ in this example.

1. Find the solution $a_n^{(h)}$ of the associated homogeneous recurrence (see above).

2. Find a particular solution $a_n^{(p)}$ of the nonhomogeneous equation. A particular solution can be found by using a *trial solution*. The trial solution depends on the nonhomogeneous term $F(n)$.

Case 1 If $F(n)$ has the form $p(n)s^n$ where $p(n)$ is a polynomial in n of degree k, s is a constant, and if s **is not** a root of the characteristic equation, then the trial solution has the same form as $q(n)s^n$, where $q(n)$ is a polynomial of degree k (see example below).

Case 2 If $F(n)$ has the form $p(n)s^n$ where $p(n)$ is a polynomial in n of degree k and s is a root of the characteristic equation with multiplicity m, then the trial solution has the form $n^m q(n)s^n$, where $q(n)$ has the same degree as $p(n)$.

3. The solution to the nonhomogenous equation is $a_n = a_n^{(h)} + a_n^{(p)}$.

Examples

Example. Solve $a_n - 5a_{n-1} + 6a_{n-2} = 7^n$
Solution. The characteristic equation: $r^2 - 5r + 6 = 0$
Therefore $r_1 = 3, r_2 = 2 \Rightarrow a_n^{(h)} = C_1 3^n + C_2 2^n$.

Since 7 is not a root of the characteristic equation (the roots are 3 and 2) hence the trial solution is $C7^n$ (Case 1).

Substituting the trial solution into the nonhomogeneous equation we have:

$$
\begin{aligned}
&C7^n - 5C7^{n-1} + 6C7^{n-2} = 7^n \\
&C7^2 - 5 \cdot 7C + 6C = 7^2 \\
&49C - 35C + 6C = 49
\end{aligned}
$$

or $C = \frac{49}{20} \Rightarrow a_n^{(p)} = \frac{49}{20}7^n$.

Combining $a_n^{(h)}$ and $a_n^{(p)}$ to get

$a_n = C_1 3^n + C_2 2^n + \frac{49}{20}7^n$.

You can find C_1 and C_2 from the initial conditions (whatever they may be).

Big-O notation: Let $f(x)$ and $g(x)$ be two functions from the set of integers, we say
$f(x) = O(g(x))$ if there are constants C and k such that:
$|f(x)| \leq C|g(x)|$, for $x > k$.

Example: $f(x) = 2x^2 + 3x - 12$, and $g(x) = x^2$ we say $f(x) = O(x^2)$.

Example: $logn! = O(nlogn)$.

Divide-and-Conquer relations

$$T(n) = aT(\frac{n}{b}) + c.$$

Assume: $n = b^k$ or $k = log_b n$.

$$T(\frac{n}{b}) = aT(\frac{n}{b^2}) + c.$$

$$T(n) = a[aT(\frac{n}{b^2}) + c] + c$$

$$T(n) = a^2T(\frac{n}{b^2}) + ac + c.$$

$$T(n) = a^kT(\frac{n}{b^k}) + (a^{k-1} + ... + a + 1)c.$$

$$T(n) = a^kT(1) + c\sum_{i=0}^{k-1} a^i.$$

Assume $T(1) = 1$.
If $a = 1$ then

$$T(n) = T(1) + c \cdot k = 1 + c \cdot log_b n = O(logn)$$

If $a > 1$ then

$$T(n) = a^k + c \sum_{i=0}^{k-1} a^i.$$

Since

$$\sum_{i=0}^{k-1} a^i = \frac{a^k - 1}{a - 1}.$$

We have

$$T(n) = a^k[1 + \frac{c}{a-1}] - \frac{c}{a-1}.$$

Therefore

$$T(n) = O(a^k) = O(a^{log_b n}) = O(n^{log_b a}).$$

More details on Divide-and-Conquer relations

We first derive solution in general case and then infer particular cases.

Theorem 1 *Let $f(n)$ be an increasing function satisfying*

$$f(n) = af(n/b) + g(n)$$

whenever n is divisible by b, where $a \geq 1$, b is an integer greater than 1, and $g(n)$ is the positive sequence of real numbers. Then

$$f(n) = a^k f\left(\frac{n}{b^k}\right) + \sum_{i=0}^{k-1} a^i g\left(\frac{n}{b^i}\right)$$

when $n = b^k$ and

$$f(n) \leq f(b^{k+1}) = a^{k+1} f(1) + \sum_{i=0}^{k} a^i g\left(\frac{n}{b^i}\right)$$

when $n \neq b^k$.

Proof. Assume $n = b^k$. We have by iterative approach

$$
\begin{aligned}
& f(n) = af(n/b) + g(n) \\
& = a[af\left(\frac{n}{b^2}\right) + g\left(\frac{n}{b}\right)] + g(n) \\
& = a^2 f\left(\frac{n}{b^2}\right) + ag\left(\frac{n}{b}\right) + g(n) \\
& = a^2[af\left(\frac{n}{b^3}\right) + g\left(\frac{n}{b^2}\right)] + ag\left(\frac{n}{b}\right) + g(n) \\
& = a^3 f\left(\frac{n}{b^3}\right) + a^2 g\left(\frac{n}{b^2}\right) n + ag\left(\frac{n}{b}\right) + g(n) \\
& \ldots \\
& = a^k f(1) + \sum_{i=0}^{k-1} a^i g\left(\frac{n}{b^i}\right).
\end{aligned}
$$

When $n \neq b^k$ then $b^k < n < b^{k+1}$ for some positive integer k. Since f is an increasing function

$$f(n) \le f(b^{k+1}) = a^{k+1} f(1) + \sum_{i=0}^{k} a^i g\left(\frac{n}{b^i}\right).$$

Special cases.

1. $g(n) = c$.

   i) $a = 1$. Let $n = b^k$. Then

   $$f(n) = a^k f(1) + c \sum_{i=0}^{k-1} a^i$$

   thus

   $$f(n) = f(1) + ck.$$

   But $n = b^k \implies k = \log_b n$. Hence

   $$f(n) = f(1) + c \log_b n = O(\log n).$$

   When $n \neq b^k$, we have

   $$\begin{aligned} f(n) &= f(1) + c(k+1) \\ &= f(1) + c + c \log_b n = O(\log n). \end{aligned}$$

   ii) $a > 1$. Let $n = b^k$. Using formula for the sum of geometric progression

   $$\sum_{i=0}^{k} r^i = \frac{r^{k+1} - 1}{r - 1}, \quad r \neq 0$$

   we have

$$
\begin{aligned}
f(n) \quad & = f(1) + c\sum_{i=0}^{k-1} a^i \\
& = a^k f(1) + c\frac{a^k - 1}{a - 1} \\
& = a^k[f(1) + \frac{a}{a-1}] + (-\frac{c}{a-1}) \\
& = \left(f(1) + \frac{a}{a-1}\right) n^{\log_b a} + (-\frac{c}{a-1}) \\
& = C_1 n^{\log_b a} + C_2 = O(n^{\log_b a})
\end{aligned}
$$

where we have used the identity $a^{\log_b n} = n^{\log_b a}$.

When $n \neq b^k$ then

$$f(n) \leq (C_1 a) n^{\log_b a} + C_2 = O(n^{\log_b a}).$$

We have thus proved

Theorem 2 *Let $f(n)$ be an increasing function satisfying*

$$f(n) = af(n/b) + c$$

whenever n is divisible by b, where $a \geq 1$, b is an integer greater than 1, and c is a positive real number. Then

$$f(n) = \begin{cases} O(\log n) & \textit{if} \quad a = 1 \\ O(n^{\log_b a}) & \textit{if} \quad a > 1. \end{cases}$$

2. $g(n) = cn^d$, where d is a positive real number.

Let $n = b^k$. By Theorem 1 we have

$$\begin{aligned} f(n) &= a^k f(n/b^k) + \sum_{i=0}^{k-1} a^i c \left(\frac{n}{b^i}\right)^d \\ &= a^k f(1) + cn^d \sum_{i=0}^{k-1} \left(\frac{a}{b^d}\right)^i . \end{aligned}$$

When $n \neq b^k$

$$f(n) \leq f(b^{k+1}) = a^{k+1} f(1) + cn^d \sum_{i=0}^{k} \left(\frac{a}{b^d}\right)^i .$$

(i) $a = b^d$.

$$\begin{aligned}
f(n) \quad &= a^k f(1) + cn^d \sum_{i=0}^{k-1} 1^i \\
&= a^k f(1) + cn^d k \\
&= a^k f(1) + cn^d \log_b n \\
&= a^{\log_b n} f(1) + cn^d \log_b n \\
&= n^{\log_b a} f(1) + cn^d \log_b n \\
&= n^d f(1) + cn^d \log_b n \\
&= O(n^d \log n).
\end{aligned}$$

(ii) $a \neq b^d$. Then

$$\begin{aligned}
& f(n) \\
&= a^k f(1) + cn^d \sum_{i=0}^{k-1} \left(\frac{a}{b^d}\right)^i \\
&= a^k f(1) + cn^d \frac{\left(\frac{a}{b^d}\right)^k - 1}{\frac{a}{b^d} - 1} \\
&= a^k f(1) + cn^d \frac{\frac{a^k}{b^{kd}} b^d - b^d}{a - b^d} \\
&= a^k [f(1) + c\frac{n^d \frac{b^d}{b^{kd}}}{a - b^d}] - cn^d \frac{b^d}{a - b^d} \\
&= a^{\log_b n} [f(1) + c\frac{\left(\frac{n}{b^k}\right)^d b^d}{a - b^d}] - cn^d \frac{b^d}{a - b^d} \\
&= n^{\log_b a} [f(1) + c\frac{b^d}{a - b^d}] - cn^d \frac{b^d}{a - b^d} \\
&= C_1 n^{\log_b a} + C_2 n^d
\end{aligned}$$

where $C_1 = [f(1) + c\frac{b^d}{a-b^d}]$ and $C_2 = c\frac{b^d}{b^d-a}$. If $a > b^d$ then $\log_b a > d$ and the last equation implies

$$f(n) = O(n^{\log_b a})$$

otherwise if $a < b^d$ then $\log_b a < d$ and we have

$$f(n) = O(n^d).$$

If $n \neq b^k$ then

$$f(n) \leq f(b^{k+1}) = a^{k+1} f(1) + cn^d \sum_{i=0}^{k} \left(\frac{a}{b^d}\right)^i$$
$$= \begin{cases} O(n^{\log_b a}) & \text{if} \quad a > b^d \\ O(n^d) & \text{if} \quad a < b^d \end{cases}$$

whenever $a \neq b^d$ and

$$\begin{aligned} & f(n) \leq f(b^{k+1}) \\ & = a^{k+1} f(1) + cn^d (k+1) \\ & = O(n^d \log n) \end{aligned}$$

when $a = b^d$. We have thus proven the master theorem

Theorem 3 *Let $f(n)$ be an increasing function satisfying*

$$f(n) = af(n/b) + cn^d$$

whenever n is divisible by b, where $a \geq 1$, b is an integer greater than 1, and c, d are positive real numbers. Then

$$f(n) = \begin{cases} O(n^{\log_b a}) & \textit{if} \quad a > b^d \\ O(n^d \log n) & \textit{if} \quad a = b^d \\ O(n^d) & \textit{if} \quad a < b^d. \end{cases}$$

Example. Solve

$$f(n) = 8f(n/2) + n^2$$

where f is an increasing function, $f(1) = 1$ and show that $f(n) = O(n^{\log 3})$.

Solution. Following the proof Theorem 3 we have $a = 8, b = 2, c = 1, d = 2$, thus $a > b^d$ and

$$\begin{aligned} f(n) \quad &= [1 + \frac{1 \cdot 2^2}{8 - 2^2}]n^{\log_2 8} + \frac{1 \cdot 2^2}{2^2 - 8}n^2 \\ &= 2n^3 - n^2 \\ &= O(n^3) \end{aligned}$$

which agrees with Theorem 3.

Here is the complete solution.

$$\begin{aligned}
f(n) &= 8f(n/2) + n^2 \\
&= 8[8f(n/2^2) + (n/2)^2] + n^2 \\
&= 8^2 f(n/2^2) + 8(n/2)^2 + n^2 \\
&= 8^2[8f(n/2^3) + (n/2^2)^2] + 8(n/2)^2 + n^2 \\
&= 8^3 f(n/2^3) + 8^2(n/2^2)^2 + 8(n/2)^2 + n^2 \\
&= 8^k f(1) + n^2 \sum_{i=0}^{k-1} 8^i \left(\frac{1}{2^2}\right)^i \\
&= 8^k f(1) + n^2 \sum_{i=0}^{k-1} \left(\frac{8}{4}\right)^i \\
&= 8^k f(1) + n^2 \frac{2^k - 1}{2 - 1} \\
&= 8^k + n^2(2^k - 1).
\end{aligned}$$

Next, $k = \log_2 n$, $\quad 8^k = 8^{\log_2 n} = n^{\log_2 8} = n^3$. Thus

$$\begin{aligned}
f(n) &= 8^{\log_2 n} + n^2(2^{\log_2 n} - 1) \\
&= n^{\log_2 8} + n^2(n^{\log_2 2} - 1) \\
&= n^3 + n^2(n - 1) \\
&= 2n^3 - n^2.
\end{aligned}$$

Hence

$$f(n) = 2n^3 - n^2 = O(n^3).$$

Inclusion-Exclusion

Example: How many solutions are there to the equation:

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 30$$

where x_i, $i = 1, 2, 3, 4, 5, 6$, is nonnegative integer such that

1. $x_i > 1$ for $i = 1, 2, 3, 4, 5, 6$?

2. $x_1 \leq 5$?

3. $x_1 \leq 7$ and $x_2 > 8$?

Solution.

1. We require each $x_i \geq 2$. This uses up 12 of the 30 total required, so the problem is the same as finding the number of solutions to $x'_1 + x'_2 + x'_3 + x'_4 + x'_5 + x'_6 = 18$ with each $x'_i s = x_i - 1$ a nonnegative integer. The number of solutions is therefore $C(6 + 18 - 1, 18) = C(23, 18) = 33649$.

2. The number of solutions without restriction is $C(6 + 30 - 1, 30) = C(35, 30) = 324632$. The number of solutions violating the restriction by having $x_1 \geq 6$ is $C(6 + 24 - 1, 24) = C(29, 24) = 118755$. Therefore the answer is 324632 - 118755 = 205877.

3. The number of solutions with $x_2 \geq 9$ (as required) but without the restriction on x_1 is $C(6+21-1, 21) = C(26, 21) = 65780$. The number of solution violating the additional restriction by having $x_1 \geq 8$ is $C(6+13-1, 13) = C(18, 13) = 8568$. Therefore the answer is 65780 - 8568 = 57212.

Example: How many solutions are there to the equation:

$$x_1 + x_2 + x_3 = 13$$

where x_i, $i = 1, 2, 3$ are nonnegative integer such that

$$1 \leq x_1 \leq 4, x_2 \leq 6, x_3 \leq 9?$$

Solution. First we take care of the double constraint on x_1 by substitution $x_1 = p + 1$, where $0 \leq p \leq 3$. The original problem is equivalent to finding solutions to

$$p + d + h = 12$$

where p, d, h are nonnegative integers such that

$$p \leq 3, d \leq 6, h \leq 9?$$

Let S denote the set of all nonnegative integer solutions (p, d, h) of $p + d + h = 12$;

let P denote the set of all (p, d, h) in S such that $p \geq 4$;

let D denote the set of all (p, d, h) in S such that $d \geq 7$;

let H denote the set of all (p, d, h) in S such that $h \geq 10$.

By the principle of inclusion-exclusion, we have

$$\begin{aligned}
&|S-(P\cup D\cup H)| \\
&= |S|-(|P|+|D|+|H|) \\
&+(|P\cap D|+|P\cap H|+|D\cap H|) \\
&-(|P\cap D\cap H|) \qquad (1)
\end{aligned}$$

we also have $|S| = \binom{14}{2} = 91$ and
$|P| = \binom{10}{2} = 45$ (the number of nonnegative integer solutions of $p' + d + h = 8$),
$|D| = \binom{7}{2} = 21$ (the number of nonnegative integer solutions of $p + d' + h = 5$),
$|H| = \binom{4}{2} = 6$ (the number of nonnegative integer solutions of $p + d + h' = 2$),
$|P \cap D| = \binom{3}{2} = 3$ (the number of nonnegative integer solutions of $p' + d' + h = 1$),
and $|P \cap H| = |D \cap H| = |P \cap D \cap H| = 0$.
Substituting these partial results to (1) we get the answer 22.

Example: A well-known result implies that a composite integer is divisible by a prime not exceeding its square root. Find a number of primes not exceeding 100.

Solution.

The only primes less than 10 are 2,3,5,7, so the primes not exceeding 100 are these four primes and all positive integers $1 < n \leq 100$ not divisible by 2,3,5,7. Let A_i be a subset of elements that have property P_i that an integer is divisible by $i, i = 2, 3, 5, 7$ and let $|A_i| = N(P_i)$. By the principle of inclusion-exclusion the answer is

$$\begin{aligned}
&4 + N(P_2'P_3'P_5'P_7') \\
&= 4 + (99 - N(P_2) - N(P_3) - N(P_5) - N(P_7) \\
&+N(P_2P_3) + N(P_2P_5) + N(P_2P_7) \\
&+N(P_3P_5) + N(P_3P_7) + N(P_5P_7) \\
&-N(P_2P_3P_5) - N(P_2P_3P_7) - N(P_2P_5P_7) - N(P_3P_3P_7) \\
&+N(P_2P_3P_5P_7)) \\
&= 4 + (99 - \left\lfloor \frac{100}{2} \right\rfloor - \left\lfloor \frac{100}{3} \right\rfloor - \left\lfloor \frac{100}{5} \right\rfloor - \left\lfloor \frac{100}{7} \right\rfloor \\
&+ \left\lfloor \frac{100}{2 \cdot 3} \right\rfloor + \left\lfloor \frac{100}{2 \cdot 5} \right\rfloor + \left\lfloor \frac{100}{2 \cdot 7} \right\rfloor + \left\lfloor \frac{100}{3 \cdot 5} \right\rfloor + \left\lfloor \frac{100}{3 \cdot 7} \right\rfloor \\
&- \left\lfloor \frac{100}{2 \cdot 3 \cdot 5} \right\rfloor - \left\lfloor \frac{100}{2 \cdot 3 \cdot 7} \right\rfloor - \left\lfloor \frac{100}{2 \cdot 5 \cdot 7} \right\rfloor - \left\lfloor \frac{100}{3 \cdot 5 \cdot 7} \right\rfloor \\
&+ \left\lfloor \frac{100}{2 \cdot 3 \cdot 5 \cdot 7} \right\rfloor) \\
&= 4 + (99 - 50 - 33 - 20 - 14 \\
&+16 + 10 + 7 + 6 + 4 + 2 - 3 - 2 - 1 - 0 + 0) \\
&= 25.
\end{aligned}$$

5 Graph Theory

Informally, a graph is a bunch of dots and lines where the lines connect some pairs of dots. An example is shown in Figure 5.1. The dots are called *nodes* (or *vertices*) and the lines are called *edges*.

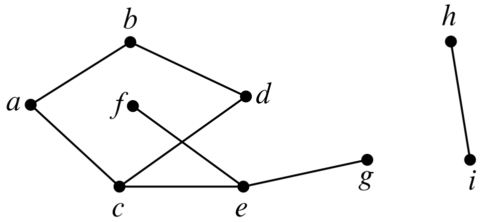


Figure 5.1 An example of a graph with 9 nodes and 8 edges.

Graphs are ubiquitous in computer science because they provide a handy way to represent a relationship between pairs of objects. The objects represent items of interest such as programs, people, cities, or web pages, and we place an edge between a pair of nodes if they are related in a certain way. For example, an edge between a pair of people might indicate that they like (or, in alternate scenarios, that they don't like) each other. An edge between a pair of courses might indicate that one needs to be taken before the other.

In this chapter, we will focus our attention on simple graphs where the relationship denoted by an edge is symmetric. Afterward, in Chapter 6, we consider the situation where the edge denotes a one-way relationship, for example, where one web page points to the other.[1]

5.1 Definitions

5.1.1 Simple Graphs

Definition 5.1.1. A *simple graph* G consists of a nonempty set V, called the *vertices* (aka *nodes*[2]) of G, and a set E of two-element subsets of V. The members of E are called the *edges* of G, and we write $G = (V, E)$.

[1] Two Stanford students analyzed such a graph to become multibillionaires. So, pay attention to graph theory, and who knows what might happen!

[2] We will use the terms vertex and node interchangeably.

The vertices correspond to the dots in Figure 5.1, and the edges correspond to the lines. The graph in Figure 5.1 is expressed mathematically as $G = (V, E)$, where:

$$V = \{a, b, c, d, e, f, g, h, i\}$$
$$E = \{\, \{a, b\}, \{a, c\}, \{b, d\}, \{c, d\}, \{c, e\}, \{e, f\}, \{e, g\}, \{h, i\}\,\}.$$

Note that $\{a, b\}$ and $\{b, a\}$ are different descriptions of the same edge, since sets are unordered. In this case, the graph $G = (V, E)$ has 9 nodes and 8 edges.

Definition 5.1.2. Two vertices in a simple graph are said to be *adjacent* if they are joined by an edge, and an edge is said to be *incident* to the vertices it joins. The number of edges incident to a vertex v is called the *degree* of the vertex and is denoted by $\deg(v)$; equivalently, the degree of a vertex is equals the number of vertices adjacent to it.

For example, in the simple graph shown in Figure 5.1, vertex a is adjacent to b and b is adjacent to d, and the edge $\{a, c\}$ is incident to vertices a and c. Vertex h has degree 1, d has degree 2, and $\deg(e) = 3$. It is possible for a vertex to have degree 0, in which case it is not adjacent to any other vertices. A simple graph does not need to have any edges at all —in which case, the degree of every vertex is zero and $|E| = 0$[3] —but it does need to have at least one vertex, that is, $|V| \geq 1$.

Note that simple graphs do *not* have any *self-loops* (that is, an edge of the form $\{a, a\}$) since an edge is defined to be a set of *two* vertices. In addition, there is at most one edge between any pair of vertices in a simple graph. In other words, a simple graph does not contain *multiedges* or *multiple edges*. That is because E is a set. Lastly, and most importantly, simple graphs do not contain *directed edges* (that is, edges of the form (a, b) instead of $\{a, b\}$).

There's no harm in relaxing these conditions, and some authors do, but we don't need self-loops, multiple edges between the same two vertices, or graphs with no vertices, and it's simpler not to have them around. We will consider graphs with directed edges (called *directed graphs* or *digraphs*) at length in Chapter 6. Since we'll only be considering simple graphs in this chapter, we'll just call them "graphs" from now on.

5.1.2 Some Common Graphs

Some graphs come up so frequently that they have names. The *complete graph* on n vertices, denoted K_n, has an edge between every two vertices, for a total of $n(n-1)/2$ edges. For example, K_5 is shown in Figure 5.2.

The *empty graph* has no edges at all. For example, the empty graph with 5 nodes is shown in Figure 5.3.

[3] The *cardinality*, $|E|$, of the set E is the number of elements in E.

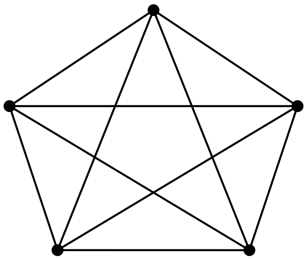


Figure 5.2 The complete graph on 5 nodes, K_5.

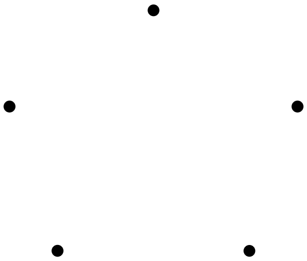


Figure 5.3 The empty graph with 5 nodes.

The n-node graph containing $n - 1$ edges in sequence is known as the *line graph* L_n. More formally, $L_n = (V, E)$ where

$$V = \{v_1, v_2, \ldots, v_n\}$$

and

$$E = \{\,\{v_1, v_2\}, \{v_2, v_3\}, \ldots, \{v_{n-1}, v_n\}\,\}$$

For example, L_5 is displayed in Figure 5.4.

If we add the edge $\{v_n, v_1\}$ to the line graph L_n, we get the graph C_n consisting of a simple cycle. For example, C_5 is illustrated in Figure 5.5.

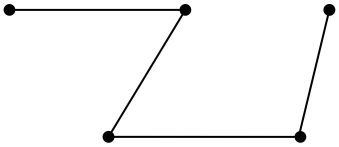


Figure 5.4 The 5-node line graph L_5.

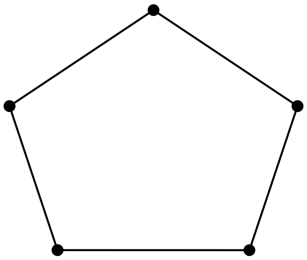


Figure 5.5 The 5-node cycle graph C_5.

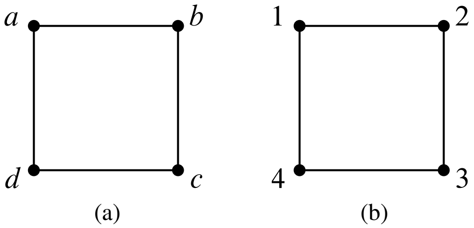


Figure 5.6 Two graphs that are isomorphic to C_4.

5.1.3 Isomorphism

Two graphs that look the same might actually be different in a formal sense. For example, the two graphs in Figure 5.6 are both simple cycles with 4 vertices, but one graph has vertex set $\{a, b, c, d\}$ while the other has vertex set $\{1, 2, 3, 4\}$. Strictly speaking, these graphs are different mathematical objects, but this is a frustrating distinction since the graphs *look the same*!

Fortunately, we can neatly capture the idea of "looks the same" through the notion of graph isomorphism.

Definition 5.1.3. If $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are two graphs, then we say that G_1 is *isomorphic* to G_2 iff there exists a *bijection*[4] $f : V_1 \to V_2$ such that for every pair of vertices $u, v \in V_1$:

$$\{u, v\} \in E_1 \quad \text{iff} \quad \{f(u), f(v)\} \in E_2.$$

The function f is called an *isomorphism* between G_1 and G_2.

In other words, two graphs are isomorphic if they are the same up to a relabeling of their vertices. For example, here is an isomorphism between vertices in the two

[4] A bijection $f : V_1 \to V_2$ is a function that associates every node in V_1 with a unique node in V_2 and vice-versa. We will study bijections more deeply in Part III.

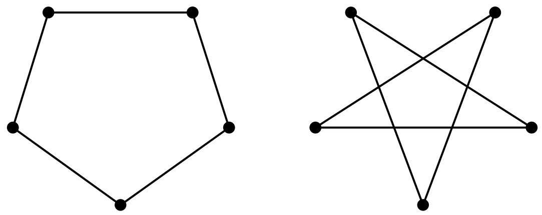


Figure 5.7 Two ways of drawing C_5.

graphs shown in Figure 5.6:

a corresponds to 1 $\quad$ b corresponds to 2
d corresponds to 4 $\quad$ c corresponds to 3.

You can check that there is an edge between two vertices in the graph on the left if and only if there is an edge between the two corresponding vertices in the graph on the right.

Two isomorphic graphs may be drawn very differently. For example, we have shown two different ways of drawing C_5 in Figure 5.7.

Isomorphism preserves the connection properties of a graph, abstracting out what the vertices are called, what they are made out of, or where they appear in a drawing of the graph. More precisely, a property of a graph is said to be *preserved under isomorphism* if whenever G has that property, every graph isomorphic to G also has that property. For example, isomorphic graphs must have the same number of vertices. What's more, if f is a graph isomorphism that maps a vertex, v, of one graph to the vertex, $f(v)$, of an isomorphic graph, then by definition of isomorphism, every vertex adjacent to v in the first graph will be mapped by f to a vertex adjacent to $f(v)$ in the isomorphic graph. This means that v and $f(v)$ will have the same degree. So if one graph has a vertex of degree 4 and another does not, then they can't be isomorphic. In fact, they can't be isomorphic if the number of degree 4 vertices in each of the graphs is not the same.

Looking for preserved properties can make it easy to determine that two graphs are not isomorphic, or to actually find an isomorphism between them if there is one. In practice, it's frequently easy to decide whether two graphs are isomorphic. However, no one has yet found a *general* procedure for determining whether two graphs are isomorphic that is *guaranteed* to run in polynomial time[5] in $|V|$.

Having such a procedure would be useful. For example, it would make it easy to search for a particular molecule in a database given the molecular bonds. On

[5] *I.e.*, in an amount of time that is upper-bounded by $|V|^c$ where c is a fixed number independent of $|V|$.

the other hand, knowing there is no such efficient procedure would also be valuable: secure protocols for encryption and remote authentication can be built on the hypothesis that graph isomorphism is computationally exhausting.

5.1.4 Subgraphs

Definition 5.1.4. A graph $G_1 = (V_1, E_1)$ is said to be a *subgraph* of a graph $G_2 = (V_2, E_2)$ if $V_1 \subseteq V_2$ and $E_1 \subseteq E_2$.

For example, the empty graph on n nodes is a subgraph of L_n, L_n is a subgraph of C_n, and C_n is a subgraph of K_n. Also, the graph $G = (V, E)$ where

$$V = \{g, h, i\} \quad \text{and} \quad E = \{\{h, i\}\}$$

is a subgraph of the graph in Figure 5.1. On the other hand, any graph containing an edge $\{g, h\}$ would not be a subgraph of the graph in Figure 5.1 because the graph in Figure 5.1 does not contain this edge.

Note that since a subgraph is itself a graph, the endpoints of any edge in a subgraph must also be in the subgraph. In other words if $G' = (V', E')$ is a subgraph of some graph G, and $\{v_i, v_j\} \in E'$, then it must be the case that $v_i \in V'$ and $v_j \in V'$.

5.1.5 Weighted Graphs

Sometimes, we will use edges to denote a connection between a pair of nodes where the connection has a *capacity* or *weight*. For example, we might be interested in the capacity of an Internet fiber between a pair of computers, the resistance of a wire between a pair of terminals, the tension of a spring connecting a pair of devices in a dynamical system, the tension of a bond between a pair of atoms in a molecule, or the distance of a highway between a pair of cities.

In such cases, it is useful to represent the system with an *edge-weighted* graph (aka a *weighted graph*). A weighted graph is the same as a simple graph except that we associate a real number (that is, the weight) with each edge in the graph. Mathematically speaking, a weighted graph consists of a graph $G = (V, E)$ and a weight function $w : E \to \mathbb{R}$. For example, Figure 5.8 shows a weighted graph where the weight of edge $\{a, b\}$ is 5.

5.1.6 Adjacency Matrices

There are many ways to represent a graph. We have already seen two ways: you can draw it, as in Figure 5.8 for example, or you can represent it with sets —as in $G = (V, E)$. Another common representation is with an adjacency matrix.

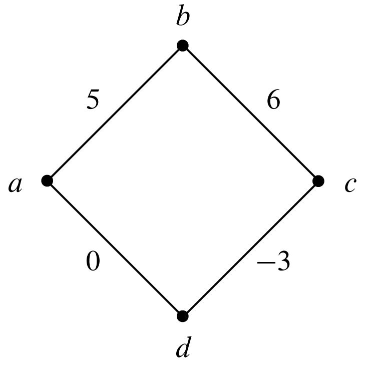


Figure 5.8 A 4-node weighted graph where the edge $\{a, b\}$ has weight 5.

$$\begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \qquad \begin{pmatrix} 0 & 5 & 0 & 0 \\ 5 & 0 & 6 & 0 \\ 0 & 6 & 0 & -3 \\ 0 & 0 & -3 & 0 \end{pmatrix}$$

(a) (b)

Figure 5.9 Examples of adjacency matrices. (a) shows the adjacency matrix for the graph in Figure 5.6(a) and (b) shows the adjacency matrix for the weighted graph in Figure 5.8. In each case, we set $v_1 = a$, $v_2 = b$, $v_3 = c$, and $v_4 = d$ to construct the matrix.

Definition 5.1.5. Given an n-node graph $G = (V, E)$ where $V = \{v_1, v_2, \ldots, v_n\}$, the *adjacency matrix* for G is the $n \times n$ matrix $A_G = \{a_{ij}\}$ where

$$a_{ij} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \in E \\ 0 & \text{otherwise.} \end{cases}$$

If G is a weighted graph with edge weights given by $w : E \to \mathbb{R}$, then the adjacency matrix for G is $A_G = \{a_{ij}\}$ where

$$a_{ij} = \begin{cases} w(\{v_i, v_j\}) & \text{if } \{v_i, v_j\} \in E \\ 0 & \text{otherwise.} \end{cases}$$

For example, Figure 5.9 displays the adjacency matrices for the graphs shown in Figures 5.6(a) and 5.8 where $v_1 = a$, $v_2 = b$, $v_3 = c$, and $v_4 = d$.

5.2 Matching Problems

We begin our study of graph theory by considering the scenario where the nodes in a graph represent people and the edges represent a relationship between pairs of people such as “likes”, “marries”, and so on. Now, you may be wondering what marriage has to do with computer science, and with good reason. It turns out that the techniques we will develop apply to much more general scenarios where instead of matching men to women, we need to match packets to paths in a network, applicants to jobs, or Internet traffic to web servers. And, as we will describe later, these techniques are widely used in practice.

In our first example, we will show how graph theory can be used to debunk an urban legend about sexual practices in America. Yes, you read correctly. So, fasten your seat belt—who knew that math might actually be interesting!

5.2.1 Sex in America

On average, who has more opposite-gender partners: men or women?

Sexual demographics have been the subject of many studies. In one of the largest, researchers from the University of Chicago interviewed a random sample of 2500 Americans over several years to try to get an answer to this question. Their study, published in 1994, and entitled *The Social Organization of Sexuality* found that, on average, men have 74% more opposite-gender partners than women.

Other studies have found that the disparity is even larger. In particular, ABC News claimed that the average man has 20 partners over his lifetime, and the average woman has 6, for a percentage disparity of 233%. The ABC News study, aired on Primetime Live in 2004, purported to be one of the most scientific ever done, with only a 2.5% margin of error. It was called “American Sex Survey: A peek between the sheets.” The promotion for the study is even better:

> A ground breaking ABC News “Primetime Live” survey finds a range of eye-popping sexual activities, fantasies and attitudes in this country, confirming some conventional wisdom, exploding some myths—and venturing where few scientific surveys have gone before.

Probably that last part about going where few scientific surveys have gone before is pretty accurate!

Yet again, in August, 2007, the N.Y. Times reported on a study by the National Center for Health Statistics of the U.S. Government showing that men had seven partners while women had four.

Anyway, whose numbers do you think are more accurate, the University of Chicago, ABC News, or the National Center for Health Statistics?—don't answer; this is a setup question like "When did you stop beating your wife?" Using a little graph theory, we will now explain why none of these findings can be anywhere near the truth.

Let's model the question of heterosexual partners in graph theoretic terms. To do this, we'll let G be the graph whose vertices, V, are all the people in America. Then we split V into two separate subsets: M, which contains all the males, and F, which contains all the females.[6] We'll put an edge between a male and a female iff they have been sexual partners. A possible subgraph of this graph is illustrated in Figure 5.10 with males on the left and females on the right.

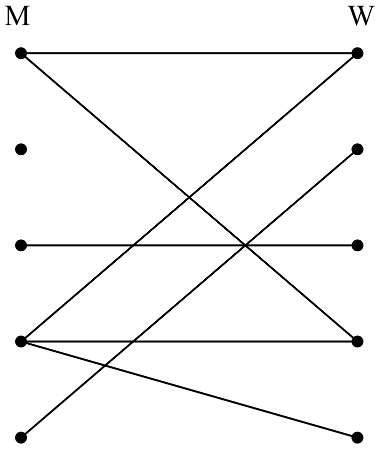


Figure 5.10 A possible subgraph of the sex partners graph.

Actually, G is a pretty hard graph to figure out, let alone draw. The graph is *enormous*: the US population is about 300 million, so $|V| \approx 300M$. In the United States, approximately 50.8% of the populatin is female and 49.2% is male, and so $|M| \approx 147.6M$, and $|F| \approx 152.4M$. And we don't even have trustworthy estimates of how many edges there are, let alone exactly which couples are adjacent. But it turns out that we don't need to know any of this to debunk the sex surveys—we just need to figure out the relationship between the average number of partners per male and partners per female. To do this, we note that every edge is incident to exactly one M vertex and one F vertex (remember, we're only considering male-female relationships); so the sum of the degrees of the M vertices equals the number of edges, and the sum of the degrees of the F vertices equals the

[6] For simplicity, we'll ignore the possibility of someone being both, or neither, a man and a woman.

number of edges. So these sums are equal:

$$\sum_{x \in M} \deg(x) = \sum_{y \in F} \deg(y).$$

If we divide both sides of this equation by the product of the sizes of the two sets, $|M| \cdot |F|$, we obtain

$$\left(\frac{\sum_{x \in M} \deg(x)}{|M|} \right) \cdot \frac{1}{|F|} = \left(\frac{\sum_{y \in F} \deg(y)}{|F|} \right) \cdot \frac{1}{|M|} \tag{5.1}$$

Notice that

$$\frac{\sum_{x \in M} \deg(x)}{|M|}$$

is simply the average degree of a node in M. This is the average number of opposite-gender partners for a male in America. Similarly,

$$\frac{\sum_{x \in F} \deg(x)}{|F|}$$

is the average degree of a node in F, which is the average number of opposite-gender partners for a female in America. Hence, Equation 5.1 implies that on average, an American male has $|F|/|M|$ times as many opposite-gender partners as the average American female.

From the Census Bureau reports, we know that there are slightly more females than males in America; in particular $|F|/|M|$ is about 1.035. So we know that on average, males have 3.5% more opposite-gender partners than females. Of course, this statistic really says nothing about any sex's promiscuity or selectivity. Remarkably, promiscuity is completely irrelevant in this analysis. That is because the ratio of the average number of partners is completely determined by the relative number of males and females. Collectively, males and females have the same number of opposite gender partners, since it takes one of each set for every partnership, but there are fewer males, so they have a higher ratio. This means that the University of Chicago, ABC, and the Federal Government studies are way off. After a huge effort, they gave a totally wrong answer.

There's no definite explanation for why such surveys are consistently wrong. One hypothesis is that males exaggerate their number of partners—or maybe females downplay theirs—but these explanations are speculative. Interestingly, the principal author of the National Center for Health Statistics study reported that she knew the results had to be wrong, but that was the data collected, and her job was to report it.

The same underlying issue has led to serious misinterpretations of other survey data. For example, a few years ago, the Boston Globe ran a story on a survey of the study habits of students on Boston area campuses. Their survey showed that on average, minority students tended to study with non-minority students more than the other way around. They went on at great length to explain why this “remarkable phenomenon” might be true. But it’s not remarkable at all—using our graph theory formulation, we can see that all it says is that there are fewer minority students than non-minority students, which is, of course what “minority” means.

The Handshaking Lemma

The previous argument hinged on the connection between a sum of degrees and the number edges. There is a simple connection between these quantities in any graph:

Lemma 5.2.1 (The Handshaking Lemma)**.** *The sum of the degrees of the vertices in a graph equals twice the number of edges.*

Proof. Every edge contributes two to the sum of the degrees, one for each of its endpoints. ■

Lemma 5.2.1 is called the *Handshake Lemma* because if we total up the number of people each person at a party shakes hands with, the total will be twice the number of handshakes that occurred.

5.2.2 Bipartite Matchings

There were two kinds of vertices in the “Sex in America” graph—males and females, and edges only went between the two kinds. Graphs like this come up so frequently that they have earned a special name—they are called *bipartite graphs*.

Definition 5.2.2. A *bipartite graph* is a graph together with a partition of its vertices into two sets, L and R, such that every edge is incident to a vertex in L and to a vertex in R.

The bipartite matching problem is related to the sex-in-America problem that we just studied; only now the goal is to get everyone happily married. As you might imagine, this is not possible for a variety of reasons, not the least of which is the fact that there are more women in America than men. So, it is simply not possible to marry every woman to a man so that every man is married only once.

But what about getting a mate for every man so that every woman is married only once? Is it possible to do this so that each man is paired with a woman that he likes? The answer, of course, depends on the bipartite graph that represents who

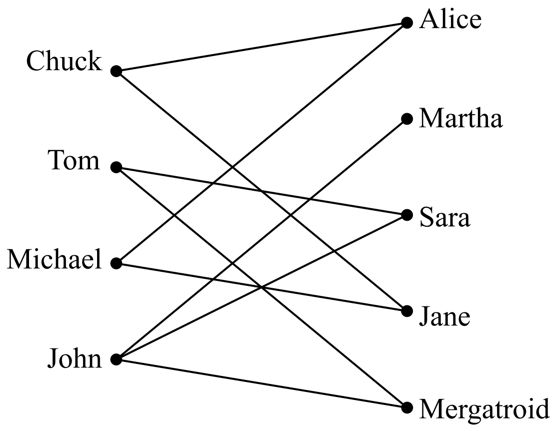


Figure 5.11 A graph where an edge between a man and woman denotes that the man likes the woman.

likes who, but the good news is that it is possible to find natural properties of the who-likes-who graph that completely determine the answer to this question.

In general, suppose that we have a set of men and an equal-sized or larger set of women, and there is a graph with an edge between a man and a woman if the man likes the woman. Note that in this scenario, the "likes" relationship need not be symmetric, since for the time being, we will only worry about finding a mate for each man that he likes.[7] (Later, we will consider the "likes" relationship from the female perspective as well.) For example, we might obtain the graph in Figure 5.11.

In this problem, a *matching* will mean a way of assigning every man to a woman so that different men are assigned to different women, and a man is always assigned to a woman that he likes. For example, one possible matching for the men is shown in Figure 5.12.

The Matching Condition

A famous result known as Hall's Matching Theorem gives necessary and sufficient conditions for the existence of a matching in a bipartite graph. It turns out to be a remarkably useful mathematical tool.

We'll state and prove Hall's Theorem using man-likes-woman terminology. Define *the set of women liked by a given set of men* to consist of all women liked by at least one of those men. For example, the set of women liked by Tom and John in

[7] By the way, we do not mean to imply that marriage should or should not be of a heterosexual nature. Nor do we mean to imply that men should get their choice instead of women. It's just that with bipartite graphs, the edges only connected male nodes to female nodes and there are fewer men in America. So please don't take offense.

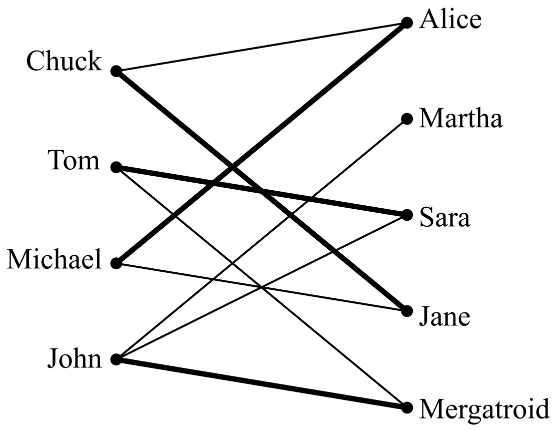


Figure 5.12 One possible matching for the men is shown with bold edges. For example, John is matched with Jane.

Figure 5.11 consists of Martha, Sarah, and Mergatroid. For us to have any chance at all of matching up the men, the following *matching condition* must hold:

Every subset of men likes at least as large a set of women.

For example, we can not find a matching if some set of 4 men like only 3 women. Hall's Theorem says that this necessary condition is actually sufficient; if the matching condition holds, then a matching exists.

Theorem 5.2.3. *A matching for a set of men M with a set of women W can be found if and only if the matching condition holds.*

Proof. First, let's suppose that a matching exists and show that the matching condition holds. Consider an arbitrary subset of men. Each man likes at least the woman he is matched with. Therefore, every subset of men likes at least as large a set of women. Thus, the matching condition holds.

Next, let's suppose that the matching condition holds and show that a matching exists. We use strong induction on $|M|$, the number of men, on the predicate:

$$\begin{aligned} P(m) ::= {} & \text{for any set of } m \text{ men } M \text{, if the matching condition holds} \\ & \text{for } M \text{, then there is a matching for } M. \end{aligned}$$

Base Case ($|M| = 1$): If $|M| = 1$, then the matching condition implies that the lone man likes at least one woman, and so a matching exists.

Inductive Step: We need to show that $P(m)$ IMPLIES $P(m + 1)$. Suppose that $|M| = m + 1 \geq 2$.

Case 1: Every proper subset[8] of men likes a *strictly larger* set of women. In this case, we have some latitude: we pair an arbitrary man with a woman he likes and send them both away. The matching condition still holds for the remaining men and women since we have removed only one woman, so we can match the rest of the men by induction.

Case 2: Some proper subset of men $X \subset M$ likes an *equal-size* set of women $Y \subset W$. We match the men in X with the women in Y by induction and send them all away. We can also match the rest of the men by induction if we show that the matching condition holds for the remaining men and women. To check the matching condition for the remaining people, consider an arbitrary subset of the remaining men $X' \subseteq (M - X)$, and let Y' be the set of remaining women that they like. We must show that $|X'| \leq |Y'|$. Originally, the combined set of men $X \cup X'$ liked the set of women $Y \cup Y'$. So, by the matching condition, we know:

$$|X \cup X'| \leq |Y \cup Y'|$$

We sent away $|X|$ men from the set on the left (leaving X') and sent away an equal number of women from the set on the right (leaving Y'). Therefore, it must be that $|X'| \leq |Y'|$ as claimed.

So in both cases, there is a matching for the men, which completes the proof of the Inductive step. The theorem follows by induction. ■

The proof of Theorem 5.2.3 gives an algorithm for finding a matching in a bipartite graph, albeit not a very efficient one. However, efficient algorithms for finding a matching in a bipartite graph do exist. Thus, if a problem can be reduced to finding a matching, the problem can be solved from a computational perspective.

A Formal Statement

Let's restate Theorem 5.2.3 in abstract terms so that you'll not always be condemned to saying, "Now this group of men likes at least as many women..."

Definition 5.2.4. A *matching* in a graph, G, is a set of edges such that no two edges in the set share a vertex. A matching is said to *cover* a set, L, of vertices iff each vertex in L has an edge of the matching incident to it. A matching is said to be *perfect* if every node in the graph is incident to an edge in the matching. In any graph, the set $N(S)$, of *neighbors* of some set, S, of vertices is the set of all vertices adjacent to some vertex in S. That is,

$$N(S) ::= \{\, r \mid \{s, r\} \text{ is an edge for some } s \in S \,\}.$$

[8] Recall that a subset A of B is *proper* if $A \neq B$.

S is called a *bottleneck* if

$$|S| > |N(S)|.$$

Theorem 5.2.5 (Hall's Theorem)**.** *Let G be a bipartite graph with vertex partition L, R. There is matching in G that covers L iff no subset of L is a bottleneck.*

An Easy Matching Condition

The bipartite matching condition requires that *every* subset of men has a certain property. In general, verifying that every subset has some property, even if it's easy to check any particular subset for the property, quickly becomes overwhelming because the number of subsets of even relatively small sets is enormous—over a billion subsets for a set of size 30. However, there is a simple property of vertex degrees in a bipartite graph that guarantees the existence of a matching. Namely, call a bipartite graph *degree-constrained* if vertex degrees on the left are at least as large as those on the right. More precisely,

Definition 5.2.6. A bipartite graph G with vertex partition L, R where $|L| \leq |R|$ is *degree-constrained* if $\deg(l) \geq \deg(r)$ for every $l \in L$ and $r \in R$.

For example, the graph in Figure 5.11 is degree constrained since every node on the left is adjacent to at least two nodes on the right while every node on the right is incident to at most two nodes on the left.

Theorem 5.2.7. *Let G be a bipartite graph with vertex partition L, R where $|L| \leq |R|$. If G is degree-constrained, then there is a matching that covers L.*

Proof. The proof is by contradiction. Suppose that G is degree constrained but that there is no matching that covers L. By Theorem 5.2.5, this means that there must be a bottleneck $S \subseteq L$.

Let d be a value such that $\deg(l) \geq x \geq \deg(r)$ for every $l \in L$ and $r \in R$. Since every edge incident to a node in S is incident to a node in $N(S)$, we know that

$$|N(S)|x \geq |S|x$$

and thus that

$$|N(S)| \geq |S|.$$

This means that S is not a bottleneck, which is a contradiction. Hence G has a matching that covers L. ■

Regular graphs provide a large class of graphs that often arise in practice that are degree constrained. Hence, we can use Theorem 5.2.7 to prove that every regular bipartite graph has a perfect matching. This turns out to be a surprisingly useful result in computer science

Definition 5.2.8. A graph is said to be *regular* if every node has the same degree.

Theorem 5.2.9. *Every regular bipartite graph has a perfect matching.*

Proof. Let G be a regular bipartite graph with vertex partition L, R where $|L| \leq |R|$. Since regular graphs are degree-constrained, we know by Theorem 5.2.7 that there must be a matching in G that covers L. Since G is regular, we also know that $|L| = |R|$ and thus the matching must also cover R. This means that every node in G is incident to an edge in the matching and thus G has a perfect matching. ■

5.2.3 The Stable Marriage Problem

We next consider a version of the bipartite matching problem where there are an equal number of men and women, and where each person has preferences about who they would like to marry. In fact, we assume that each man has a complete list of all the women ranked according to his preferences, with no ties. Likewise, each woman has a ranked list of all of the men.

The preferences don’t have to be symmetric. That is, Jennifer might like Brad best, but Brad doesn’t necessarily like Jennifer best. The goal is to marry everyone: every man must marry exactly one woman and vice-versa—no polygamy. Moreover, we would like to find a matching between men and women that is *stable* in the sense that there is no pair of people that prefer each other to their spouses.

For example, suppose *every* man likes Angelina best, and every woman likes Brad best, but Brad and Angelina are married to other people, say Jennifer and Billy Bob. Now *Brad and Angelina prefer each other to their spouses*, which puts their marriages at risk: pretty soon, they’re likely to start spending late nights together working on problem sets!

This unfortunate situation is illustrated in Figure 5.13, where the digits “1” and “2” near a man shows which of the two women he ranks first second, respectively, and similarly for the women.

More generally, in any matching, a man and woman who are not married to each other and who like each other better than their spouses, is called a *rogue couple*. In the situation shown in Figure 5.13, Brad and Angelina would be a rogue couple.

Having a rogue couple is not a good thing, since it threatens the stability of the marriages. On the other hand, if there are no rogue couples, then for any man and woman who are not married to each other, at least one likes their spouse better than the other, and so they won’t be tempted to start an affair.

Definition 5.2.10. A *stable matching* is a matching with no rogue couples.

The question is, given everybody’s preferences, how do you find a stable set of marriages? In the example consisting solely of the four people in Figure 5.13, we

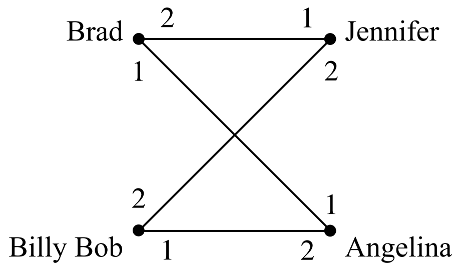


Figure 5.13 Preferences for four people. Both men like Angelina best and both women like Brad best.

could let Brad and Angelina both have their first choices by marrying each other. Now neither Brad nor Angelina prefers anybody else to their spouse, so neither will be in a rogue couple. This leaves Jen not-so-happily married to Billy Bob, but neither Jen nor Billy Bob can entice somebody else to marry them, and so there is a stable matching.

Surprisingly, there is always a stable matching among a group of men and women. The surprise springs in part from considering the apparently similar "buddy" matching problem. That is, if people can be paired off as buddies, regardless of gender, then a stable matching *may not* be possible. For example, Figure 5.14 shows a situation with a love triangle and a fourth person who is everyone's last choice. In this figure Mergatroid's preferences aren't shown because they don't even matter. Let's see why there is no stable matching.

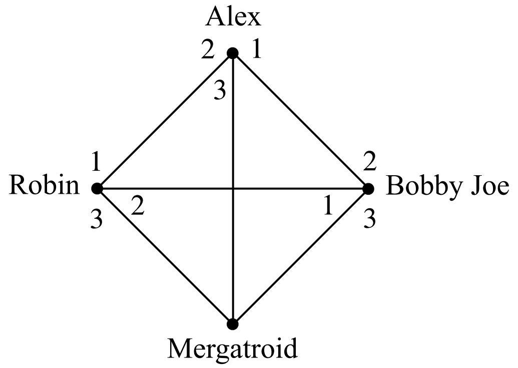


Figure 5.14 Some preferences with no stable buddy matching.

Lemma 5.2.11. *There is no stable buddy matching among the four people in Figure 5.14.*

Proof. We'll prove this by contradiction.

Assume, for the purposes of contradiction, that there is a stable matching. Then there are two members of the love triangle that are matched. Since preferences in the triangle are symmetric, we may assume in particular, that Robin and Alex are matched. Then the other pair must be Bobby-Joe matched with Mergatroid.

But then there is a rogue couple: Alex likes Bobby-Joe best, and Bobby-Joe prefers Alex to his buddy Mergatroid. That is, Alex and Bobby-Joe are a rogue couple, contradicting the assumed stability of the matching. ■

So getting a stable *buddy* matching may not only be hard, it may be impossible. But when mens are only allowed to marry women, and vice versa, then it turns out that a stable matching can always be found.[9]

The Mating Ritual

The procedure for finding a stable matching involves a *Mating Ritual* that takes place over several days. The following events happen each day:

Morning: Each woman stands on her balcony. Each man stands under the balcony of his favorite among the women on his list, and he serenades her. If a man has no women left on his list, he stays home and does his math homework.

Afternoon: Each woman who has one or more suitors serenading her, says to her favorite among them, "We might get engaged. Come back tomorrow." To the other suitors, she says, "No. I will never marry you! Take a hike!"

Evening: Any man who is told by a woman to take a hike, crosses that woman off his list.

Termination condition: When a day arrives in which every woman has at most one suitor, the ritual ends with each woman marrying her suitor, if she has one.

There are a number of facts about this Mating Ritual that we would like to prove:

- The Ritual eventually reaches the termination condition.
- Everybody ends up married.
- The resulting marriages are stable.

There is a Marriage Day

It's easy to see why the Mating Ritual has a terminal day when people finally get married. Every day on which the ritual hasn't terminated, at least one man crosses a woman off his list. (If the ritual hasn't terminated, there must be some woman serenaded by at least two men, and at least one of them will have to cross her off his

[9] Once again, we disclaim any political statement here—its just the way that the math works out.

list). If we start with n men and n women, then each of the n men's lists initially has n women on it, for a total of n^2 list entries. Since no women ever gets added to a list, the total number of entries on the lists decreases every day that the Ritual continues, and so the Ritual can continue for at most n^2 days.

They All Live Happily Every After...

We still have to prove that the Mating Ritual leaves everyone in a stable marriage. To do this, we note one very useful fact about the Ritual: if a woman has a favorite suitor on some morning of the Ritual, then that favorite suitor will still be serenading her the next morning—because his list won't have changed. So she is sure to have today's favorite man among her suitors tomorrow. That means she will be able to choose a favorite suitor tomorrow who is at least as desirable to her as today's favorite. So day by day, her favorite suitor can stay the same or get better, never worse. This sounds like an invariant, and it is.

Definition 5.2.12. Let P be the predicate: For every woman, w, and every man, m, if w is crossed off m's list, then w has a suitor whom she prefers over m.

Lemma 5.2.13. *P is an invariant for The Mating Ritual.*

Proof. By induction on the number of days.

Base Case: In the beginning (that is, at the end of day 0), every woman is on every list—no one has been crossed off and so P is vacuously true.

Inductive Step: Assume P is true at the end of day d and let w be a woman that has been crossed off a man m's list by the end of day $d + 1$.

Case 1: w was crossed off m's list on day $d + 1$. Then, w must have a suitor she prefers on day $d + 1$.

Case 2: w was crossed off m's list prior to day $d + 1$. Since P is true at the end of day d, this means that w has a suitor she prefers to m on day d. She therefore has the same suitor or someone she prefers better at the end of day $d + 1$.

In both cases, P is true at the end of day $d + 1$ and so P must be an invariant. ■

With Lemma 5.2.13 in hand, we can now prove:

Theorem 5.2.14. *Everyone is married by the Mating Ritual.*

Proof. By contradiction. Assume that it is the last day of the Mating Ritual and someone does not get married. Since there are an equal number of men and women, and since bigamy is not allowed, this means that at least one man (call him Bob) and at least one woman do not get married.

Since Bob is not married, he can't be serenading anybody and so his list must be empty. This means that Bob has crossed every woman off his list and so, by invariant P, every woman has a suitor whom she prefers to Bob. Since it is the last day and every woman still has a suitor, this means that every woman gets married. This is a contradiction since we already argued that at least one woman is *not* married. Hence our assumption must be false and so everyone must be married. ■

Theorem 5.2.15. *The Mating Ritual produces a stable matching.*

Proof. Let Brad and Jen be any man and woman, respectively, that are *not* married to each other on the last day of the Mating Ritual. We will prove that Brad and Jen are not a rogue couple, and thus that all marriages on the last day are stable. There are two cases to consider.

Case 1: Jen is not on Brad's list by the end. Then by invariant P, we know that Jen has a suitor (and hence a husband) that she prefers to Brad. So she's not going to run off with Brad—Brad and Jen cannot be a rogue couple.

Case 2: Jen is on Brad's list. But since Brad is not married to Jen, he must be choosing to serenade his wife instead of Jen, so he must prefer his wife. So he's not going to run off with Jen—once again, Brad and Jenn are not a rogue couple. ■

... Especially the Men

Who is favored by the Mating Ritual, the men or the women? The women *seem* to have all the power: they stand on their balconies choosing the finest among their suitors and spurning the rest. What's more, we know their suitors can only change for the better as the Ritual progresses. Similarly, a man keeps serenading the woman he most prefers among those on his list until he must cross her off, at which point he serenades the next most preferred woman on his list. So from the man's perspective, the woman he is serenading can only change for the worse. Sounds like a good deal for the women.

But it's not! The fact is that from the beginning, the men are serenading their first choice woman, and the desirability of the woman being serenaded decreases only enough to ensure overall stability. The Mating Ritual actually does as well as possible for all the men and does the worst possible job for the women.

To explain all this we need some definitions. Let's begin by observing that while The Mating Ritual produces one stable matching, there may be other stable matchings among the same set of men and women. For example, reversing the roles of men and women will often yield a different stable matching among them.

But some spouses might be out of the question in all possible stable matchings. For example, given the preferences shown in Figure 5.13, Brad is just not in the realm of possibility for Jennifer, since if you ever pair them, Brad and Angelina will form a rogue couple.

Definition 5.2.16. Given a set of preference lists for all men and women, one person is in another person's *realm of possible spouses* if there is a stable matching in which the two people are married. A person's *optimal spouse* is their most preferred person within their realm of possibility. A person's *pessimal spouse* is their least preferred person in their realm of possibility.

Everybody has an optimal and a pessimal spouse, since we know there is at least one stable matching, namely, the one produced by the Mating Ritual. Now here is the shocking truth about the Mating Ritual:

Theorem 5.2.17. *The Mating Ritual marries every man to his optimal spouse.*

Proof. By contradiction. Assume for the purpose of contradiction that some man does not get his optimal spouse. Then there must have been a day when he crossed off his optimal spouse—otherwise he would still be serenading (and would ultimately marry) her or some even more desirable woman.

By the Well Ordering Principle, there must be a *first* day when a man (call him "Keith") crosses off his optimal spouse (call her Nicole). According to the rules of the Ritual, Keith crosses off Nicole because Nicole has a preferred suitor (call him Tom), so

$$\text{Nicole prefers Tom to Keith.} \qquad (*)$$

Since this is the first day an optimal woman gets crossed off, we know that Tom had not previously crossed off his optimal spouse, and so

$$\text{Tom ranks Nicole at least as high as his optimal spouse.} \qquad (**)$$

By the definition of an optimal spouse, there must be some stable set of marriages in which Keith gets his optimal spouse, Nicole. But then the preferences given in $(*)$ and $(**)$ imply that Nicole and Tom are a rogue couple within this supposedly stable set of marriages (think about it). This is a contradiction. ■

Theorem 5.2.18. *The Mating Ritual marries every woman to her pessimal spouse.*

Proof. By contradiction. Assume that the theorem is not true. Hence there must be a stable set of marriages $\mathcal{M}$ where some woman (call her Nicole) is married to a man (call him Tom) that she likes less than her spouse in The Mating Ritual (call him Keith). This means that

$$\text{Nicole prefers Keith to Tom.} \qquad (+)$$

By Theorem 5.2.17 and the fact that Nicole and Keith are married in the Mating Ritual, we know that

$$\text{Keith prefers Nicole to his spouse in } \mathcal{M}. \qquad (++)$$

This means that Keith and Nicole form a rogue couple in $\mathcal{M}$, which contradicts the stability of $\mathcal{M}$. ■

Applications

The Mating Ritual was first announced in a paper by D. Gale and L.S. Shapley in 1962, but ten years before the Gale-Shapley paper was published, and unknown by them, a similar algorithm was being used to assign residents to hospitals by the National Resident Matching Program (NRMP)[10]. The NRMP has, since the turn of the twentieth century, assigned each year's pool of medical school graduates to hospital residencies (formerly called "internships") with hospitals and graduates playing the roles of men and women. (In this case, there may be multiple women married to one man, a scenario we consider in the problem section at the end of the chapter.). Before the Ritual-like algorithm was adopted, there were chronic disruptions and awkward countermeasures taken to preserve assignments of graduates to residencies. The Ritual resolved these problems so successfully, that it was used essentially without change at least through 1989.[11]

The Internet infrastructure company, Akamai, also uses a variation of the Mating Ritual to assign web traffic to its servers. In the early days, Akamai used other combinatorial optimization algorithms that got to be too slow as the number of servers (over 65,000 in 2010) and requests (over 800 billion per day) increased. Akamai switched to a Ritual-like approach since it is fast and can be run in a distributed manner. In this case, web requests correspond to women and web servers correspond to men. The web requests have preferences based on latency and packet loss, and the web servers have preferences based on cost of bandwidth and collocation.

Not surprisingly, the Mating Ritual is also used by at least one large online dating agency. Even here, there is no serenading going on—everything is handled by computer.

[10] Of course, there is no serenading going on in the hospitals—the preferences are submitted to a program and the whole process is carried out by a computer.

[11] Much more about the Stable Marriage Problem can be found in the very readable mathematical monograph by Dan Gusfield and Robert W. Irving, The Stable Marriage Problem: Structure and Algorithms, MIT Press, Cambridge, Massachusetts, 1989, 240 pp.

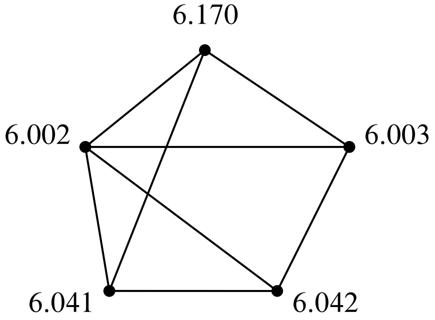


Figure 5.15 A scheduling graph for five exams. Exams connected by an edge cannot be given at the same time.

5.3 Coloring

In Section 5.2, we used edges to indicate an affinity between a pair of nodes. We now consider situations where it is useful to use edges to represent a *conflict* between a pair of nodes. For example, consider the following exam scheduling problem.

5.3.1 An Exam Scheduling Problem

Each term, the MIT Schedules Office must assign a time slot for each final exam. This is not easy, because some students are taking several classes with finals, and (even at MIT) a student can take only one test during a particular time slot. The Schedules Office wants to avoid all conflicts. Of course, you can make such a schedule by having every exam in a different slot, but then you would need hundreds of slots for the hundreds of courses, and the exam period would run all year! So, the Schedules Office would also like to keep exam period short.

The Schedules Office's problem is easy to describe as a graph. There will be a vertex for each course with a final exam, and two vertices will be adjacent exactly when some student is taking both courses. For example, suppose we need to schedule exams for 6.041, 6.042, 6.002, 6.003 and 6.170. The scheduling graph might appear as in Figure 5.15.

6.002 and 6.042 cannot have an exam at the same time since there are students in both courses, so there is an edge between their nodes. On the other hand, 6.042 and 6.170 can have an exam at the same time if they're taught at the same time (which they sometimes are), since no student can be enrolled in both (that is, no student *should* be enrolled in both when they have a timing conflict).

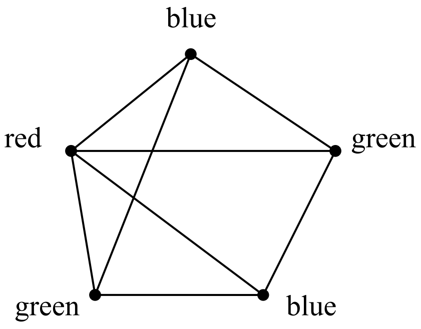


Figure 5.16 A 3-coloring of the exam graph from Figure 5.15.

We next identify each time slot with a color. For example, Monday morning is red, Monday afternoon is blue, Tuesday morning is green, etc. Assigning an exam to a time slot is then equivalent to coloring the corresponding vertex. The main constraint is that *adjacent vertices must get different colors*—otherwise, some student has two exams at the same time. Furthermore, in order to keep the exam period short, we should try to color all the vertices using as *few different colors as possible*. As shown in Figure 5.16, three colors suffice for our example.

The coloring in Figure 5.16 corresponds to giving one final on Monday morning (red), two Monday afternoon (blue), and two Tuesday morning (green). Can we use fewer than three colors? No! We can't use only two colors since there is a triangle in the graph, and three vertices in a triangle must all have different colors.

This is an example of a *graph coloring problem*: given a graph G, assign colors to each node such that adjacent nodes have different colors. A color assignment with this property is called a *valid coloring* of the graph—a "*coloring*," for short. A graph G is k*-colorable* if it has a coloring that uses at most k colors.

Definition 5.3.1. The minimum value of k for which a graph G has a valid k-coloring is called its *chromatic number*, $\chi(G)$.

In general, trying to figure out if you can color a graph with a fixed number of colors can take a long time. It's a classic example of a problem for which no fast algorithms are known. It is easy to check if a coloring works, but it seems really hard to find it. (If you figure out how, then you can get a $1 million Clay prize.)

5.3.2 Degree-Bounded Coloring

There are some simple graph properties that give useful upper bounds on the chromatic number. For example, if the graph is bipartite, then we can color it with 2 colors (one color for the nodes in the "left" set and a second color for the nodes in the "right" set). In fact, if the graph has any edges at all, then being bipartite is equivalent to being 2-colorable.

Alternatively, if the graph is planar, then the famous 4-Color Theorem says that the graph is 4-colorable. This is a hard result to prove, but we will come close in Section 5.8 where we define planar graphs and prove that they are 5-colorable.

The chromatic number of a graph can also be shown to be small if the vertex degrees of the graph are small. In particular, if we have an upper bound on the degrees of all the vertices in a graph, then we can easily find a coloring with only one more color than the degree bound.

Theorem 5.3.2. *A graph with maximum degree at most k is $(k + 1)$-colorable.*

The natural way to try to prove this theorem is to use induction on k. Unfortunately, this approach leads to disaster. It is not that it is impossible, just that it is extremely painful and would ruin your week if you tried it on an exam. When you encounter such a disaster when using induction on graphs, it is usually best to change what you are inducting on. In graphs, typical good choices for the induction parameter are n, the number of nodes, or e, the number of edges.

Proof of Theorem 5.3.2. We use induction on the number of vertices in the graph, which we denote by n. Let $P(n)$ be the proposition that an n-vertex graph with maximum degree at most k is $(k + 1)$-colorable.

Base case $(n = 1)$: A 1-vertex graph has maximum degree 0 and is 1-colorable, so $P(1)$ is true.

Inductive step: Now assume that $P(n)$ is true, and let G be an $(n+1)$-vertex graph with maximum degree at most k. Remove a vertex v (and all edges incident to it), leaving an n-vertex subgraph, H. The maximum degree of H is at most k, and so H is $(k + 1)$-colorable by our assumption $P(n)$. Now add back vertex v. We can assign v a color (from the set of $k + 1$ colors) that is different from all its adjacent vertices, since there are at most k vertices adjacent to v and so at least one of the $k + 1$ colors is still available. Therefore, G is $(k + 1)$-colorable. This completes the inductive step, and the theorem follows by induction. ■

Sometimes $k + 1$ colors is the best you can do. For example, in the complete graph, K_n, every one of its n vertices is adjacent to all the others, so all n must be assigned different colors. Of course n colors is also enough, so $\chi(K_n) = n$. In this case, every node has degree $k = n - 1$ and so this is an example where Theorem 5.3.2 gives the best possible bound. By a similar argument, we can show that Theorem 5.3.2 gives the best possible bound for *any* graph with degree bounded by k that has K_{k+1} as a subgraph.

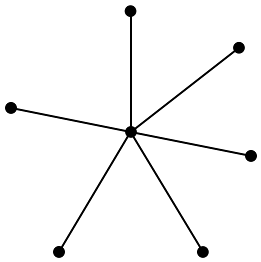


Figure 5.17 A 7-node star graph.

But sometimes $k + 1$ colors is far from the best that you can do. For example, the n-node *star graph* shown in Figure 5.17 has maximum degree $n - 1$ but can be colored using just 2 colors.

5.3.3 Why coloring?

One reason coloring problems frequently arise in practice is because scheduling conflicts are so common. For example, at Akamai, a new version of software is deployed over each of 75,000 servers every few days. The updates cannot be done at the same time since the servers need to be taken down in order to deploy the software. Also, the servers cannot be handled one at a time, since it would take forever to update them all (each one takes about an hour). Moreover, certain pairs of servers cannot be taken down at the same time since they have common critical functions. This problem was eventually solved by making a 75,000-node conflict graph and coloring it with 8 colors—so only 8 waves of install are needed!

Another example comes from the need to assign frequencies to radio stations. If two stations have an overlap in their broadcast area, they can't be given the same frequency. Frequencies are precious and expensive, so you want to minimize the number handed out. This amounts to finding the minimum coloring for a graph whose vertices are the stations and whose edges connect stations with overlapping areas.

Coloring also comes up in allocating registers for program variables. While a variable is in use, its value needs to be saved in a register. Registers can be reused for different variables but two variables need different registers if they are referenced during overlapping intervals of program execution. So register allocation is the coloring problem for a graph whose vertices are the variables; vertices are adjacent if their intervals overlap, and the colors are registers. Once again, the goal is to minimize the number of colors needed to color the graph.

Finally, there's the famous map coloring problem stated in Proposition 1.3.4. The question is how many colors are needed to color a map so that adjacent territories get different colors? This is the same as the number of colors needed to color a graph that can be drawn in the plane without edges crossing. A proof that four colors are enough for *planar* graphs was acclaimed when it was discovered about thirty years ago. Implicit in that proof was a 4-coloring procedure that takes time proportional to the number of vertices in the graph (countries in the map). Surprisingly, it's another of those million dollar prize questions to find an efficient procedure to tell if a planar graph really *needs* four colors or if three will actually do the job. (It's always easy to tell if an *arbitrary* graph is 2-colorable.) In Section 5.8, we'll develop enough planar graph theory to present an easy proof that all planar graphs are 5-colorable.

5.4 Getting from A to B in a Graph

5.4.1 Paths and Walks

Definition 5.4.1. A *walk*[12] in a graph, G, is a sequence of vertices

$$v_0, v_1, \dots, v_k$$

and edges

$$\{v_0, v_1\}, \{v_1, v_2\}, \dots, \{v_{k-1}, v_k\}$$

such that $\{v_i, v_{i+1}\}$ is an edge of G for all i where $0 \le i < k$. The walk is said to *start* at v_0 and to *end* at v_k, and the *length* of the walk is defined to be k. An edge, $\{u, v\}$, is *traversed* n times by the walk if there are n different values of i such that $\{v_i, v_{i+1}\} = \{u, v\}$. A *path* is a walk where all the v_i's are different, that is, $i \neq j$ implies $v_i \neq v_j$. For simplicity, we will refer to paths and walks by the sequence of vertices.[13]

For example, the graph in Figure 5.18 has a length 6 path a, b, c, d, e, f, g. This is the longest path in the graph. Of course, the graph has walks with arbitrarily large lengths; for example, $a, b, a, b, a, b, \dots$.

The length of a walk or path is the total number of times it traverses edges, which is *one less* than its length as a sequence of vertices. For example, the length 6 path a, b, c, d, e, f, g contains a sequence of 7 vertices.

[12] Some texts use the word *path* for our definition of walk and the term *simple path* for our definition of path.

[13] This works fine for simple graphs since the edges in a walk are completely determined by the sequence of vertices and there is no ambiguity. For graphs with multiple edges, we would need to specify the edges as well as the nodes.

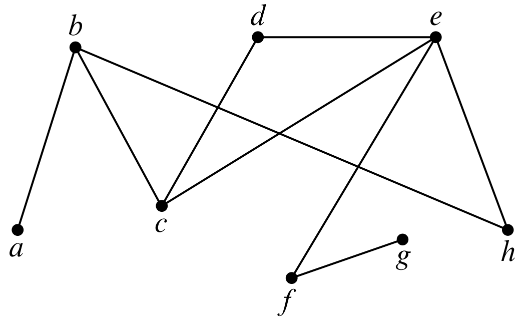


Figure 5.18 A graph containing a path a, b, c, d, e, f, g of length 6.

5.4.2 Finding a Path

Where there's a walk, there's a path. This is sort of obvious, but it's easy enough to prove rigorously using the Well Ordering Principle.

Lemma 5.4.2. *If there is a walk from a vertex u to a vertex v in a graph, then there is a path from u to v.*

Proof. Since there is a walk from u to v, there must, by the Well-ordering Principle, be a minimum length walk from u to v. If the minimum length is zero or one, this minimum length walk is itself a path from u to v. Otherwise, there is a minimum length walk

$$v_0, v_1, \dots, v_k$$

from $u = v_0$ to $v = v_k$ where $k \geq 2$. We claim this walk must be a path. To prove the claim, suppose to the contrary that the walk is not a path; that is, some vertex on the walk occurs twice. This means that there are integers i, j such that $0 \leq i < j \leq k$ with $v_i = v_j$. Then deleting the subsequence

$$v_{i+1}, \dots, v_j$$

yields a strictly shorter walk

$$v_0, v_1, \dots, v_i, v_{j+1}, v_{j+2}, \dots, v_k$$

from u to v, contradicting the minimality of the given walk. ■

Actually, we proved something stronger:

Corollary 5.4.3. *For any walk of length k in a graph, there is a path of length* at most *k with the same endpoints. Moreover, the shortest walk between a pair of vertices is, in fact, a path.*

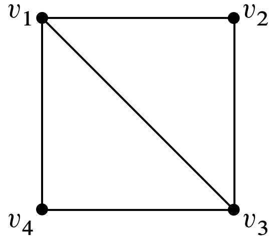


Figure 5.19 A graph for which there are 5 walks of length 3 from v_1 to v_4. The walks are (v_1, v_2, v_1, v_4), (v_1, v_3, v_1, v_4), (v_1, v_4, v_1, v_4), (v_1, v_2, v_3, v_4), and (v_1, v_4, v_3, v_4).

5.4.3 Numbers of Walks

Given a pair of nodes that are connected by a walk of length k in a graph, there are often many walks that can be used to get from one node to the other. For example, there are 5 walks of length 3 that start at v_1 and end at v_4 in the graph shown in Figure 5.19.

There is a surprising relationship between the number of walks of length k between a pair of nodes in a graph G and the kth power of the adjacency matrix A_G for G. The relationship is captured in the following theorem.

Theorem 5.4.4. *Let $G = (V, E)$ be an n-node graph with $V = \{v_1, v_2, \ldots, v_n\}$ and let $A_G = \{a_{ij}\}$ denote the adjacency matrix for G. Let $a_{ij}^{(k)}$ denote the (i, j)-entry of the kth power of A_G. Then the number of walks of length k between v_i and v_j is $a_{ij}^{(k)}$.*

In other words, we can determine the number of walks of length k between any pair of nodes simply by computing the kth power of the adjacency matrix! That's pretty amazing.

For example, the first three powers of the adjacency matrix for the graph in Figure 5.19 are:

$$A = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \qquad A^2 = \begin{pmatrix} 3 & 1 & 2 & 1 \\ 1 & 2 & 1 & 2 \\ 2 & 1 & 3 & 1 \\ 1 & 2 & 1 & 2 \end{pmatrix} \qquad A^3 = \begin{pmatrix} 4 & 5 & 5 & 5 \\ 5 & 2 & 5 & 2 \\ 5 & 5 & 4 & 5 \\ 5 & 2 & 5 & 2 \end{pmatrix}$$

Sure enough, the $(1, 4)$ coordinate of A^3 is $a_{14}^{(3)} = 5$, which is the number of length 3 walks from v_1 to v_4. And $a_{24}^{(3)} = 2$, which is the number of length 3 walks from v_2 to v_4. By proving the theorem, we'll discover why it is true and thereby uncover the relationship between matrix multiplication and numbers of walks.

Proof of Theorem 5.4.4. The proof is by induction on k. We will let $P(k)$ be the predicate that the theorem is true for k. Let $P_{ij}^{(k)}$ denote the number of walks of length k between v_i and v_j. Then $P(k)$ is the predicate

$$\forall i, j \in [1, n].\ P_{ij}^{(k)} = a_{ij}^{(k)}. \tag{5.2}$$

Base Case ($k = 1$): There are two cases to consider:

Case 1: $\{v_i, v_j\} \in E$. Then $P_{ij}^{(1)} = 1$ since there is precisely one walk of length 1 between v_i and v_j. Moreover, $\{v_i, v_j\} \in E$ means that $a_{ij}^{(1)} = a_{ij} = 1$. So, $P_{ij}^{(1)} = a_{ij}^{(1)}$ in this case.

Case 2: $\{v_i, v_j\} \notin E$. Then $P_{ij}^{(1)} = 0$ since there cannot be any walks of length 1 between v_i and v_j. Moreover, $\{v_i, v_j\} \notin E$ means that $a_{ij} = 0$. So, $P_{ij}^{(1)} = a_{ij}^{(1)}$ in this case as well.

Hence, $P(1)$ must be true.

Inductive Step: Assume $P(k)$ is true. In other words, assume that equation 5.2 holds.

We can group (and thus count the number of) walks of length $k+1$ from v_i to v_j according to the first edge in the walk (call it $\{v_i, v_t\}$). This means that

$$P_{ij}^{(k+1)} = \sum^{t:\{v_i,v_t\}\in E} P_{tj}^{(k)} \tag{5.3}$$

where the sum is over all t such that $\{v_i, v_t\}$ is an edge. Using the fact that $a_{ij} = 1$ if $\{v_i, v_t\} \in E$ and $a_{it} = 0$ otherwise, we can rewrite Equation 5.3 as follows:

$$P_{ij}^{(k+1)} = \sum_{t=1}^{n} a_{it} P_{tj}^{(k)}.$$

By the inductive hypothesis, $P_{tj}^{(k)} = a_{tj}^{(k)}$ and thus

$$P_{ij}^{(k+1)} = \sum_{t=1}^{n} a_{it} a_{tj}^{(k)}.$$

But the formula for matrix multiplication gives that

$$a_{ij}^{(k+1)} = \sum_{t=1}^{n} a_{it} a_{tj}^{(k)}.$$

and so we must have $P_{ij}^{(k+1)} = a_{ij}^{(k+1)}$ for all $i, j \in [1, n]$. Hence $P(k+1)$ is true and the induction is complete. ■

5.4.4 Shortest Paths

Although the connection between the power of the adjacency matrix and the number of walks is cool (at least if you are a mathematician), the problem of counting walks does not come up very often in practice. Much more important is the problem of finding the shortest path between a pair of nodes in a graph.

There is good news and bad news to report on this front. The good news is that it is not very hard to find a shortest path. The bad news is that you can't win one of those million dollar prizes for doing it.

In fact, there are several good algorithms known for finding a Shortest Path between a pair of nodes. The simplest to explain (but not the fastest) is to compute the powers of the adjacency matrix one by one until the value of $a_{ij}^{(k)}$ exceeds 0. That's because Theorem 5.4.4 and Corollary 5.4.3 imply that the length of the shortest path between v_i and v_j will be the smallest value of k for which $a_{ij}^{(k)} > 0$.

Paths in Weighted Graphs

The problem of computing shortest paths in a weighted graph frequently arises in practice. For example, when you drive home for vacation, you usually would like to take the shortest route.

Definition 5.4.5. Given a weighted graph, the length of a path in the graph is the sum of the weights of the edges in the path.

Finding shortest paths in weighted graphs is not a lot harder than finding shortest paths in unweighted graphs. We won't show you how to do it here, but you will study algorithms for finding shortest paths if you take an algorithms course. Not surprisingly, the proof of correctness will use induction.

5.5 Connectivity

Definition 5.5.1. Two vertices in a graph are said to be *connected* if there is a path that begins at one and ends at the other. By convention, every vertex is considered to be connected to itself by a path of length zero.

Definition 5.5.2. A graph is said to be *connected* when every pair of vertices are connected.

5.5.1 Connected Components

Being connected is usually a good property for a graph to have. For example, it could mean that it is possible to get from any node to any other node, or that it is possible to communicate between any pair of nodes, depending on the application.

But not all graphs are connected. For example, the graph where nodes represent cities and edges represent highways might be connected for North American cities, but would surely not be connected if you also included cities in Australia. The same is true for communication networks like the Internet—in order to be protected from viruses that spread on the Internet, some government networks are completely isolated from the Internet.

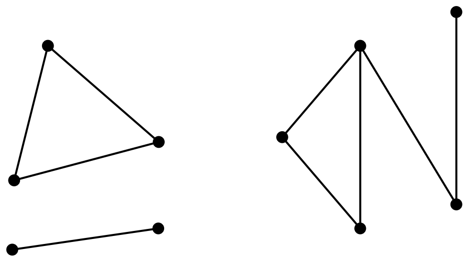


Figure 5.20 One graph with 3 connected components.

For example, the diagram in Figure 5.20 looks like a picture of three graphs, but is intended to be a picture of *one* graph. This graph consists of three pieces (subgraphs). Each piece by itself is connected, but there are no paths between vertices in different pieces. These connected pieces of a graph are called its *connected components*.

Definition 5.5.3. A *connected component* is a subgraph of a graph consisting of some vertex and every node and edge that is connected to that vertex.

So a graph is connected iff it has exactly one connected component. At the other extreme, the empty graph on n vertices has n connected components.

5.5.2 k-Connected Graphs

If we think of a graph as modeling cables in a telephone network, or oil pipelines, or electrical power lines, then we not only want connectivity, but we want connectivity that survives component failure. A graph is called k*-edge connected* if it takes at least k "edge-failures" to disconnect it. More precisely:

Definition 5.5.4. Two vertices in a graph are k*-edge connected* if they remain connected in every subgraph obtained by deleting $k - 1$ edges. A graph with at least two vertices is k-edge connected[14] if every two of its vertices are k-edge connected.

[14] The corresponding definition of connectedness based on deleting vertices rather than edges is common in Graph Theory texts and is usually simply called "k-connected" rather than "k-vertex connected."

So 1-edge connected is the same as connected for both vertices and graphs. Another way to say that a graph is k-edge connected is that every subgraph obtained from it by deleting at most $k - 1$ edges is connected. For example, in the graph in Figure 5.18, vertices c and e are 3-edge connected, b and e are 2-edge connected, g and e are 1-edge connected, and no vertices are 4-edge connected. The graph as a whole is only 1-edge connected. The complete graph, K_n, is $(n - 1)$-edge connected.

If two vertices are connected by k edge-disjoint paths (that is, no two paths traverse the same edge), then they are obviously k-edge connected. A fundamental fact, whose ingenious proof we omit, is Menger's theorem which confirms that the converse is also true: if two vertices are k-edge connected, then there are k edge-disjoint paths connecting them. It even takes some ingenuity to prove this for the case $k = 2$.

5.5.3 The Minimum Number of Edges in a Connected Graph

The following theorem says that a graph with few edges must have many connected components.

Theorem 5.5.5. *Every graph with v vertices and e edges has at least $v - e$ connected components.*

Of course for Theorem 5.5.5 to be of any use, there must be fewer edges than vertices.

Proof. We use induction on the number of edges, e. Let $P(e)$ be the proposition that

> for every v, every graph with v vertices and e edges has at least $v - e$ connected components.

Base case:($e = 0$). In a graph with 0 edges and v vertices, each vertex is itself a connected component, and so there are exactly $v = v - 0$ connected components. So $P(e)$ holds.

Inductive step: Now we assume that the induction hypothesis holds for every e-edge graph in order to prove that it holds for every $(e+1)$-edge graph, where $e \geq 0$. Consider a graph, G, with $e + 1$ edges and v vertices. We want to prove that G has at least $v - (e + 1)$ connected components. To do this, remove an arbitrary edge $\{a, b\}$ and call the resulting graph G'. By the induction assumption, G' has at least $v - e$ connected components. Now add back the edge $\{a, b\}$ to obtain the original graph G. If a and b were in the same connected component of G', then G has the same connected components as G', so G has at least $v-e > v-(e+1)$ components.

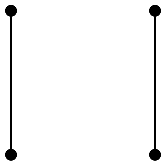


Figure 5.21 A counterexample graph to the False Claim.

Otherwise, if a and b were in different connected components of G', then these two components are merged into one component in G, but all other components remain unchanged, reducing the number of components by 1. Therefore, G has at least $(v-e)-1 = v-(e+1)$ connected components. So in either case, $P(e+1)$ holds. This completes the Inductive step. The theorem now follows by induction. ■

Corollary 5.5.6. *Every connected graph with v vertices has at least $v - 1$ edges.*

A couple of points about the proof of Theorem 5.5.5 are worth noticing. First, we used induction on the number of edges in the graph. This is very common in proofs involving graphs, as is induction on the number of vertices. When you're presented with a graph problem, these two approaches should be among the first you consider.

The second point is more subtle. Notice that in the inductive step, we took an arbitrary $(n+1)$-edge graph, threw out an edge so that we could apply the induction assumption, and then put the edge back. You'll see this shrink-down, grow-back process very often in the inductive steps of proofs related to graphs. This might seem like needless effort; why not start with an n-edge graph and add one more to get an $(n + 1)$-edge graph? That would work fine in this case, but opens the door to a nasty logical error called *buildup error*.

5.5.4 Build-Up Error

False Claim. *If every vertex in a graph has degree at least 1, then the graph is connected.*

There are many counterexamples; for example, see Figure 5.21.

False proof. We use induction. Let $P(n)$ be the proposition that if every vertex in an n-vertex graph has degree at least 1, then the graph is connected.

Base case: There is only one graph with a single vertex and has degree 0. Therefore, $P(1)$ is vacuously true, since the if-part is false.

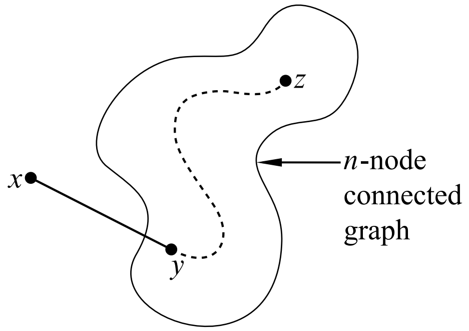


Figure 5.22 Adding a vertex x with degree at least 1 to a connected n-node graph.

Inductive step: We must show that $P(n)$ implies $P(n+1)$ for all $n \geq 1$. Consider an n-vertex graph in which every vertex has degree at least 1. By the assumption $P(n)$, this graph is connected; that is, there is a path between every pair of vertices. Now we add one more vertex x to obtain an $(n+1)$-vertex graph as shown in Figure 5.22.

All that remains is to check that there is a path from x to every other vertex z. Since x has degree at least one, there is an edge from x to some other vertex; call it y. Thus, we can obtain a path from x to z by adjoining the edge $\{x, y\}$ to the path from y to z. This proves $P(n+1)$.

By the principle of induction, $P(n)$ is true for all $n \geq 1$, which proves the theorem ■

Uh-oh... this proof looks fine! Where is the bug? It turns out that the faulty assumption underlying this argument is that *every $(n+1)$-vertex graph with minimum degree 1 can be obtained from an n-vertex graph with minimum degree 1 by adding 1 more vertex*. Instead of starting by considering an arbitrary $(n+1)$-node graph, this proof only considered $(n+1)$-node graphs that you can make by starting with an n-node graph with minimum degree 1.

The counterexample in Figure 5.21 shows that this assumption is false; there is no way to build the 4-vertex graph in Figure 5.21 from a 3-vertex graph with minimum degree 1. Thus the first error in the proof is the statement "This proves $P(n+1)$."

This kind of flaw is known as "build-up error." Usually, build-up error arises from a faulty assumption that every size $n+1$ graph with some property can be "built up" from a size n graph with the same property. (This assumption is correct for some properties, but incorrect for others—such as the one in the argument above.)

One way to avoid an accidental build-up error is to use a "shrink down, grow back" process in the inductive step; that is, start with a size $n + 1$ graph, remove a vertex (or edge), apply the inductive hypothesis $P(n)$ to the smaller graph, and then add back the vertex (or edge) and argue that $P(n + 1)$ holds. Let's see what would have happened if we'd tried to prove the claim above by this method:

Revised inductive step: We must show that $P(n)$ implies $P(n + 1)$ for all $n \geq 1$. Consider an $(n + 1)$-vertex graph G in which every vertex has degree at least 1. Remove an arbitrary vertex v, leaving an n-vertex graph G' in which every vertex has degree… uh oh!

The reduced graph G' might contain a vertex of degree 0, making the inductive hypothesis $P(n)$ inapplicable! We are stuck—and properly so, since the claim is false!

Always use shrink-down, grow-back arguments and you'll never fall into this trap.

5.6 Around and Around We Go

5.6.1 Cycles and Closed Walks

Definition 5.6.1. A *closed walk*[15] in a graph G is a sequence of vertices

$$v_0, v_1, \ldots, v_k$$

and edges

$$\{v_0, v_1\}, \{v_1, v_2\}, \ldots, \{v_{k-1}, v_k\}$$

where v_0 is the same node as v_k and $\{v_i, v_{i+1}\}$ is an edge of G for all i where $0 \leq i < k$. The *length* of the closed walk is k. A closed walk is said to be a *cycle* if $k \geq 3$ and $v_0, v_1, \ldots, v_{k-1}$ are all different.

For example, b, c, d, e, c, b is a closed walk of length 5 in the graph shown in Figure 5.18. It is not a cycle since it contains node c twice. On the other hand, c, d, e, c is a cycle of length 3 in this graph since every node appears just once.

There are many ways to represent the same closed walk or cycle. For example, b, c, d, e, c, b is the same as c, d, e, c, b, c (just starting at node c instead of node b) and the same as b, c, e, d, c, b (just reversing the direction).

[15] Some texts use the word *cycle* for our definition of closed walk and *simple cycle* for our definition of cycle.

Cycles are similar to paths, except that the last node is the first node and the notion of first and last does not matter. Indeed, there are many possible vertex orders that can be used to describe cycles and closed walks, whereas walks and paths have a prescribed beginning, end, and ordering.

5.6.2 Odd Cycles and 2-Colorability

We have already seen that determining the chromatic number of a graph is a challenging problem. There is a special case where this problem is very easy; namely, the case where every cycle in the graph has even length. In this case, the graph is 2-colorable! Of course, this is optimal if the graph has any edges at all. More generally, we will prove

Theorem 5.6.2. *The following properties of a graph are equivalent (that is, if the graph has any one of the properties, then it has all of the properties):*

1. *The graph is bipartite.*
2. *The graph is 2-colorable.*
3. *The graph does not contain any cycles with odd length.*
4. *The graph does not contain any closed walks with odd length.*

Proof. We will show that property 1 IMPLIES property 2, property 2 IMPLIES property 3, property 3 IMPLIES property 4, and property 4 IMPLIES property 1. This will show that all four properties are equivalent by repeated application of Rule 2.1.2 in Section 2.1.2.

1 IMPLIES 2 Assume that $G = (V, E)$ is a bipartite graph. Then V can be partitioned into two sets L and R so that no edge connects a pair of nodes in L nor a pair of nodes in R. Hence, we can use one color for all the nodes in L and a second color for all the nodes in R. Hence $\chi(G) = 2$.

2 IMPLIES 3 Let $G = (V, E)$ be a 2-colorable graph and

$$C ::= v_0, v_1, \ldots, v_k$$

be any cycle in G. Consider any 2-coloring for the nodes of G. Since $\{v_i, v_{i+1}\} \in E$, v_i and v_{i+1} must be differently colored for $0 \leq i < k$. Hence v_0, v_2, v_4, ..., have one color and v_1, v_3, v_5, ..., have the other color. Since C is a cycle, v_k is the same node as v_0, which means they must have the same color, and so k must be an even number. This means that C has even length.

3 IMPLIES 4 The proof is by contradiction. Assume for the purposes of contradiction that G is a graph that does not contain any cycles with odd length (that is, G satisfies Property 3) but that G *does* contain a closed walk with odd length (that is, G does not satisfy Property 4).

Let

$$w ::= v_0, v_1, v_2, \ldots, v_k$$

be the *shortest* closed walk with odd length in G. Since G has no odd-length cycles, w cannot be a cycle. Hence $v_i = v_j$ for some $0 \le i < j < k$. This means that w is the union of two closed walks:

$$v_0, v_1, \ldots, v_i, v_{j+1}, v_{j+2}, \ldots, v_k$$

and

$$v_i, v_{i+1}, \ldots, v_j.$$

Since w has odd length, one of these two closed walks must also have odd length and be shorter than w. This contradicts the minimality of w. Hence 3 IMPLIES 4.

4 IMPLIES 1 Once again, the proof is by contradiction. Assume for the purposes of contradictin that G is a graph without any closed walks with odd length (that is, G satisfies Property 4) but that G is *not* bipartite (that is, G does not satisfy Property 1).

Since G is not bipartite, it must contain a connected component $G' = (V', E')$ that is not bipartite. Let v be some node in V'. For every node $u \in V'$, define

$$\begin{aligned}\text{dist}(u) ::= &\text{ the length of the shortest path from } u \text{ to } v \text{ in } G'.\\ &\text{If } u = v, \text{ the distance is zero.}\end{aligned}$$

Partition V' into sets L and R so that

$$\begin{aligned}L &= \{\, u \mid \text{dist}(u) \text{ is even} \,\},\\ R &= \{\, u \mid \text{dist}(u) \text{ is odd} \,\}.\end{aligned}$$

Since G' is not bipartite, there must be a pair of adjacent nodes u_1 and u_2 that are both in L or both in R. Let e denote the edge incident to u_1 and u_2.

Let P_i denote a shortest path in G' from u_i to v for $i = 1, 2$. Because u_1 and u_2 are both in L or both in R, it must be the case that P_1 and P_2 both have even length or they both have odd length. In either case, the union of P_1, P_2, and e forms a closed walk with odd length, which is a contradiction. Hence 4 IMPLIES 1. ■

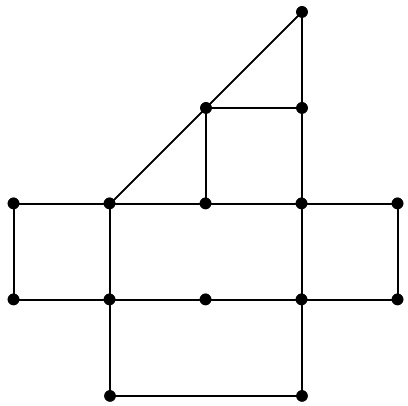


Figure 5.23 A possible floor plan for a museum. Can you find a walk that traverses every edge exactly once?

Theorem 5.6.2 turns out to be useful since bipartite graphs come up fairly often in practice. We'll see examples when we talk about planar graphs in Section 5.8 and when we talk about packet routing in communication networks in Chapter 6.

5.6.3 Euler Tours

Can you walk every hallway in the Museum of Fine Arts *exactly once*? If we represent hallways and intersections with edges and vertices, then this reduces to a question about graphs. For example, could you visit every hallway exactly once in a museum with the floor plan in Figure 5.23?

The entire field of graph theory began when Euler asked whether the seven bridges of Königsberg could all be traversed exactly once—essentially the same question we asked about the Museum of Fine Arts. In his honor, an *Euler walk* is a defined to be a walk that traverses every edge in a graph exactly once. Similarly, an *Euler tour* is an Euler walk that starts and finishes at the same vertex. Graphs with Euler tours and Euler walks both have simple characterizations.

Theorem 5.6.3. *A connected graph has an Euler tour if and only if every vertex has even degree.*

Proof. We first show that if a graph has an Euler tour, then every vertex has even degree. Assume that a graph $G = (V, E)$ has an Euler tour $v_0, v_1, \ldots, v_k$ where $v_k = v_0$. Since every edge is traversed once in the tour, $k = |E|$ and the degree of a node u in G is the number of times that node appears in the sequence $v_0, v_1, \ldots,$ v_{k-1} times two. We multiply by two since if $u = v_i$ for some i where $0 < i < k$, then both $\{v_{i-1}, v_i\}$ and $\{v_i, v_{i+1}\}$ are edges incident to u in G. If $u = v_0 = v_k$, then both $\{v_{k-1}, v_k\}$ and $\{v_0, v_1\}$ are edges incident to u in G. Hence, the degree of every node is even.

We next show that if the degree of every node is even in a graph $G = (V, E)$, then there is an Euler tour. Let

$$W ::= v_0, v_1, \ldots, v_k$$

be the longest walk in G that traverses *no edge more than once*[16]. W must traverse every edge incident to v_k; otherwise the walk could be extended and W would not be the longest walk that traverses all edges at most once. Moreover, it must be that $v_k = v_0$ and that W is a closed walk, since otherwise v_k would have odd degree in W (and hence in G), which is not possible by assumption.

We conclude the argument with a proof by contradiction. Suppose that W is not an Euler tour. Because G is a connected graph, we can find an edge not in W but incident to some vertex in W. Call this edge $\{u, v_i\}$. But then we can construct a walk W' that is longer than W but that still uses no edge more than once:

$$W' ::= u, v_i, v_{i+1}, \ldots, v_k, v_1, v_2, \ldots, v_i.$$

This contradicts the definition of W, so W must be an Euler tour after all. ■

It is not difficult to extend Theorem 5.6.3 to prove that a connected graph G has an Euler walk if and only if precisely 0 or 2 nodes in G have odd degree. Hence, we can conclude that the graph shown in Figure 5.23 has an Euler walk but not an Euler tour since the graph has precisely two nodes with odd degree.

Although the proof of Theorem 5.6.3 does not explicitly define a method for finding an Euler tour when one exists, it is not hard to modify the proof to produce such a method. The idea is to grow a tour by continually splicing in closed walks until all the edges are consumed.

5.6.4 Hamiltonian Cycles

Hamiltonian cycles are the unruly cousins of Euler tours.

Definition 5.6.4. A *Hamiltonian cycle* in a graph G is a cycle that visits every *node* in G exactly once. Similarly, a *Hamiltonian* path is a path in G that visits every node exactly once.

[16] Did you notice that we are using a variation of the Well Ordering Principle here when we implicitly assume that a longest walk exists? This is ok since the length of a walk where no edge is used more than once is at most $|E|$.

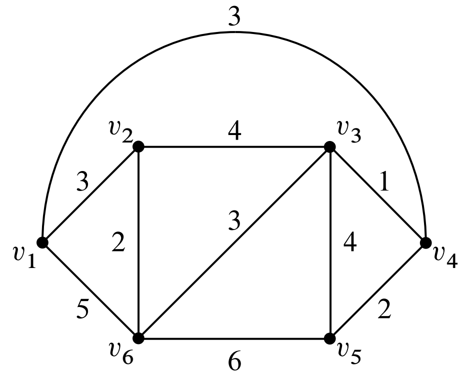


Figure 5.24 A weighted graph. Can you find a cycle with weight 15 that visits every node exactly once?

Although Hamiltonian cycles sound similar to Euler tours—one visits every node once while the other visits every edge once—finding a Hamiltonian cycle can be a lot harder than finding an Euler tour. The same is true for Hamiltonian paths. This is because no one has discovered a simple characterization of all graphs with a Hamiltonian cycle. In fact, determining whether a graph has a Hamiltonian cycle is the same category of problem as the SAT problem of Section 1.5 and the coloring problem in Section 5.3; you get a million dollars for finding an efficient way to determine when a graph has a Hamiltonian cycle—or proving that no procedure works efficiently on all graphs.

5.6.5 The Traveling Salesperson Problem

As if the problem of finding a Hamiltonian cycle is not hard enough, when the graph is weighted, we often want to find a Hamiltonian cycle that has least possible weight. This is a very famous optimization problem known as the Traveling Salesperson Problem.

Definition 5.6.5. Given a weighted graph G, the *weight* of a cycle in G is defined as the sum of the weights of the edges in the cycle.

For example, consider the graph shown in Figure 5.24 and suppose that you would like to visit every node once and finish at the node where you started. Can you find way to do this by traversing a cycle with weight 15?

Needless to say, if you can figure out a fast procedure that finds the optimal cycle for the traveling salesperson, let us know so that we can win a million dollars.

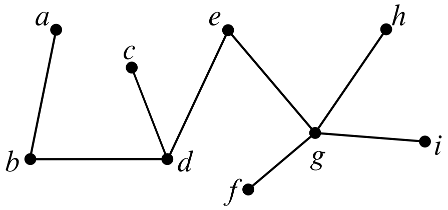


Figure 5.25 A 9-node tree.

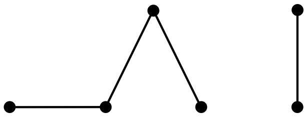


Figure 5.26 A 6-node forest consisting of 2 component trees. Note that this 6-node graph is not itself a tree since it is not connected.

5.7 Trees

As we have just seen, finding good cycles in a graph can be trickier than you might first think. But what if a graph has no cycles at all? Sounds pretty dull. But graphs without cycles (called *acyclic graphs*) are probably the most important graphs of all when it comes to computer science.

5.7.1 Definitions

Definition 5.7.1. A connected acyclic graph is called a *tree*.

For example, Figure 5.25 shows an example of a 9-node tree.

The graph shown in Figure 5.26 is not a tree since it is not connected, but it is a forest. That's because, of course, it consists of a collection of trees.

Definition 5.7.2. If every connected component of a graph G is a tree, then G is a *forest*.

One of the first things you will notice about trees is that they tend to have a lot of nodes with degree one. Such nodes are called *leaves*.

Definition 5.7.3. A *leaf* is a node with degree 1 in a tree (or forest).

For example, the tree in Figure 5.25 has 5 leaves and the forest in Figure 5.26 has 4 leaves.

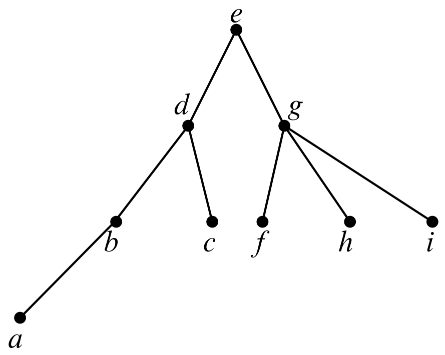


Figure 5.27 The tree from Figure 5.25 redrawn in a leveled fashion, with node E as the root.

Trees are a fundamental data structure in computer science. For example, information is often stored in tree-like data structures and the execution of many recursive programs can be modeled as the traversal of a tree. In such cases, it is often useful to draw the tree in a leveled fashion where the node in the top level is identified as the *root*, and where every edge joins a *parent* to a *child*. For example, we have redrawn the tree from Figure 5.25 in this fashion in Figure 5.27. In this example, node d is a child of node e and a parent of nodes b and c.

In the special case of *ordered binary trees*, every node is the parent of at most 2 children and the children are labeled as being a left-child or a right-child.

5.7.2 Properties

Trees have many unique properties. We have listed some of them in the following theorem.

Theorem 5.7.4. *Every tree has the following properties:*

1. *Any connected subgraph is a tree.*
2. *There is a unique simple path between every pair of vertices.*
3. *Adding an edge between nonadjacent nodes in a tree creates a graph with a cycle.*
4. *Removing any edge disconnects the graph.*
5. *If the tree has at least two vertices, then it has at least two leaves.*
6. *The number of vertices in a tree is one larger than the number of edges.*

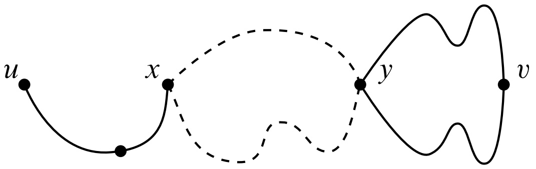


Figure 5.28 If there are two paths between u and v, the graph must contain a cycle.

Proof.

1. A cycle in a subgraph is also a cycle in the whole graph, so any subgraph of an acyclic graph must also be acyclic. If the subgraph is also connected, then by definition, it is a tree.

2. Since a tree is connected, there is at least one path between every pair of vertices. Suppose for the purposes of contradiction, that there are two different paths between some pair of vertices u and v. Beginning at u, let x be the first vertex where the paths diverge, and let y be the next vertex they share. (For example, see Figure 5.28.) Then there are two paths from x to y with no common edges, which defines a cycle. This is a contradiction, since trees are acyclic. Therefore, there is exactly one path between every pair of vertices.

3. An additional edge $\{u, v\}$ together with the unique path between u and v forms a cycle.

4. Suppose that we remove edge $\{u, v\}$. Since the tree contained a unique path between u and v, that path must have been $\{u, v\}$. Therefore, when that edge is removed, no path remains, and so the graph is not connected.

5. Let $v_1, \ldots, v_m$ be the sequence of vertices on a longest path in the tree. Then $m \geq 2$, since a tree with two vertices must contain at least one edge. There cannot be an edge $\{v_1, v_i\}$ for $2 < i \leq m$; otherwise, vertices $v_1, \ldots, v_i$ would from a cycle. Furthermore, there cannot be an edge $\{u, v_1\}$ where u is not on the path; otherwise, we could make the path longer. Therefore, the only edge incident to v_1 is $\{v_1, v_2\}$, which means that v_1 is a leaf. By a symmetric argument, v_m is a second leaf.

6. We use induction on the proposition $P(n) ::=$ there are $n - 1$ edges in any n-vertex tree.

   Base Case ($n = 1$): $P(1)$ is true since a tree with 1 node has 0 edges and $1 - 1 = 0$.

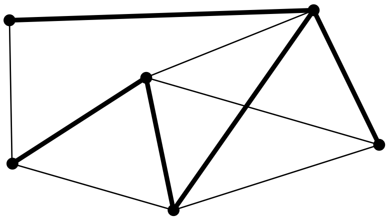


Figure 5.29 A graph where the edges of a spanning tree have been thickened.

> **Inductive step**: Now suppose that $P(n)$ is true and consider an $(n+1)$-vertex tree, T. Let v be a leaf of the tree. You can verify that deleting a vertex of degree 1 (and its incident edge) from any connected graph leaves a connected subgraph. So by part 1 of Theorem 5.7.4, deleting v and its incident edge gives a smaller tree, and this smaller tree has $n-1$ edges by induction. If we re-attach the vertex v and its incident edge, then we find that T has $n = (n+1) - 1$ edges. Hence, $P(n+1)$ is true, and the induction proof is complete. ■

Various subsets of properties in Theorem 5.7.4 provide alternative characterizations of trees, though we won't prove this. For example, a *connected* graph with a number of vertices one larger than the number of edges is necessarily a tree. Also, a graph with unique paths between every pair of vertices is necessarily a tree.

5.7.3 Spanning Trees

Trees are everywhere. In fact, every connected graph contains a subgraph that is a tree with the same vertices as the graph. This is a called a *spanning tree* for the graph. For example, Figure 5.29 is a connected graph with a spanning tree highlighted.

Theorem 5.7.5. *Every connected graph contains a spanning tree.*

Proof. By contradiction. Assume there is some connected graph G that has no spanning tree and let T be a connected subgraph of G, with the same vertices as G, and with the smallest number of edges possible for such a subgraph. By the assumption, T is not a spanning tree and so it contains some cycle:

$$\{v_0, v_1\}, \{v_1, v_2\}, \ldots, \{v_k, v_0\}$$

Suppose that we remove the last edge, $\{v_k, v_0\}$. If a pair of vertices x and y was joined by a path not containing $\{v_k, v_0\}$, then they remain joined by that path. On the other hand, if x and y were joined by a path containing $\{v_k, v_0\}$, then they

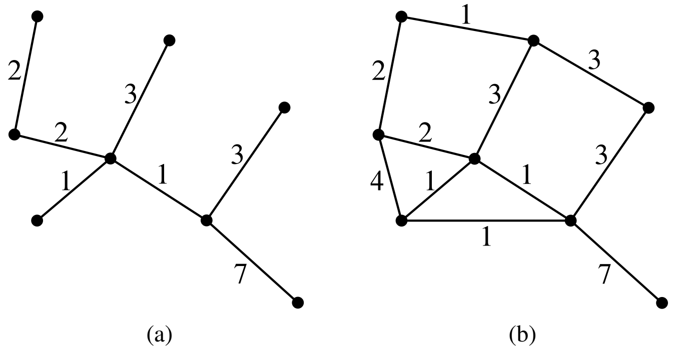


Figure 5.30 A spanning tree (a) with weight 19 for a graph (b).

remain joined by a walk containing the remainder of the cycle. By Lemma 5.4.2, they must also then be joined by a path. So all the vertices of G are still connected after we remove an edge from T. This is a contradiction, since T was defined to be a minimum size connected subgraph with all the vertices of G. So the theorem must be true. ■

5.7.4 Minimum Weight Spanning Trees

Spanning trees are interesting because they connect all the nodes of a graph using the smallest possible number of edges. For example the spanning tree for the 6-node graph shown in Figure 5.29 has 5 edges.

Spanning trees are very useful in practice, but in the real world, not all spanning trees are equally desirable. That's because, in practice, there are often costs associated with the edges of the graph.

For example, suppose the nodes of a graph represent buildings or towns and edges represent connections between buildings or towns. The cost to actually make a connection may vary a lot from one pair of buildings or towns to another. The cost might depend on distance or topography. For example, the cost to connect LA to NY might be much higher than that to connect NY to Boston. Or the cost of a pipe through Manhattan might be more than the cost of a pipe through a cornfield.

In any case, we typically represent the cost to connect pairs of nodes with a weighted edge, where the weight of the edge is its cost. The weight of a spanning tree is then just the sum of the weights of the edges in the tree. For example, the weight of the spanning tree shown in Figure 5.30 is 19.

The goal, of course, is to find the spanning tree with minimum weight, called the min-weight spanning tree (MST for short).

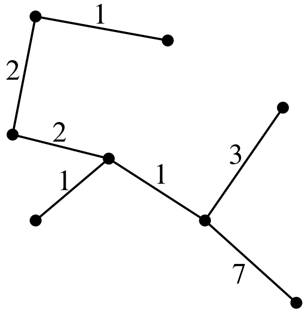


Figure 5.31 An MST with weight 17 for the graph in Figure 5.30(b).

Definition 5.7.6. The *min-weight spanning tree* (MST) of an edge-weighted graph G is the spanning tree of G with the smallest possible sum of edge weights.

Is the spanning tree shown in Figure 5.30(a) an MST of the weighted graph shown in Figure 5.30(b)? Actually, it is not, since the tree shown in Figure 5.31 is also a spanning tree of the graph shown in Figure 5.30(b), and this spanning tree has weight 17.

What about the tree shown in Figure 5.31? Is it an MST? It seems to be, but how do we prove it? In general, how do we find an MST? We could, of course, enumerate all trees, but this could take forever for very large graphs.

Here are two possible algorithms:

Algorithm 1. *Grow a tree one edge at a time by adding the minimum weight edge possible to the tree, making sure that you have a tree at each step.*

Algorithm 2. *Grow a subgraph one edge at a time by adding the minimum-weight edge possible to the subgraph, making sure that you have an acyclic subgraph at each step.*

For example, in the weighted graph we have been considering, we might run Algorithm 1 as follows. We would start by choosing one of the weight 1 edges, since this is the smallest weight in the graph. Suppose we chose the weight 1 edge on the bottom of the triangle of weight 1 edges in our graph. This edge is incident to two weight 1 edges, a weight 4 edge, a weight 7 edge, and a weight 3 edge. We would then choose the incident edge of minimum weight. In this case, one of the two weight 1 edges. At this point, we cannot choose the third weight 1 edge since this would form a cycle, but we can continue by choosing a weight 2 edge. We might end up with the spanning tree shown in Figure 5.32, which has weight 17, the smallest we've seen so far.

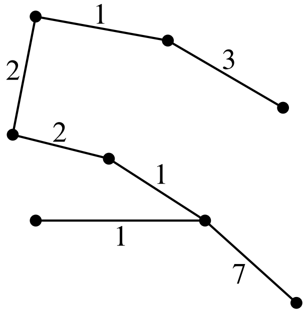


Figure 5.32 A spanning tree found by Algorithm 1.

Now suppose we instead ran Algorithm 2 on our graph. We might again choose the weight 1 edge on the bottom of the triangle of weight 1 edges in our graph. Now, instead of choosing one of the weight 1 edges it touches, we might choose the weight 1 edge on the top of the graph. Note that this edge still has minimum weight, and does not cause us to form a cycle, so Algorithm 2 can choose it. We would then choose one of the remaining weight 1 edges. Note that neither causes us to form a cycle. Continuing the algorithm, we may end up with the same spanning tree in Figure 5.32, though this need not always be the case.

It turns out that both algorithms work, but they might end up with different MSTs. The MST is not necessarily unique—indeed, if all edges of an n-node graph have the same weight ($= 1$), then all spanning trees have weight $n - 1$.

These are examples of greedy approaches to optimization. Sometimes it works and sometimes it doesn't. The good news is that it works to find the MST. In fact, both variations work. It's a little easier to prove it for Algorithm 2, so we'll do that one here.

Theorem 5.7.7. *For any connected, weighted graph G, Algorithm 2 produces an MST for G.*

Proof. The proof is a bit tricky. We need to show the algorithm terminates, that is, that if we have selected fewer than $n - 1$ edges, then we can always find an edge to add that does not create a cycle. We also need to show the algorithm creates a tree of minimum weight.

The key to doing all of this is to show that the algorithm never gets stuck or goes in a bad direction by adding an edge that will keep us from ultimately producing an MST. The natural way to prove this is to show that the set of edges selected at any point is contained in some MST—that is, we can always get to where we need to be. We'll state this as a lemma.

Lemma 5.7.8. *For any $m \geq 0$, let S consist of the first m edges selected by Algorithm 2. Then there exists some MST $T = (V, E)$ for G such that $S \subseteq E$, that is, the set of edges that we are growing is always contained in some MST.*

We'll prove this momentarily, but first let's see why it helps to prove the theorem. Assume the lemma is true. Then how do we know Algorithm 2 can always find an edge to add without creating a cycle? Well, as long as there are fewer than $n - 1$ edges picked, there exists some edge in $E - S$ and so there is an edge that we can add to S without forming a cycle. Next, how do we know that we get an MST at the end? Well, once $m = n - 1$, we know that S is an MST.

Ok, so the theorem is an easy corollary of the lemma. To prove the lemma, we'll use induction on the number of edges chosen by the algorithm so far. This is very typical in proving that an algorithm preserves some kind of invariant condition—induct on the number of steps taken, that is, the number of edges added.

Our inductive hypothesis $P(m)$ is the following: for any G and any set S of m edges initially selected by Algorithm 2, there exists an MST $T = (V, E)$ of G such that $S \subseteq E$.

For the base case, we need to show $P(0)$. In this case, $S = \emptyset$, so $S \subseteq E$ trivially holds for any MST $T = (V, E)$.

For the inductive step, we assume $P(m)$ holds and show that it implies $P(m+1)$. Let e denote the $(m+1)$st edge selected by Algorithm 2, and let S denote the first m edges selected by Algorithm 2. Let $T^* = (V^*, E^*)$ be the MST such that $S \subseteq E^*$, which exists by the inductive hypothesis. There are now two cases:

Case 1: $e \in E^*$, in which case $S \cup \{e\} \subseteq E^*$, and thus $P(m + 1)$ holds.

Case 2: $e \notin E^*$, as illustrated in Figure 5.33. Now we need to find a different MST that contains S and e.

What happens when we add e to T^*? Since T^* is a tree, we get a cycle. (Here we used part 3 of Theorem 5.7.4.) Moreover, the cycle cannot only contains edges in S, since e was chosen so that together with the edges in S, it does not form a cycle. This implies that $\{e\} \cup T^*$ contains a cycle that contains an edge e' of $E^* - S$. For example, such an e' is shown in Figure 5.33.

Note that the weight of e is at most that of e'. This is because Algorithm 2 picks the minimum weight edge that does not make a cycle with S. Since $e' \in T^*$, e' cannot make a cycle with S and if the weight of e were greater than the weight of e', Algorithm 2 would not have selected e ahead of e'.

Okay, we're almost done. Now we'll make an MST that contains $S \cup \{e\}$. Let $T^{**} = (V, E^{**})$ where $E^{**} = (E^* - \{e'\}) \cup \{e\}$, that is, we swap e and e' in T^*.

Claim 5.7.9. *T^{**} is an MST.*

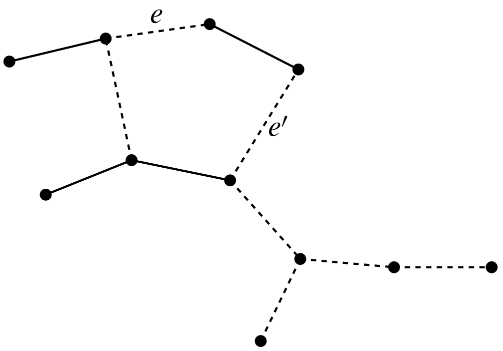


Figure 5.33 The graph formed by adding e to T^*. Edges of S are denoted with solid lines and edges of $E^* - S$ are denoted with dashed lines.

Proof of claim. We first show that T^{**} is a spanning tree. T^{**} is acyclic because it was produced by removing an edge from the only cycle in $T^* \cup \{e\}$. T^{**} is connected since the edge we deleted from $T^* \cup \{e\}$ was on a cycle. Since T^{**} contains all the nodes of G, it must be a spanning tree for G.

Now let's look at the weight of T^{**}. Well, since the weight of e was at most that of e', the weight of T^{**} is at most that of T^*, and thus T^{**} is an MST for G. ■

Since $S \cup \{e\} \subseteq E^{**}$, $P(m + 1)$ holds. Thus, Algorithm 2 must eventually produce an MST. This will happens when it adds $n - 1$ edges to the subgraph it builds. ■

So now we know for sure that the MST for our example graph has weight 17 since it was produced by Algorithm 2. And we have a fast algorithm for finding a minimum-weight spanning tree for any graph.

5.8 Planar Graphs

5.8.1 Drawing Graphs in the Plane

Suppose there are three dog houses and three human houses, as shown in Figure 5.34. Can you find a route from each dog house to each human house such that no route crosses any other route?

A *quadrapus* is a little-known animal similar to an octopus, but with four arms. Suppose there are five quadrapi resting on the sea floor, as shown in Figure 5.35.

Figure 5.34 Three dog houses and and three human houses. Is there a route from each dog house to each human house so that no pair of routes cross each other?

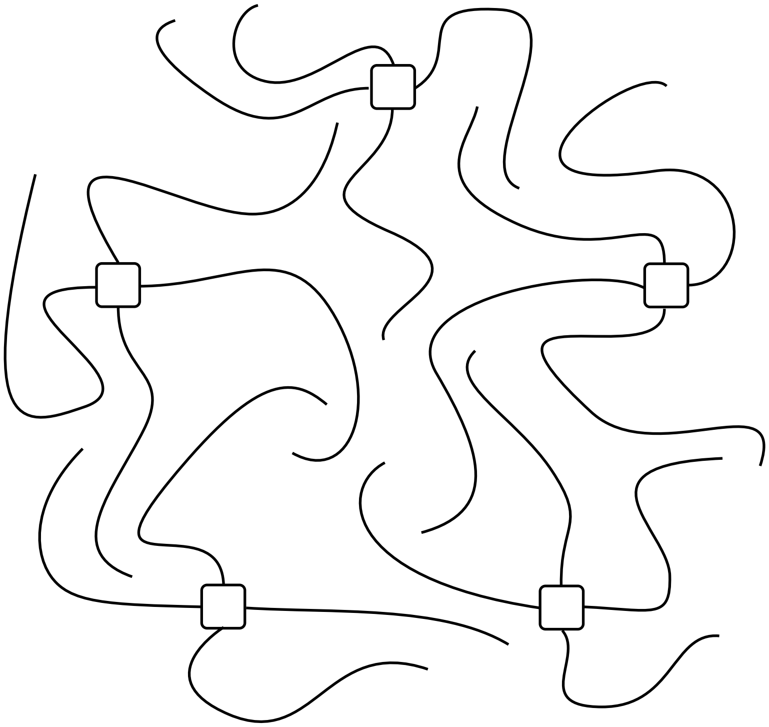


Figure 5.35 Five quadrapi (4-armed creatures).

Can each quadrapus simultaneously shake hands with every other in such a way that no arms cross?

Definition 5.8.1. A *drawing of a graph in the plane* consists of an assignment of vertices to distinct points in the plane and an assignment of edges to smooth, non-self-intersecting curves in the plane (whose endpoints are the nodes incident to the edge). The drawing is *planar* (that is, it is a *planar drawing*) if none of the curves "cross"—that is, if the only points that appear on more than one curve are the vertex points. A *planar graph* is a graph that has a planar drawing.

Thus, these two puzzles are asking whether the graphs in Figure 5.36 are planar; that is, whether they can be redrawn so that no edges cross. The first graph is called the *complete bipartite graph*, $K_{3,3}$, and the second is K_5.

In each case, the answer is, "No—but almost!" In fact, if you remove an edge from either of them, then the resulting graphs *can* be redrawn in the plane so that no edges cross. For example, we have illustrated the planar drawings for each resulting graph in Figure 5.37.

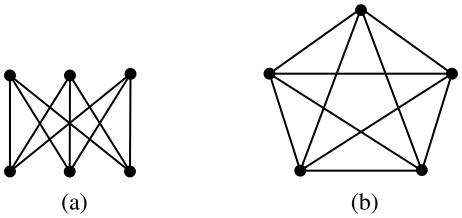


Figure 5.36 $K_{3,3}$ (a) and K_5 (b). Can you redraw these graphs so that no pairs of edges cross?

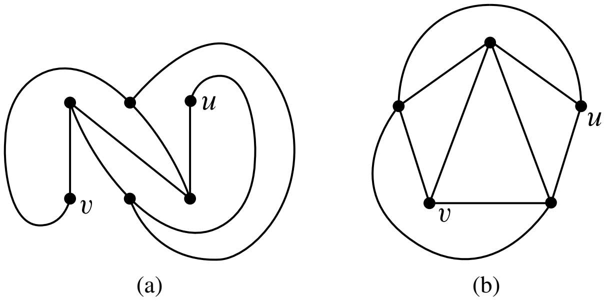


Figure 5.37 Planar drawings of $K_{3,3} - \{u, v\}$ (a) and $K_5 - \{u, v\}$ (b).

Planar drawings have applications in circuit layout and are helpful in displaying graphical data such as program flow charts, organizational charts, and scheduling conflicts. For these applications, the goal is to draw the graph in the plane with as few edge crossings as possible. (See the box on the following page for one such example.)

5.8.2 A Recursive Definition for Planar Graphs

Definition 5.8.1 is perfectly precise but has the challenge that it requires us to work with concepts such as a "smooth curve" when trying to prove results about planar graphs. The trouble is that we have not really laid the groundwork from geometry and topology to be able to reason carefully about such concepts. For example, we haven't really defined what it means for a curve to be smooth—we just drew a simple picture (for example, Figure 5.37) and hoped you would get the idea.

Relying on pictures to convey new concepts is generally not a good idea and can sometimes lead to disaster (or, at least, false proofs). Indeed, it is because of this issue that there have been so many false proofs relating to planar graphs over time.[18] Such proofs usually rely way too heavily on pictures and have way too many statements like,

> As you can see from Figure ABC, it must be that property XYZ holds for all planar graphs.

The good news is that there is another way to define planar graphs that uses only discrete mathematics. In particular, we can define the class of planar graphs as a recursive data type. In order to understand how it works, we first need to understand the concept of a *face* in a planar drawing.

Faces

In a planar drawing of a graph. the curves corresponding to the edges divide up the plane into connected regions. These regions are called the *continuous faces*[19] of the drawing. For example, the drawing in Figure 5.38 has four continuous faces. Face IV, which extends off to infinity in all directions, is called the *outside face*.

Notice that the vertices along the boundary of each of the faces in Figure 5.38 form a cycle. For example, labeling the vertices as in Figure 5.39, the cycles for the face boundaries are

$$abca \qquad abda \qquad bcdb \qquad acda. \tag{5.4}$$

[18] The false proof of the 4-Color Theorem for planar graphs is not the only example.

[19] Most texts drop the word *continuous* from the definition of a face. We need it to differentiate the connected region in the plane from the closed walk in the graph that bounds the region, which we will call a *discrete face*.

When wires are arranged on a surface, like a circuit board or microchip, crossings require troublesome three-dimensional structures. When Steve Wozniak designed the disk drive for the early Apple II computer, he struggled mightily to achieve a nearly planar design:

> For two weeks, he worked late each night to make a satisfactory design. When he was finished, he found that if he moved a connector he could cut down on feedthroughs, making the board more reliable. To make that move, however, he had to start over in his design. This time it only took twenty hours. He then saw another feedthrough that could be eliminated, and again started over on his design. "The final design was generally recognized by computer engineers as brilliant and was by engineering aesthetics beautiful. Woz later said, 'It's something you can only do if you're the engineer and the PC board layout person yourself. That was an artistic layout. The board has virtually no feedthroughs.' "[17]

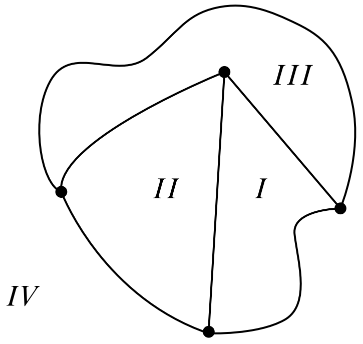


Figure 5.38 A planar drawing with four faces.

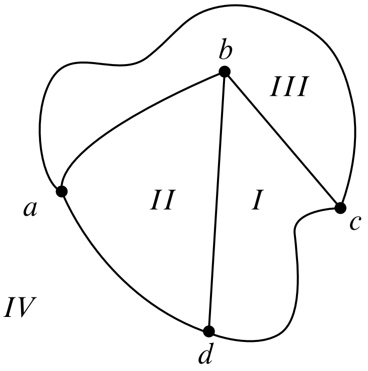


Figure 5.39 The drawing with labeled vertices.

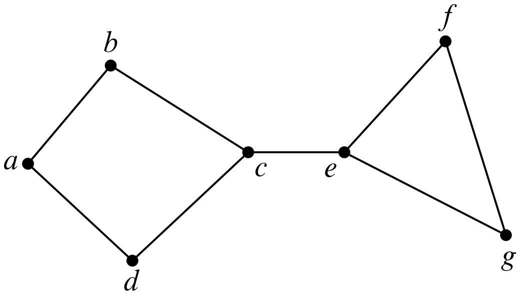


Figure 5.40 A planar drawing with a *bridge*, namely the edge $\{c, e\}$.

These four cycles correspond nicely to the four continuous faces in Figure 5.39. So nicely, in fact, that we can identify each of the faces in Figure 5.39 by its cycle. For example, the cycle $abca$ identifies face III. Hence, we say that the cycles in Equation 5.4 are the *discrete faces* of the graph in Figure 5.39. We use the term "discrete" since cycles in a graph are a discrete data type (as opposed to a region in the plane, which is a continuous data type).

Unfortunately, continuous faces in planar drawings are not always bounded by cycles in the graph—things can get a little more complicated. For example, consider the planar drawing in Figure 5.40. This graph has what we will call a *bridge* (namely, the edge $\{c, e\}$) and the outer face is

$$abcefgecda.$$

This is not a cycle, since it has to traverse the bridge $\{c, e\}$ twice, but it is a closed walk.

As another example, consider the planar drawing in Figure 5.41. This graph has what we will call a *dongle* (namely, the nodes v, x, y, and w, and the edges incident

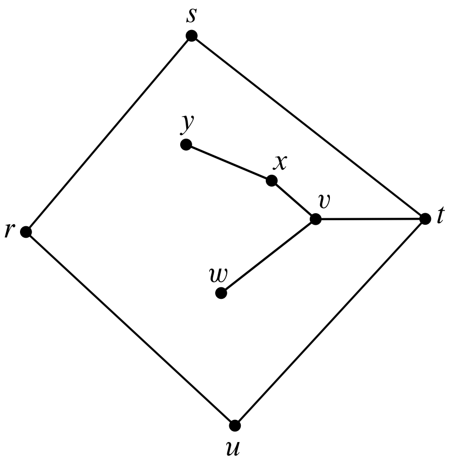


Figure 5.41 A planar drawing with a *dongle*, namely the subgraph with nodes v, w, x, y.

to them) and the inner face is

$$rstvxyxvwvtur.$$

This is not a cycle because it has to traverse *every* edge of the dongle twice—once "coming" and once "going," but once again, it is a closed walk.

It turns out that bridges and dongles are the only complications, at least for connected graphs. In particular, every continuous face in a planar drawing corresponds to a closed walk in the graph. We refer to such closed walks as the *discrete faces* of the drawing.

A Recursive Definition for Planar Embeddings

The association between the continuous faces of a planar drawing and closed walks will allow us to characterize a planar drawing in terms of the closed walks that bound the continuous faces. In particular, it leads us to the discrete data type of *planar embeddings* that we can use in place of continuous planar drawings. Namely, we'll define a planar embedding recursively to be the set of boundary-tracing closed walks that we could get by drawing one edge after another.

Definition 5.8.2. A *planar embedding* of a *connected* graph consists of a nonempty set of closed walks of the graph called the *discrete faces* of the embedding. Planar embeddings are defined recursively as follows:

Base case: If G is a graph consisting of a single vertex v, then a planar embedding of G has one discrete face, namely the length zero closed walk v.

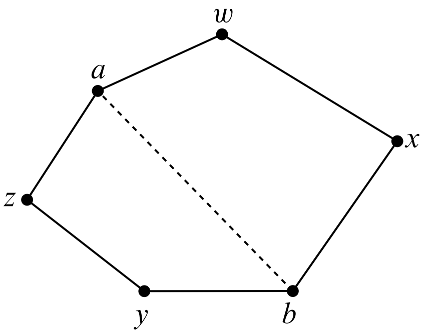


Figure 5.42 The "split a face" case.

Constructor Case (split a face): Suppose G is a connected graph with a planar embedding, and suppose a and b are distinct, nonadjacent vertices of G that appear on some discrete face γ of the planar embedding. That is, γ is a closed walk of the form

$$a \ldots b \ldots a.$$

Then the graph obtained by adding the edge $\{a, b\}$ to the edges of G has a planar embedding with the same discrete faces as G, except that face γ is replaced by the two discrete faces[20]

$$a \ldots ba \quad \text{and} \quad ab \ldots a,$$

as illustrated in Figure 5.42.

Constructor Case (add a bridge): Suppose G and H are connected graphs with planar embeddings and disjoint sets of vertices. Let a be a vertex on a discrete face, γ, in the embedding of G. That is, γ is of the form

$$a \ldots a.$$

Similarly, let b be a vertex on a discrete face, δ, in the embedding of H. So δ is of the form

$$b \cdots b.$$

Then the graph obtained by connecting G and H with a new edge, $\{a, b\}$, has a planar embedding whose discrete faces are the union of the discrete faces of G and

[20] There is a special case of this rule. If G is a line graph beginning with a and ending with b, then the cycles into which γ splits are actually the same. That's because adding edge $\{a, b\}$ creates a simple cycle graph, C_n, that divides the plane into an "inner" and an "outer" region with the same border. In order to maintain the correspondence between continuous faces and discrete faces, we have to allow two "copies" of this same cycle to count as discrete faces.

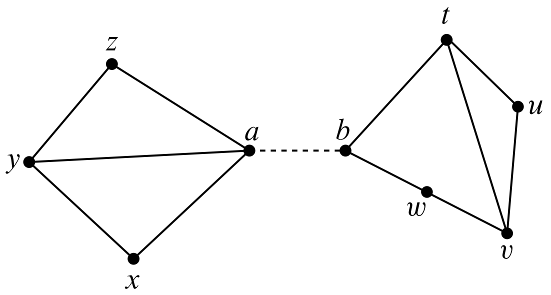


Figure 5.43 The "add a bridge" case.

H, except that faces γ and δ are replaced by one new face

$$a \ldots ab \cdots ba.$$

This is illustrated in Figure 5.43, where the faces of G and H are:

$$G : \{axyza,\ axya,\ ayza\} \qquad H : \{btuvwb,\ btvwb,\ tuvt\},$$

and after adding the bridge $\{a, b\}$, there is a single connected graph with faces

$$\{axyzabtuvwba,\ axya,\ ayza,\ btvwb,\ tuvt\}.$$

Does It Work?

Yes! In general, a graph is planar if and only if each of its connected components has a planar embedding as defined in Definition 5.8.2. Unfortunately, proving this fact requires a bunch of mathematics that we don't cover in this text—stuff like geometry and topology. Of course, that is why we went to the trouble of including Definition 5.8.2—we don't want to deal with that stuff in this text and now that we have a recursive definition for planar graphs, we won't need to. That's the good news.

The bad news is that Definition 5.8.2 looks a lot more complicated than the intuitively simple notion of a drawing where edges don't cross. It seems like it would be easier to stick to the simple notion and give proofs using pictures. Perhaps so, but your proofs are more likely to be complete and correct if you work from the discrete Definition 5.8.2 instead of the continuous Definition 5.8.1.

Where Did the Outer Face Go?

Every planar drawing has an immediately-recognizable outer face—its the one that goes to infinity in all directions. But where is the outer face in a planar embedding?

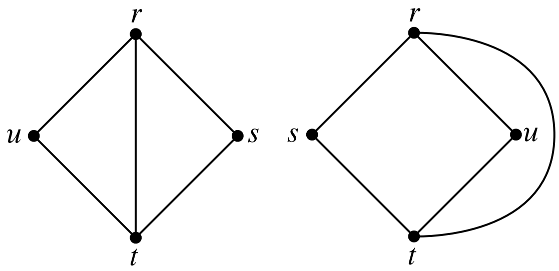


Figure 5.44 Two illustrations of the same embedding.

There isn't one! That's because there really isn't any need to distinguish one. In fact, a planar embedding could be drawn with any given face on the outside. An intuitive explanation of this is to think of drawing the embedding on a *sphere* instead of the plane. Then any face can be made the outside face by "puncturing" that face of the sphere, stretching the puncture hole to a circle around the rest of the faces, and flattening the circular drawing onto the plane.

So pictures that show different "outside" boundaries may actually be illustrations of the same planar embedding. For example, the two embeddings shown in Figure 5.44 are really the same.

This is what justifies the "add a bridge" case in Definition 5.8.2: whatever face is chosen in the embeddings of each of the disjoint planar graphs, we can draw a bridge between them without needing to cross any other edges in the drawing, because we can assume the bridge connects two "outer" faces.

5.8.3 Euler's Formula

The value of the recursive definition is that it provides a powerful technique for proving properties of planar graphs, namely, structural induction. For example, we will now use Definition 5.8.2 and structural induction to establish one of the most basic properties of a connected planar graph; namely, the number of vertices and edges completely determines the number of faces in every possible planar embedding of the graph.

Theorem 5.8.3 (Euler's Formula)**.** *If a connected graph has a planar embedding, then*

$$v - e + f = 2$$

where v is the number of vertices, e is the number of edges, and f is the number of faces.

For example, in Figure 5.38, $|V| = 4$, $|E| = 6$, and $f = 4$. Sure enough, $4 - 6 + 4 = 2$, as Euler's Formula claims.

Proof. The proof is by structural induction on the definition of planar embeddings. Let $P(\mathcal{E})$ be the proposition that $v - e + f = 2$ for an embedding, $\mathcal{E}$.

Base case: ($\mathcal{E}$ is the one-vertex planar embedding). By definition, $v = 1$, $e = 0$, and $f = 1$, so $P(\mathcal{E})$ indeed holds.

Constructor case (split a face): Suppose G is a connected graph with a planar embedding, and suppose a and b are distinct, nonadjacent vertices of G that appear on some discrete face, $\gamma = a \ldots b \cdots a$, of the planar embedding.

Then the graph obtained by adding the edge $\{a, b\}$ to the edges of G has a planar embedding with one more face and one more edge than G. So the quantity $v - e + f$ will remain the same for both graphs, and since by structural induction this quantity is 2 for G's embedding, it's also 2 for the embedding of G with the added edge. So P holds for the constructed embedding.

Constructor case (add bridge): Suppose G and H are connected graphs with planar embeddings and disjoint sets of vertices. Then connecting these two graphs with a bridge merges the two bridged faces into a single face, and leaves all other faces unchanged. So the bridge operation yields a planar embedding of a connected graph with $v_G + v_H$ vertices, $e_G + e_H + 1$ edges, and $f_G + f_H - 1$ faces. Since

$$\begin{aligned}
&(v_G + v_H) - (e_G + e_H + 1) + (f_G + f_H - 1) \\
&\quad = (v_G - e_G + f_G) + (v_H - e_H + f_H) - 2 \\
&\quad = (2) + (2) - 2 && \text{(by structural induction hypothesis)} \\
&\quad = 2,
\end{aligned}$$

$v - e + f$ remains equal to 2 for the constructed embedding. That is, $P(E)$ also holds in this case.

This completes the proof of the constructor cases, and the theorem follows by structural induction. ■

5.8.4 Bounding the Number of Edges in a Planar Graph

Like Euler's formula, the following lemmas follow by structural induction from Definition 5.8.2.

Lemma 5.8.4. *In a planar embedding of a connected graph, each edge is traversed once by each of two different faces, or is traversed exactly twice by one face.*

Lemma 5.8.5. *In a planar embedding of a connected graph with at least three vertices, each face is of length at least three.*

Combining Lemmas 5.8.4 and 5.8.5 with Euler's Formula, we can now prove that planar graphs have a limited number of edges:

Theorem 5.8.6. *Suppose a connected planar graph has $v \geq 3$ vertices and e edges. Then*

$$e \leq 3v - 6.$$

Proof. By definition, a connected graph is planar iff it has a planar embedding. So suppose a connected graph with v vertices and e edges has a planar embedding with f faces. By Lemma 5.8.4, every edge is traversed exactly twice by the face boundaries. So the sum of the lengths of the face boundaries is exactly $2e$. Also by Lemma 5.8.5, when $v \geq 3$, each face boundary is of length at least three, so this sum is at least $3f$. This implies that

$$3f \leq 2e. \tag{5.5}$$

But $f = e - v + 2$ by Euler's formula, and substituting into (5.5) gives

$$\begin{aligned} 3(e - v + 2) &\leq 2e \\ e - 3v + 6 &\leq 0 \\ e &\leq 3v - 6 \end{aligned}$$ ■

5.8.5 Returning to K_5 and $K_{3,3}$

Theorem 5.8.6 lets us prove that the quadrapi can't all shake hands without crossing. Representing quadrapi by vertices and the necessary handshakes by edges, we get the complete graph, K_5. Shaking hands without crossing amounts to showing that K_5 is planar. But K_5 is connected, has 5 vertices and 10 edges, and $10 > 3 \cdot 5 - 6$. This violates the condition of Theorem 5.8.6 required for K_5 to be planar, which proves

Corollary 5.8.7. *K_5 is not planar.*

We can also use Euler's Formula to show that $K_{3,3}$ is not planar. The proof is similar to that of Theorem 5.8.6 except that we use the additional fact that $K_{3,3}$ is a bipartite graph.

Theorem 5.8.8. *$K_{3,3}$ is not planar.*

Proof. By contradiction. Assume $K_{3,3}$ is planar and consider any planar embedding of $K_{3,3}$ with f faces. Since $K_{3,3}$ is bipartite, we know by Theorem 5.6.2 that $K_{3,3}$ does not contain any closed walks of odd length. By Lemma 5.8.5, every face has length at least 3. This means that every face in any embedding of $K_{3,3}$ must have length at least 4. Plugging this fact into the proof of Theorem 5.8.6, we find that the sum of the lengths of the face boundaries is exactly $2e$ and at least $4f$. Hence,

$$4f \leq 2e$$

for any bipartite graph.

Plugging in $e = 9$ and $v = 6$ for $K_{3,3}$ in Euler's Formula, we find that

$$f = 2 + e - v = 5.$$

But

$$4 \cdot 5 \not\leq 2 \cdot 9,$$

and so we have a contradiction. Hence $K_{3,3}$ must not be planar. ■

5.8.6 Another Characterization for Planar Graphs

We did not choose to pick on K_5 and $K_{3,3}$ because of their application to dog houses or quadrapi shaking hands. Rather, we selected these graphs as examples because they provide another way to characterize the set of planar graphs.

Theorem 5.8.9 (Kuratowski)**.** *A graph is not planar if and only if it contains K_5 or $K_{3,3}$ as a minor.*

Definition 5.8.10. A *minor* of a graph G is a graph that can be obtained by repeatedly[21] deleting vertices, deleting edges, and merging *adjacent* vertices of G. *Merging* two adjacent vertices, n_1 and n_2 of a graph means deleting the two vertices and then replacing them by a new "merged" vertex, m, adjacent to all the vertices that were adjacent to either of n_1 or n_2, as illustrated in Figure 5.45.

For example, Figure 5.46 illustrates why C_3 is a minor of the graph in Figure 5.46(a). In fact C_3 is a minor of a connected graph G if and only if G is not a tree.

We will not prove Theorem 5.8.9 here, nor will we prove the following handy facts, which are obvious given the continuous Definition 5.8.1, and which can be proved using the recursive Definition 5.8.2.

Lemma 5.8.11. *Deleting an edge from a planar graph leaves another planar graph.*

Corollary 5.8.12. *Deleting a vertex from a planar graph, along with all its incident edges, leaves another planar graph.*

Theorem 5.8.13. *Any subgraph of a planar graph is planar.*

Theorem 5.8.14. *Merging two adjacent vertices of a planar graph leaves another planar graph.*

[21] The three operations can be performed in any order and in any quantities, or not at all.

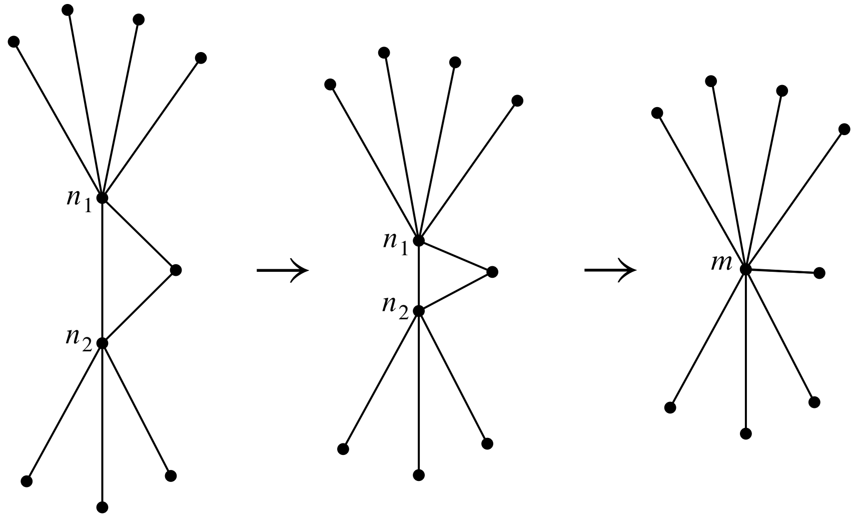


Figure 5.45 Merging adjacent vertices n_1 and n_2 into new vertex, m.

5.8.7 Coloring Planar Graphs

We've covered a lot of ground with planar graphs, but not nearly enough to prove the famous 4-color theorem. But we can get awfully close. Indeed, we have done almost enough work to prove that every planar graph can be colored using only 5 colors. We need only one more lemma:

Lemma 5.8.15. *Every planar graph has a vertex of degree at most five.*

Proof. By contradiction. If every vertex had degree at least 6, then the sum of the vertex degrees is at least $6v$, but since the sum of the vertex degrees equals $2e$, by the Handshake Lemma (Lemma 5.2.1), we have $e \geq 3v$ contradicting the fact that $e \leq 3v - 6 < 3v$ by Theorem 5.8.6. ■

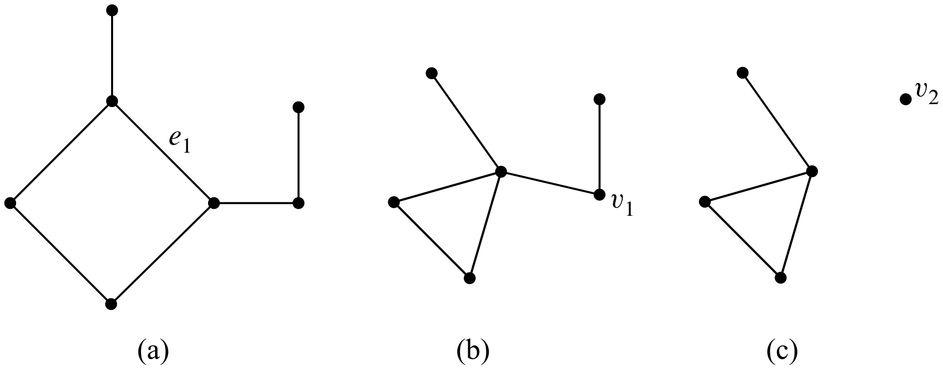
Theorem 5.8.16. *Every planar graph is five-colorable.*

Proof. The proof will be by strong induction on the number, v, of vertices, with induction hypothesis:

> Every planar graph with v vertices is five-colorable.

Base cases ($v \leq 5$): immediate.

Inductive case: Suppose G is a planar graph with $v + 1$ vertices. We will describe a five-coloring of G.



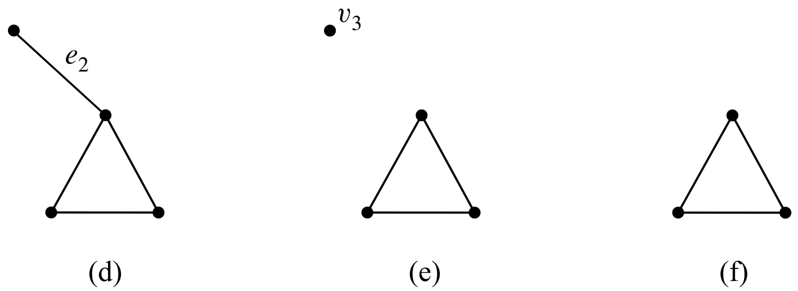


Figure 5.46 One method by which the graph in (a) can be reduced to C_3 (f), thereby showing that C_3 is a minor of the graph. The steps are: merging the nodes incident to e_1 (b), deleting v_1 and all edges incident to it (c), deleting v_2 (d), deleting e_2, and deleting v_3 (f).

First, choose a vertex, g, of G with degree at most 5; Lemma guarantees there will be such a vertex.

Case 1: ($\deg(g) < 5$): Deleting g from G leaves a graph, H, that is planar by Corollary 5.8.12, and, since H has v vertices, it is five-colorable by induction hypothesis. Now define a five coloring of G as follows: use the five-coloring of H for all the vertices besides g, and assign one of the five colors to g that is not the same as the color assigned to any of its neighbors. Since there are fewer than 5 neighbors, there will always be such a color available for g.

Case 2: ($\deg(g) = 5$): If the five neighbors of g in G were all adjacent to each other, then these five vertices would form a nonplanar subgraph isomorphic to K_5, contradicting Theorem 5.8.13 (since K_5 is not planar). So there must be two neighbors, n_1 and n_2, of g that are not adjacent. Now merge n_1 and g into a new vertex, m. In this new graph, n_2 is adjacent to m, and the graph is planar by Theorem 5.8.14. So we can then merge m and n_2 into a another new vertex, m', resulting in a new graph, G', which by Theorem 5.8.14 is also planar. Since G' has $v - 1$ vertices, it is five-colorable by the induction hypothesis.

Define a five coloring of G as follows: use the five-coloring of G' for all the vertices besides g, n_1 and n_2. Next assign the color of m' in G' to be the color of the neighbors n_1 and n_2. Since n_1 and n_2 are not adjacent in G, this defines a proper five-coloring of G except for vertex g. But since these two neighbors of g have the same color, the neighbors of g have been colored using fewer than five colors altogether. So complete the five-coloring of G by assigning one of the five colors to g that is not the same as any of the colors assigned to its neighbors.

■

5.8.8 Classifying Polyhedra

The Pythagoreans had two great mathematical secrets, the irrationality of $\sqrt{2}$ and a geometric construct that we're about to rediscover!

A *polyhedron* is a convex, three-dimensional region bounded by a finite number of polygonal faces. If the faces are identical regular polygons and an equal number of polygons meet at each corner, then the polyhedron is *regular*. Three examples of regular polyhedra are shown in Figure 5.34: the tetrahedron, the cube, and the octahedron.

We can determine how many more regular polyhedra there are by thinking about planarity. Suppose we took *any* polyhedron and placed a sphere inside it. Then we could project the polyhedron face boundaries onto the sphere, which would give an image that was a planar graph embedded on the sphere, with the images of the

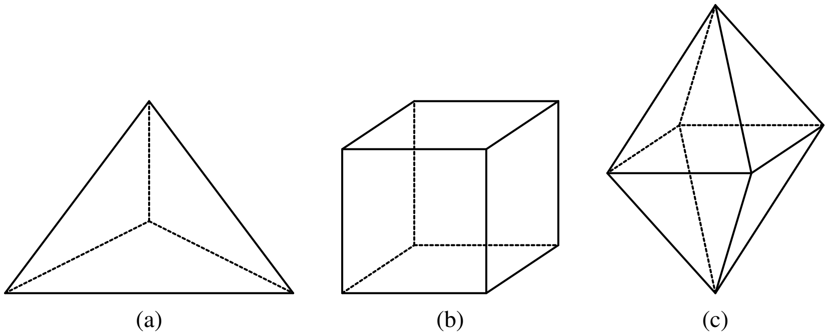


Figure 5.47 The tetrahedron (a), cube (b), and octahedron (c).

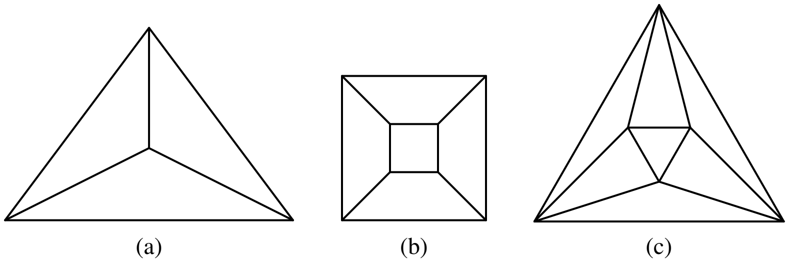


Figure 5.48 Planar embeddings of the tetrahedron (a), cube (b, and octahedron (c).

corners of the polyhedron corresponding to vertices of the graph. We’ve already observed that embeddings on a sphere are the same as embeddings on the plane, so Euler’s formula for planar graphs can help guide our search for regular polyhedra.

For example, planar embeddings of the three polyhedra in Figure 5.34 are shown in Figure 5.48.

Let m be the number of faces that meet at each corner of a polyhedron, and let n be the number of edges on each face. In the corresponding planar graph, there are m edges incident to each of the v vertices. By the Handshake Lemma 5.2.1, we know:

$$mv = 2e.$$

Also, each face is bounded by n edges. Since each edge is on the boundary of two faces, we have:

$$nf = 2e$$

Solving for v and f in these equations and then substituting into Euler’s formula

n	m	v	e	f	polyhedron
3	3	4	6	4	tetrahedron
4	3	8	12	6	cube
3	4	6	12	8	octahedron
3	5	12	30	20	icosahedron
5	3	20	30	12	dodecahedron

Figure 5.49 The only possible regular polyhedra.

gives:

$$\frac{2e}{m} - e + \frac{2e}{n} = 2$$

which simplifies to

$$\frac{1}{m} + \frac{1}{n} = \frac{1}{e} + \frac{1}{2} \tag{5.6}$$

Equation 5.6 places strong restrictions on the structure of a polyhedron. Every nondegenerate polygon has at least 3 sides, so $n \geq 3$. And at least 3 polygons must meet to form a corner, so $m \geq 3$. On the other hand, if either n or m were 6 or more, then the left side of the equation could be at most $1/3 + 1/6 = 1/2$, which is less than the right side. Checking the finitely-many cases that remain turns up only five solutions, as shown in Figure 5.49. For each valid combination of n and m, we can compute the associated number of vertices v, edges e, and faces f. And polyhedra with these properties do actually exist. The largest polyhedron, the dodecahedron, was the other great mathematical secret of the Pythagorean sect.

The 5 polyhedra in Figure 5.49 are the only possible regular polyhedra. So if you want to put more than 20 geocentric satellites in orbit so that they *uniformly* blanket the globe—tough luck!

6.042J / 18.062J Mathematics for Computer Science
Fall 2010

Trees

MAT230

Discrete Mathematics

Fall 2019

Outline

1. What is a Tree?

2. Spanning Trees

3. Rooted Trees

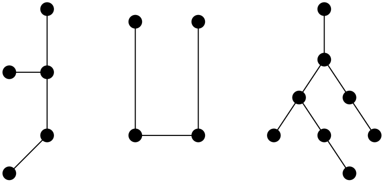
4. Binary Trees

Definitions

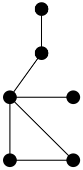
Definition

A **tree** is a connected undirected graph that has no cycles or self-loops.

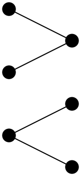
Some examples:



Trees



Not a tree:
has a cycle



Not a tree:
disconnected

Definition

A **forest** is an undirected graph whose components are all trees.

A Theorem About Trees

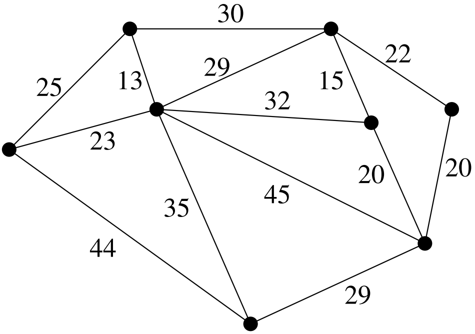
Theorem

Let $T = (V, E)$ be an undirected graph with no self-loops and $|V| = n$. Then the following statements are equivalent:

1. *T is a tree.*
2. *Any two vertices of T are connected by exactly one path.*
3. *T is connected and every edge is an* **isthmus** *(its removal disconnects T).*
4. *T contains no cycles, but the addition of any new edge creates exactly one cycle.*
5. *T is connected and has $n - 1$ edges.*

Spanning Trees and Minimum Spanning Trees

Suppose the following graph represents distance in miles between towns. The towns are to be connected by high-speed network cable. Assuming the cost of cables is proportional to their length and bandwidth is not a limiting factor, the most cost effective network will be a tree that *spans* the graph.



Definitions

Definition

Let $G = (V, E)$ be a connected undirected graph. A **spanning set** for G is a subset E' of E such that (V, E') is connected.

Definition

Let G be a connected undirected graph. The subgraph T is a **spanning tree for** G if T is a tree and every node in G is a node in T.

Definition

If G is a weighted graph, then T is a **minimal spanning tree of** G if it is a spanning tree and no other spanning tree of G has smaller total weight.

Minimal Spanning Trees (MST)

Suppose $G = (V, E, w)$ is a weighted connected undirected graph. The **minimal spanning tree problem** is to find a spanning tree $T = (V, E')$ for G such that $\displaystyle\sum_{e \in E'} w(e)$ is as small as possible.

Unlike the Traveling Salesman Problem, solving the MST problem is relatively easy. We consider two algorithms.

Prim's Algorithm

Let $G = (V, E, w)$ be a weighted connected undirected graph.

```
1. Pick any vertex $v \in V$.
2. $V' = \{v\}$
3. $V_0 = V - \{v\}$
4. $E' = \{\}$
5. While $V' \neq V$ do:
    a. Find $u \in V'$ and $v \in V_0$ such that edge $e = \{u, v\}$ has minimum weight
    b. $E' = E' \cup e$
    c. $V' = V' \cup \{v\}$
    d. $V_0 = V_0 - \{v\}$
```

Upon termination, $T = (V, E')$ is a minimum spanning tree for G.

Kruskal's Algorithm

Let $G = (V, E, w)$ be a weighted connected undirected graph.

1. Find edge $e \in E$ with minimum weight
2. $E' = \{e\}$
3. $E_0 = E - \{e\}$
4. $V' = \{v : v \text{ is a vertex for which } e \text{ is an incident edge}\}$
5. While $V' \neq V$ or (V', E') not connected do:
    - a. Find edge $e \in E_0$ with minimum weight that will not complete a cycle in (V', E').
    - b. $E_0 = E_0 - e$
    - c. $E' = E' \cup e$
    - d. $V' = V' \cup \{v : v \text{ is a vertex for which } e \text{ is an incident edge}\}$

Upon termination, $T = (V, E')$ is a minimum spanning tree for G.

Rooted Trees

Definition

Every nonempty tree can have a particular vertex called a **root**. In a rooted tree, the root is at **level 0**. The **level** of all other vertices is one greater than the number of edges in the walk from the root to the vertex. The **height** of a tree is the number of levels in the tree.

Definition

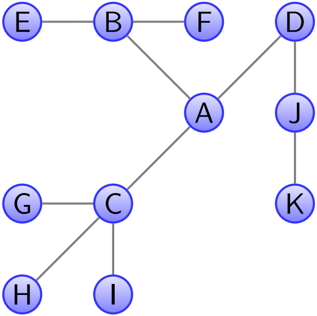
A vertex u in a rooted tree is a **parent** of a vertex v if v is adjacent to u and the level of v is one greater than the level of u. In this case v is a **child** of u. Two or more vertices are **siblings** if they have the same parent.

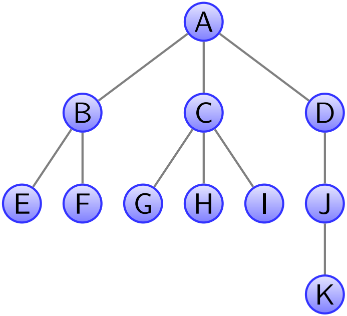
Definition

Nonroot vertices of degree 1 in a tree are called the **leaves** of the tree. All other vertices are called **internal vertices**.

Rooted Trees

On the left is a typical tree. On the right is the same tree redrawn with vertex A identified as the root.



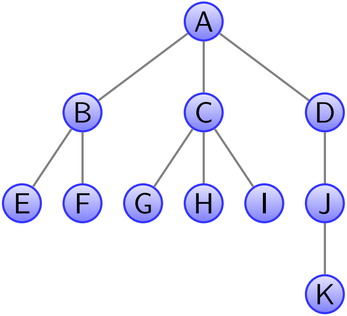


Vertex C is the parent of vertices G, H, and I. Vertices E and F are children of vertex B and so are siblings. Vertices E, F, G, H, I, and J are all at level 2. Vertices E, F, G, H, I, and K are leaves while A, B, C, D, and J are internal vertices.

Definition

A **subtree** is connected component of a rooted tree T that does not include the root of T. Every subtree is itself a rooted tree.

- This tree has subtrees rooted at B, C, and D.
- There is also a subtree rooted at J.
- Finally, each leaf is a subtree.

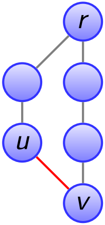


Kruskal's Algorithm (Version 2)

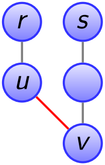
Working with rooted trees makes Kruskal's algorithm easier to describe.

Suppose there is an edge e between vertices u and v.

If u and v are both in a tree rooted at r then the edge e will create a cycle.



However, if u is in a tree rooted at r and v is in a tree rooted at s, adding edge e does not create a cycle, but merges the two trees.



In this case, we can choose either r or s to be the root of the merged tree.

Kruskal's Algorithm (Version 2)

Let $G = (V, E, w)$ be a weighted connected undirected graph and suppose we have function rootof(v) that returns the root of the tree containing vertex v.

1. Create sorted edge list E_L (sort by increasing weight)
2. Initialize a list R of rooted trees with each vertex in the tree as the root of its own tree.
3. $E' = \{\}$
4. While R contains more than one entry **and** E_L not empty do:
    - a. remove edge $e = \{u, v\}$ from E_L
    - b. if rootof(u) $\neq$ rootof(v) then
        - ★ $E' = E' \cup \{e\}$
        - ★ Merge tree rooted at rootof(u) into tree rooted at rootof(v) (or vice-versa). This reduces the number of entries in R by one.

Upon termination, $T = (V, E')$ is a minimum spanning tree for G.

Binary Trees

Definition

A **binary tree** T is a tree that has zero or more nodes in which each node has at most two **children**. Each nonleaf node has **left subtree** and a **right subtree**, either of which may be empty.

Question: How many vertices can be found on level k of a binary tree?

Binary Trees

Definition

A **binary tree** T is a tree that has zero or more nodes in which each node has at most two **children**. Each nonleaf node has **left subtree** and a **right subtree**, either of which may be empty.

Question: How many vertices can be found on level k of a binary tree?

Answer: $\boxed{2^k;}$ the number of possible vertices doubles on each level.

Binary Trees

Definition

A **binary tree** T is a tree that has zero or more nodes in which each node has at most two **children**. Each nonleaf node has **left subtree** and a **right subtree**, either of which may be empty.

Question: How many vertices can be found on level k of a binary tree?
Answer: $\boxed{2^k;}$ the number of possible vertices doubles on each level.

Question: How many vertices can be found in a binary tree of height h?

Binary Trees

Definition

A **binary tree** T is a tree that has zero or more nodes in which each node has at most two **children**. Each nonleaf node has **left subtree** and a **right subtree**, either of which may be empty.

Question: How many vertices can be found on level k of a binary tree?
Answer: $\boxed{2^k;}$ the number of possible vertices doubles on each level.

Question: How many vertices can be found in a binary tree of height h?
Answer: $\boxed{2^{k+1} - 1;}$ the sum of the powers of 2 from 2^0 up to 2^k.

Binary Tree Traversal

A **traversal** of a tree visit each vertex of the tree in some order determined by the connectivity within the tree. There are three common traversals of binary trees, each described by a recursive algorithm.

Preorder Traversal:

1. Visit the root of the tree
2. Preorder traverse the left subtree
3. Preorder traverse the right subtree

Inorder Traversal:

1. Inorder traverse the left subtree
2. Visit the root of the tree
3. Inorder traverse the right subtree

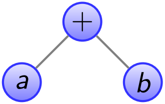
Postorder Traversal:

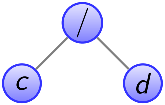
1. Postorder traverse the left subtree
2. Postorder traverse the right subtree
3. Visit the root of the tree

Expression Trees

The arithmetic operations $+$, $-$, $\times$, and $/$ are called *binary operators* since they each take two operands.

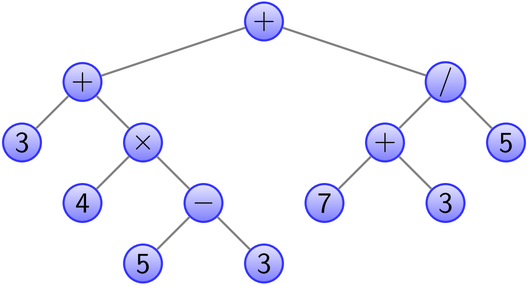
We can diagram them with trees. For example, trees for $a + b$ and c/d are





Expression Trees

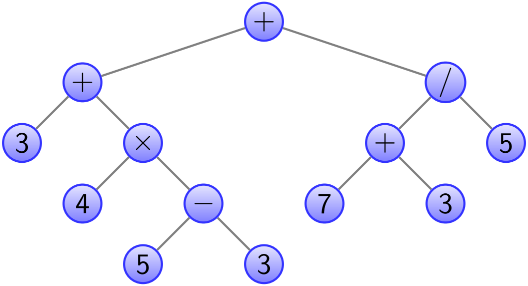
Consider the arithmetic expression $3 + 4(5 - 3) + (7 + 3)/5$. A tree for this expression is



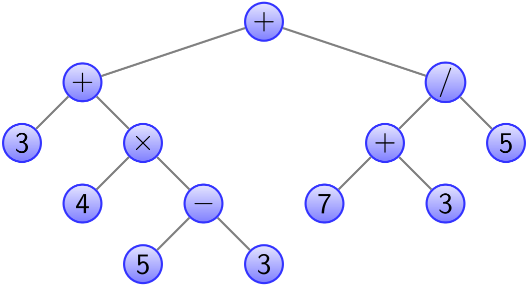
This tree is not unique, other binary trees could be used to represent the same expression.

Construct a preorder and postorder traversal of this tree.

Expression Trees

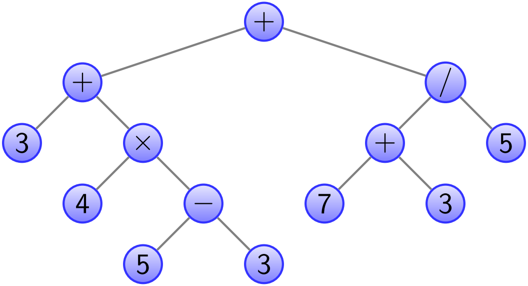


Expression Trees



Preorder traversal: $+ \quad + \quad 3 \quad \times \quad 4 \quad - \quad 5 \quad 3 \quad / \quad + \quad 7 \quad 3 \quad 5$

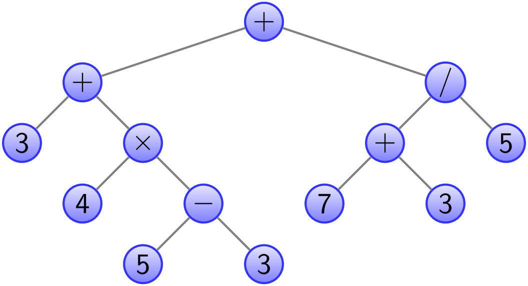
Expression Trees



Preorder traversal: $+ \quad + \quad 3 \quad \times \quad 4 \quad - \quad 5 \quad 3 \quad / \quad + \quad 7 \quad 3 \quad 5$

Postorder traversal: $3 \quad 4 \quad 5 \quad 3 \quad - \quad \times \quad + \quad 7 \quad 3 \quad + \quad 5 \quad / \quad +$

Expression Trees



Preorder traversal: $+ \quad + \quad 3 \quad \times \quad 4 \quad - \quad 5 \quad 3 \quad / \quad + \quad 7 \quad 3 \quad 5$

Postorder traversal: $3 \quad 4 \quad 5 \quad 3 \quad - \quad \times \quad + \quad 7 \quad 3 \quad + \quad 5 \quad / \quad +$

Both of these can be evaluated unambiguously without the need for parentheses.

Rajat Mittal [*]

Lecture notes: Abstract algebra

April 20, 2015

Springer

[*] Thanks to the book from Dummit and Foote and the book from Norman Biggs.
IIT Kanpur

1

Introduction

Please look at the course policies mentioned in the course homepage. Most importantly, any immoral behavior like cheating and fraud will be punished with extreme measures and without any exception.

1.1 What is this course about?

Take a look at the following questions.

- Give a number n which leaves a remainder of 20 when divided by 23 and 62 when divided by 83.
- How many different necklaces can you form with 2 black beads and 8 white beads? How many necklaces can you form with blue, green and black beads?
- What are the last two digits of of a^{40} when a is not divisible by 2 or 5?
- We know that there is an explicit formula for the roots of quadratic equation $ax^2+bx+c=0$, $\frac{-b\pm\sqrt{b^2-4ac}}{2a}$. Similarly there are explicit formulas for degree 3 and degree 4 equations. Why don't we have something for degree 5?
- When does the equations of the form $x - y = z$ make sense? If x is a natural number or an integer or a matrix or an apple or a permutation?

When we look at these questions, they seem unrelated and seem to have no common thread. Mathematicians realized long time back that problems in algebra, number theory and even geometry can be solved using very similar techniques. They were interested in finding out the common element among these proofs and were interested in searching for more domains where such techniques are applicable. It turns out that there is a single mathematical theory which can help us understand these questions in a single framework and give us answers to these seemingly non-related topics.

The mathematical framework which ties these questions together is called *abstract algebra*. Not surprisingly, given the name, the course is going to be about *abstract algebra*.

Exercise 1.1. What does *abstract* mean?

Note 1.2. The exercises given in the course notes are practice problems with the exception of this particular introduction. The exercises given in this particular document are to motivate the study of abstract algebra. You should try to think about them but remember that there are no clear answers.

We will precisely study the mathematical structures which can represent numbers, matrices, permutations, geometric objects under different parameters. The first step would be to define these mathematical (algebraic) structures like groups, rings and fields. The next step is to find properties of these algebraic structures. Finally we will also see how these properties give so many beautiful results in different areas of mathematics.

Lets start with a more basic question,

Exercise 1.3. What does *algebra* mean?

1.1.1 Arithmetic and algebra

Most of the people when asked the above question, think about numbers, equations and operations between them. So lets make the previous question more precise. What is the difference between arithmetic and algebra? Arithmetic is the study of numbers and the operations (like addition, subtraction, multiplication) between them. Algebra, intuitively, talks about equations, variables, symbols and relations between them.

The primary difference is the use of variables, which can stand for an unknown or a group of numbers. These variables are somewhat abstract but really help us in manipulating equations and solving them. It would be too cumbersome to write things in words instead of using equations and variables.

Exercise 1.4. Give an example where using a variable helps you to write a statement concisely.

Now we know what algebra is, lets talk about *abstract* part of it.

1.1.2 Abstraction

All of us like numbers (or at least understand the importance of it). One of the reason is that numbers are very well-behaved. In other words, there are so many nice properties that it is easy to manipulate and work with numbers. Lets look at one of the most fundamental properties,

Theorem 1.5. *Fundamental theorem of arithmetic: Every integer greater than* 1 *can be uniquely expressed as the product of primes up to different orderings.*

Since this property is so useful, we should ask, are there other objects which satisfy similar theorems.

Exercise 1.6. Do we have unique factorization theorem for matrices or permutations.

There is a very important methodology to generalize given proofs. You look at the proof and figure out the crucial step and properties which make the proof work. So one way to approach this question would be, carefully look at the proof of the theorem and figure out the properties of integers we have used at different step. Then check if another mathematical object satisfies the same properties.

In other words, *any* mathematical object which satisfies these properties will also have a unique factorization theorem. The abstract object which has all these properties can be given an appropriate name. This is similar to variables. As variables can take different values, this abstract object can be assigned different mathematical objects.

We will turn this method upside down. We will consider some basic properties and give a name to the abstract structure which satisfies these "basic properties".

Exercise 1.7. Who decides these basic properties?

Using these "basic properties" we will come up with multiple theorems like the unique factorization theorem above. By the above discussion any mathematical object (from arithmetic, algebra, geometry or anywhere else) which has these "basic properties" will satisfy all the theorems too. Hence in one shot we will get theorems in diverse areas.

You are already familiar with one such abstract structure, *set*. A collection of objects is called set and it needs no other property to be satisfied.

Exercise 1.8. What kind of theorems can you prove for sets?

In the course we will look at the collection of objects (sets) with certain composition properties. These will give rise to groups, rings etc.. The first such abstraction we will study is group.

Exercise 1.9. Should we choose as many basic properties as possible or as less basic properties as possible?

2

Groups

These notes are about the first abstract mathematical structure we are going to study, *groups*. You are already familiar with *set*, which is just a collection of objects. Most of the sets we encounter in mathematics are useful because of the operations we can perform on them. We can do addition, multiplication, AND, OR, take power etc..

Sets, by definition, need not have such operations. For example, $S = \{Apple, Oranges, CS203, Monitor\}$ is a set. But, if we look at more interesting sets like integers, matrices, permutations etc., we generally have operations which can be done on them. For example, you can add matrices, multiply permutations, add and multiply integers and so on.

Our next task is to define an abstract object (say a special set) with operation to compose elements inside the object. But first lets ask a basic question. What are the nice properties of addition of two natural numbers? What about integers?

To begin with, it is great that we can add two numbers, that is, the addition of any two numbers is a number. Another property not present in natural numbers is that we can always solve $a + x = b$ (a, b are given, x is unknown). Notice that we have to assume the existence of *Zero*.

Exercise 2.1. Can you think about other properties? Do they follow from the properties mentioned above?

2.1 Groups

A group G is a set with binary operation $*$, s.t.,

1. Closure: For any two elements $a, b \in G$; their composition under the binary operation $a * b \in G$.
2. Associativity: For all $a, b, c \in G$, we have $a * (b * c) = (a * b) * c$. This property basically means that any bracketing of $a_1 * a_2 * \cdots * a_k$ is same (exercise).
3. Identity: There is an element *identity* (e) in G, s.t., $a * e = e * a = a$ for all $a \in G$.
4. Inverse: For all $a \in G$, there exist $a^{-1} \in G$, s.t., $a * a^{-1} = a^{-1} * a = e$.

Note 2.2. Some texts define binary operation as something which has *closure* property. In that case, the first property is redundant. For the sake of brevity, it is sometimes easier to write xy instead of $x * y$.

Sometime we denote a group by its set and the operation, e.g., $(\mathbb{Z}, +)$ is the group of integers under addition.

Exercise 2.3. Show that integers form a group under addition (In other words, Integers have a group structure with respect to addition). Do they form a group under multiplication?

You can think of groups as being inspired by integers. In other words, we wanted to abstract out some of the fundamental properties of integers. We will later see that all groups share some properties with integers, but more interestingly, there are a lot of other groups which do not look like integers. That means there are some properties of integers which are not captured by the definition of groups. So what properties of integers do you think is not captured by groups?

To start with, we haven't specified *commutativity* as one of the basic properties. The properties are chosen so that we have many examples of groups and simultaneously we can prove a lot of theorems (properties) of this group structure. Later we will see that some important groups do not have commutativity property.

Definition 2.4. *A group is called* commutative *or* abelian *if,* $\forall a, b \in G; a * b = b * a$.

2.1.1 Examples of groups

Exercise 2.5. Can you think of any other group except integers under addition? Is it commutative?

The whole exercise of abstraction will be a waste if integers (addition) is the only set which follow group property. Indeed, there are many examples of groups around you, or at least in the mathematics books around you.

- Integers, Rationals, Reals, Complex numbers under addition. Clearly for all these 0 is the identity element. The inverse of an element is the negative of that element.
- Rationals, Reals, Complex numbers (without zero) under multiplication. Identity for these groups is the element 1. Why did we exclude integers?
- Positive rationals, positive reals under multiplication.
- The group $\mathbb{Z}_n$, set of all remainders modulo n under addition modulo n. Will it be a group under multiplication? How can you make it a group under multiplication?

Till now all the examples taken are from numbers. They are all subsets of complex numbers. Lets look at a few diverse ones.

- The symmetries of a regular polygon under composition. In other words, the operations which keep the polygon fixed. The symmetries are either obtained through rotation or reflection or combination of both. This group is called *Dihedral group.*
- The set of all permutations of $\{1, 2, \cdots, n\}$ under composition. What is the inverse element?
- The set of all $n \times n$ matrices under addition. The identity in this case is the all 0 matrix,

$$\begin{pmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{pmatrix}$$

- The set of all $n \times n$ invertible matrices of real numbers. What is the identity element?

We have seen so many examples of groups. Are they all *similar* (we will define the word *similar* later). Can we represent a group in a succinct way. One of the trivial representation is the *multiplication table* of the group. It is a matrix with rows and columns both indexed by group elements. The $(i, j)^{th}$ entry denotes the sum of i^{th} and j^{th} group element. For example, lets look at the multiplication table of $\mathbb{Z}_5^+$ under multiplication. Here $\mathbb{Z}_5^+$ denotes all the remainders modulo 5 Co-prime to 5 (gcd with 5 is 1).

	1	2	3	4
1	1	2	3	4
2	2	4	1	3
3	3	1	4	2
4	4	3	2	1

Exercise 2.6. Notice that every element occurs exactly once in every row and every column. Do you think this property is true for any group or just $\mathbb{Z}_5$?

Multiplication table gives us all the information about the group but is a pretty long description. Specifically it is quadratic in the size of the group. It turns out that groups have lot of properties which can help us in giving a more succinct representation. We already showed one property, that the identity is unique. What other theorems can be shown for groups?

2.2 Properties of groups

To start with, we need to define few quantities. Suppose we are given an element $x \neq e$ of group G. What other elements can be constructed with x. The composition with identity will not give anything new, so lets compose it with itself. Since G is a group, $x^2 := x * x, x^3 := x^2 * x$ (notice the new notation) and so on will be elements of group G. In this way we can create new elements in G except if these elements start repeating.

Suppose G is finite, then sooner or later there will exist i and j, s.t., $x^i = x^j$.

Exercise 2.7. Show that the first element which will repeat is e.

The least positive j for which $x^j = e$ is called the *order* of x and is denoted by $|x|$. Clearly the only element with order 1 is e and everything else will have a bigger order.

We will now go on to prove more properties of groups, but before that there is a warning. Groups are inspired by numbers and the notations are very similar. It is not surprising that sometimes you can get carried away and use properties of integers which are not really true for groups (e.g., commutativity).

For all the proofs for the theorems given below, notice that we will use the already known properties like closure, associativity, inverse, existence of identity. Then using those theorems we can prove other results. Now check your proofs for the exercises given in this section above.

This distinction can be made more clear by an analogy which we will use later too. Working with groups is like playing *football*. In general, for any activity you use your hands, feet or any other tool. But in case of football there is a restriction that you only use your feet. Using your feet you develop other skills which can be used to score a goal.

Our goal would be to prove theorems. Our feet will be the defining properties of groups (closure, associativity, inverse, identity). And the intermediate theorems would be like dribbling or kicking. You should not foul (use properties of integers) to prove a theorem (score a goal). So lets play football. We will use G to denote a group.

- The inverse of an element is unique.
  Proof: Suppose a has two inverses b and c. Then $c = (ba)c = b(ac) = b$. What properties of groups did we use in this proof.
- Cancellation laws: Given $a, b, x \in G$, we know $ax = bx \Rightarrow a = b$, and also $xa = xb \Rightarrow a = b$. These are called respectively the right and the left cancellation law.

  Exercise 2.8. Prove the assertion. What does it say about the rows (or columns) of multiplication table?

- $x \in G$ and x^{-1} have the same order.
  Proof: We will show that order of x^{-1} is at most the order of x, by symmetry this will prove the assertion. Suppose $x^n = e$. Multiply this equality by x^{-n} and we get $x^{-n} = e$ and hence the order of x^{-1} is less than n.

  Exercise 2.9. We did not define x^{-n}. What do you think it should be?

For a finite group we have shown that its order is less than the cardinality (also called the order) of the group. Actually order of an element can be restricted to just the divisors of the order of the group. Look carefully at the following theorem and proof.

Theorem 2.10. *Suppose G is a finite group with n elements (n is the order of the group). If d is the order of an element $x \in G$ then n is a multiple of d ($d \mid n$).*

Proof. We will prove the theorem in two steps. First, we will show that $x^n = e \;\; \forall x \in G$. Second, if there is any m, s.t., $x^m = e$ then d divides m. From these two steps the conclusion can be easily inferred.

From the cancellation laws, it is clear that $S_x = \{xg : g \in G\} = G$ as a set. All elements of S_x are distinct, in G and hence they are just a permutation of elements of G. Taking the product over all elements of S_x,

$$\Pi_{s \in S_x} s = \Pi_{g \in G} xg = x^n \Pi_{g \in G} g = x^n \Pi_{s \in S_x} s.$$

Using the first and the last step,

$$e = x^n.$$

So for every element $x \in G$, we know $x^n = e$.

For the second part, suppose $m = kd + r$ by division. Here k is the quotient and $r < d$ is the remainder. Then looking at x^m,

$$e = x^m = x^{kd+r} = x^r.$$

So there exist $r < n$, s.t. $x^r = e$. By the definition of order, $r = 0$. Hence d divides m.

Actually the proof given above is not correct.

Exercise 2.11. Where is the mistake in the proof? Hint: It is in the first part.

□

If you look at the proof of fact that $x^n = e$, then it was proved using commutativity. So we have only proved that for a *commutative* or *abelian* group the thm. 2.10 is true. It turns out that it is true for non-commutative groups too. We will prove the full generalization later with a different technique.

2.3 Isomorphism and homomorphism of a group

As discusses above we want to find out what kind of groups are there. Are they all *similar*. Let us formalize the notion of similarity now. Clearly if two sets are equal if and only if there is a bijection between them. But the bijection need not respect the composition. That means the composition properties of two groups might be completely different even if they have a bijection between them.

Exercise 2.12. Would you say that groups $(\mathbb{Z}_4, +)$ and $(\mathbb{Z}_8^+, \times)$ similar (both have four elements). The second group is the set of all remainders modulo 8 which are Co-prime to 8.

Hint: Look at the orders of different elements in these groups.

Hence for group similarity, we need to take care of composition too. Two groups are considered same if they are *isomorphic* to each other. In other words there exist an *isomorphism* between the two. To define, a group G_1 is isomorphic to group G_2, if there exist a bijection $\phi : G_1 \to G_2$, s.t.,

$$\forall g, h \in G_1 : \quad \phi(g)\phi(h) = \phi(gh).$$

The second property takes care of the composition. A related notion is called *homomorphism* where we drop the bijection criteria. So G_1 is homomorphic to G_2 if there exist a *map* $\phi : G_1 \to G_2$, s.t.,

$$\forall g, h \in G_1 : \quad \phi(g)\phi(h) = \phi(gh).$$

Exercise 2.13. Give a homomorphism which is not an isomorphism from a group G to itself.

2.4 Assignment

Exercise 2.14. For any $a_1, a_2, \cdots, a_k \in G$, show that expression $a_1 * a_2 * \cdots * a_k$ is independent of bracketing.

Hint: Show it using induction that all expression are same as $a_1 * (a_2 * (\cdots * a_k) \cdots)$.

Exercise 2.15. Prove that the identity is unique for a group.

Exercise 2.16. Which Groups are commutative from the list of groups given in the section 2.1.1?

Exercise 2.17. Prove that $G = \{a + b\sqrt{2} | a, b \in \mathbb{Q}\}$ is a group under addition.

Exercise 2.18. Which of them are groups under addition?

- The set of all rational numbers with absolute value < 1.
- The set of all rational number with absolute value ≥ 1.
- The set of all rational numbers with denominator either 1 or 2 in the reduced form.

Exercise 2.19. Find the order of following,

- 3 in $\mathbb{Z}_5, +$.
- 5 in $\mathbb{Z}_7, \times$.
- Transpositions in permutations. What about product of disjoint transpositions?

Exercise 2.20. Give an example of a finite group where order of an element is different from order of the group.

Exercise 2.21. If all elements have order 2 for a group G, prove that it is abelian.

Exercise 2.22. Show that if G_1 is isomorphic to G_2 then G_2 is isomorphic to G_1.

Exercise 2.23. Show an isomorphism from real numbers with addition to positive real numbers with multiplication.

3

Subgroups

We are interested in studying the properties and structure of the group. By properties, we mean the theorems which can be proven about groups in general. Then any mathematical construct having the group structure (satisfy closure, associativity etc.) will satisfy those theorems.

Another important task is to understand the structure of group itself. It is deeply related to the properties of group. It ultimately helps us in figuring out which groups are similar (with respect to isomorphism) and can we list out all possible kind of groups (not isomorphic to each other).

One of the natural question is that if groups can exist inside a group.

Exercise 3.1. Can we have a subset of group which itself is a group under the group operation? Try to construct such a set in $\mathbb{Z}$.

3.1 Definition

As the intuition would suggest,

Definition 3.2. *A subset H of a group G is called a subgroup if it is not empty, closed under group operation and has inverses. The notation $H \leq G$ denotes that H is a subgroup of G.*

Note 3.3. The subgroup has the same operation as the original group itself

Exercise 3.4. Why did we not consider associativity, existence of inverse?

Every group G has two trivial subgroups, e and the group G itself. Lets look at few examples of non-trivial subgroups. Try to prove that each of them is a subgroup.

- $n\mathbb{Z}$, the set of all multiples of n is a subgroup of Integers.
- Under addition, integers ($\mathbb{Z}$) are a subgroup of Rationals ($\mathbb{Q}$) which are a subgroup of Reals ($\mathbb{R}$). Reals are a subgroup of Complex numbers, $\mathbb{C}$.
- $\mathbb{Z}^+$, the set of all positive integers is not a subgroup of $\mathbb{Z}$. Why?
- The set $S = \{a + b\sqrt{2} : a, b \in \mathbb{Z}\}$ is a subgroup of $\mathbb{R}$ under addition.
- Center of a group: The *center* of a group G is the set of elements which commute with every element of G.
$$C(G) = \{h \in G : \; hg = gh \;\; \forall g \in G\}.$$
We will show that center is the subgroup. Associativity follows from G and existence of identity is clear. Suppose $h, k \in C(G)$, then for any $g \in G$,
$$g(hk) = hgk = (hk)g.$$
Hence $C(G)$ is closed. For the inverse, note that $gh = hg$ is equivalent to $h^{-1}gh = g$ and $g = hgh^{-1}$. Hence existence of inverse follows (Why?).

3.1.1 Cyclic groups

We noticed that $\{e\}$ is a subgroup of every group. Lets try to construct more subgroups. Suppose x is some element which is not the identity of the group G. If k is the order of x then $S_x = \{e, x, x^2, \cdots, x^{k-1}\}$ is a set with all distinct entries. It is clear from previous discussion of groups that S_x is a subgroup.

Exercise 3.5. Prove that S_x is a subgroup.

While proving the previous exercise, we need to use the fact that k is finite. What happens when k is infinite? Can we construct a group then? The answer is yes, if we include the inverses too. All these kind of groups, generated from a single element, are called *cyclic groups.*

Definition 3.6. *A group is called* cyclic *if it can be* generated *by a single element. In other words, there exist an element $x \in G$, s.t., all elements of G come from the set,*

$$< x >= \{\cdots, x^{-2}, x^{-1}, e, x, x^2, \cdots\}$$

There are many things to note here:

- For an infinite group, we need to consider inverses explicitly. For a finite group, inverses occur in the positive powers.
- The group *generated* by the set S is the group containing all possible elements obtained from S through composition (assuming associativity, inverses etc.).
- The notation for the group generated by S is $< S >$.

The structure of cyclic groups seem very simple. You take an element and keep composing. What different kind of cyclic groups can be there? Look at different examples of cyclic groups of order 4 in figure 3.1.1. The next theorem shows that all these are isomorphic.

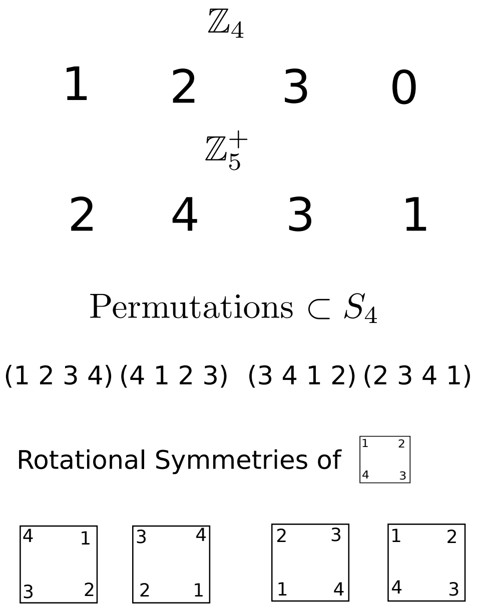


Fig. 3.1. Different cyclic groups

Theorem 3.7. *Every finite cyclic group G of order n is isomorphic to $\mathbb{Z}_n$.*

Proof. Suppose x is a generator for G. It exists by the definition of G. Then since the order is finite, group G is,

$$G = \{e, x, x^2, \cdots, x^{n-1}\}$$

Lets look at the obvious bijection ϕ from $\mathbb{Z}_n$ to G. The element k is mapped to x^k. It is a bijection because, the inverse maps x^k to k. For the above bijection,

$$\phi(j+k) = x^{j+k \mod n} = x^j * x^k = \phi(j) * \phi(k).$$

Where first inequality follows from the definition of $\mathbb{Z}_n$ and second from the fact that $x^n = 1$. This shows that ϕ is an isomorphism. Hence Proved. □

Using the previous theorem and exercise (assignment), we have given complete characterization of cyclic groups. This loosely means that we can get all the properties of any cyclic group of order n from $\mathbb{Z}_n$ and an infinite cyclic group with integers.

This is called a *classification* of cyclic groups. We would ideally like to give classification of groups and finding out more properties of groups. These two questions are not independent. We will explore both simultaneously and progress in one question helps in finding the answer for other.

Exercise 3.8. What are the subgroups of a cyclic group?

3.2 Cosets

The next step in understanding the structure of a group is to partition it using a subgroup. Suppose we are given a group G and its subgroup H. We will show that G can be partitioned into disjoint sets of equal size ($|H|$). This will imply that $|G|$ is always divisible by $|H|$. Lets define these parts first and then we can prove the fact given above.

Definition 3.9. *Cosets: The* left coset *(gH) of H with respect to an element g in G is the set of all elements which can be obtained by multiplying g with an element of H,*

$$gH = \{gh : \quad h \in H\}.$$

This is called the left coset because g is multiplied on the left. We can similarly define the right cosets Hg.

Exercise 3.10. How are left and right coset related for commutative groups?

Let us show some properties of these cosets. Remember not to use any illegal property while proving these. Without loss of generality we will assume that cosets are left. Same properties hold true for right ones too.

- Every element of G is in at least one coset. H is one of the cosets too.

  Proof. Exercise. □

- The cardinality of all cosets is equal and hence their cardinality is $|H|$.

  Proof. Consider a coset gH and a subgroup $H = \{h_1, h_2, \cdots, h_k\}$. The elements of the left coset gH are $\{gh_1, gh_2, \cdots, gh_k\}$. It is easy to show that any two elements in this set are distinct (why?). Hence all cosets have cardinality $k = |H|$. □

- For any two elements g_1, g_2 of G either g_1H, g_2H are completely distinct (disjoint) or completely same ($g_1H = g_2H$).

Proof. Suppose there is one element common in g_1H and g_2H (otherwise they are completely distinct). Say it is $g_1h_1 = g_2h_2$, then,

$$g_1 = g_2h_2h_1^{-1} \to \exists\ h \in H:\ g_1 = g_2h.$$

Now you can prove a simple exercise.

Exercise 3.11. If $\exists\ h \in H:\ g_1 = g_2h$ then show that $g_1H \subseteq g_2H$.

But if $g_1 = g_2h$ then $g_2 = g_1h^{-1}$. This will show from the previous exercise that $g_2H \subseteq g_1H$. Hence both the sets g_1H and g_2H are the same. □

Using the properties we have shown that the two columns of the following table are completely the same or completely distinct.

G/H	e	g_2	$\cdots$	g_n
e	e	g_2	$\cdots$	g_n
h_2	h_2	g_2h_2	$\cdots$	g_nh_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
h_k	h_k	g_2h_k	$\cdots$	g_nh_k

This conclusion is beautifully summarized in Lagrange's theorem.

3.2.1 Lagrange's theorem

Using the previous list of properties it is clear that if we look at the distinct cosets of H then they partition the group G into disjoint parts of equal size.

Exercise 3.12. What is the size of these parts?

Theorem 3.13. *Lagrange: Given a group G and a subgroup H of this group, the order of H divided the order of G.*

Proof. The proof is left as an exercise. You should try to do it without looking at the hint given in the next line.

Hint: From the previous discussion, the $\frac{|G|}{|H|}$ is just the number of distinct cosets of H. □

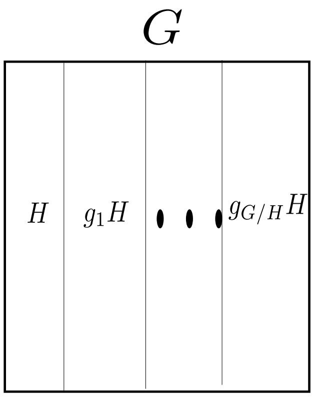


Fig. 3.2. Coset decomposition

Note 3.14. If the set of left and right cosets coincide the subgroup is called *normal.* In this case, the set of cosets actually forms a group, called the *quotient group* $\frac{G}{H}$ (What is the composition rule?).

This is a great discovery. The statement of Lagrange's theorem does not do justice to the implications. We started with an abstract structure with some basic properties like associativity, inverses etc. (group). The proof of Lagrange's theorem implies that if we can find a subgroup of the group then the whole group can be seen as a disjoint partition with all parts related to the subgroup. Notice that it is easy to construct a cyclic subgroup of a group.

Exercise 3.15. Prove that the order of an element always divides the order of a group. We had proved this for commutative groups in an earlier lecture.

Exercise 3.16. What does Lagrange's theorem say about groups with prime order?

Lets look at one application of Lagrange's theorem in the case of $\mathbb{Z}_m^\times$. We know that this group contains all the remainders mod m which are coprime (gcd is 1) to m. If m is a prime p then $\mathbb{Z}_p^\times$ contains $p-1$ elements. This proves the well known *Fermat's little theorem.*

Exercise 3.17. Fermat's little theorem: For a prime p and any number a,

$$a^{p-1} = 1 \mod p.$$

Prove this theorem.

3.3 Dihedral group

Till now most of the exercises we have done are for $\mathbb{Z}$ and $\mathbb{Z}_n$. These groups are commutative. This section will introduce you to a non-commutative subgroup.

Definition 3.18. *A Dihedral group D_{2n} is the group of symmetries of a regular n-gon.*

A regular n-gon can be rotated or reflected to get back the n-gon. The group D_{2n} is the group generated by reflection s and rotation r by the angle $\frac{2\pi}{n}$. Refer to figure 3.3 for all the symmetries of a pentagon.

For an $n - gon$ there are n rotations possible. The set of rotations form a cyclic group of order n.

Exercise 3.19. What is the inverse of rotation r. Convince yourself that set of rotations form a cyclic group.

On the other hand reflection is the inverse of itself. Hence it is an element of order 2. From the figure you can guess that there will be $2n$ symmetries of the form $s^i r^j$, where i ranges in $\{0,1\}$ and j is an element from $\{0, 1, \cdots, n-1\}$. Using this notation, rs means we apply s first and then r.

Exercise 3.20. Convince yourself that $rs \neq sr$.

Notice that we have given a description of $2n$ elements of the dihedral group D_{2n}. How can we be sure that there are no more elements generated by r and s. What about $rsrs$?

Exercise 3.21. Show that $rs = sr^{-1}$.

This relation tell us how to interchange r and s in any expression involving both. This way we can convert any element of the group generated by r and s to be of the form $s^i r^j$ with i and j ranging appropriately.

The above discussion shows the important properties (defining properties) of dihedral group.

- An element of order 2, s.
- An element of order n, r.
- The commutation relation $rs = sr^{-1}$.
- $s \neq r^i$ for any i.

Any group which is generated by two elements with the above mentioned properties will be isomorphic to D_{2n}.

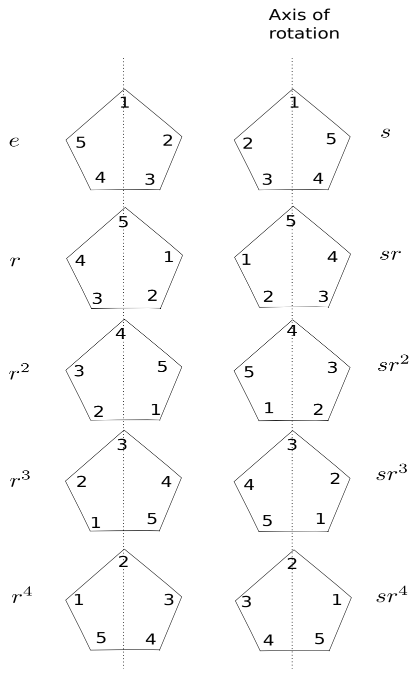


Fig. 3.3. Symmetries of a pentagon

3.4 Assignment

Exercise 3.22. List all possible subgroups of $\mathbb{Z}_6$ under addition.

Exercise 3.23. The *kernel* of a homomorphism $\phi : G \to L$ is the subset of G which maps to identity of L. Hence,

$$Ker(\phi) = \{g \in G : \ \phi(g) = e_L\}.$$

Similarly, the *image* of ϕ are the elements of L which have some element mapped to them through ϕ.

$$Img(G) = \{h \in L : \ \exists g \in G \text{ for which } \phi(g) = h.\}$$

show that $Img(G)$ and $Ker(G)$ are subgroups.

Exercise 3.24. Show that a subset H is a subgroup of G if it is non-empty and $\forall x, y \in H : \ xy^{-1} \in H$.

Note 3.25. Because H is a subset, the set of properties we need to check are much less.

Exercise 3.26. Show that $\mathbb{Z}_n$ is cyclic under addition. Give some examples of cyclic subgroups and some examples of non-cyclic subgroups in $\mathbb{Z}_n^+$ under multiplication.

Exercise 3.27. Show that all cyclic groups are commutative (abelian).

Exercise 3.28. Show that every cyclic group with infinite order (having infinite elements) is isomorphic to $\mathbb{Z}$ under addition.

Hint: Look for the obvious bijection between the group and $\mathbb{Z}$. Show that it is an isomorphism.

Exercise 3.29. Find the order of every element of group $\mathbb{Z}_p$ where p is a prime.

Exercise 3.30. Find the left cosets of $3\mathbb{Z}$ in group $\mathbb{Z}$.

Exercise 3.31. If order of a group G is prime p then show that it is isomorphic to $\mathbb{Z}_p$.

Exercise 3.32. Euler's theorem: For a number m, say $\phi(m)$ is the number of positive elements coprime to m and less than m. For any a which is co-prime to m,

$$a^{\phi(m)} = 1 \mod m.$$

Prove this theorem.

Exercise 3.33. Show that there always exist a cyclic subgroup of any finite group G.

Exercise 3.34. Show that the subgroup of a cyclic group is cyclic.

4

Orbits

In the beginning of the course we asked a question. How many different necklaces can we form using 2 black beads and 10 white beads? In the question, the numbers 2 and 10 are arbitrarily chosen. To answer this question in a meaningful way, we need to construct a strategy or theorem which will answer the above question for any such numbers. But to understand the question better, lets ask a simpler question.

Exercise 4.1. How many different necklaces can be formed using 2 white and 2 black beads?

Lets look at the question in detail. The first guess for the question would be $4! = 24$, the number of ways we can permute the four beads. But all these permutations need not be different. What do we mean by *different* necklaces? It might happen that two different permutations (σ_1 and σ_2) might be the same in the sense that σ_1 can be obtained from σ_2 using rotation. Look at figure 4 for one such example.

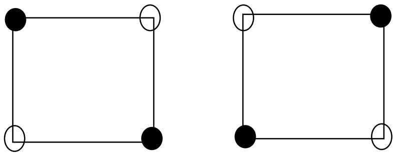


Fig. 4.1. Two permutations giving the same necklace

Using some brute force now, we can come up with all possible different necklaces for 2 white and 2 black beads (figure 4).

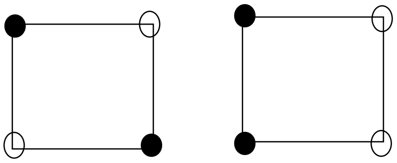


Fig. 4.2. Possible different necklaces with 2 white and 2 black beads

You can convince yourself that the question is much harder if we take bigger numbers. What should we do? When are two necklaces equivalent?

Two necklaces are equivalent if we can obtain one by applying a symmetry to other necklace (like rotation or reflection). We know from discussions in previous classes that the symmetries form the dihedral group, D_{2n}. The strategy would be to develop a general framework for groups to answer question about distinct necklaces.

4.1 Group action

The first thing to notice in the necklace problem is that there are two different objects of interest. One is the set of necklaces (set of all permutations of the necklaces) and other is the set of symmetries. A symmetry can be applied to a necklace to obtain another necklace. Lets make this action abstract.

Abstractly, given a group G and a set A, every element g of G *acts* on set A. That means for every element g there is a function from A to A which is called its action on G. For the sake of brevity, we will denote the function corresponding to the element $g \in G$ with g itself. Hence the value of $a \in A$ after action of g will be called $g(a)$.

Exercise 4.2. What is the group and what is the set for the necklace problem?

Note 4.3. It is NOT the set of distinct necklaces.

Lets look at the formal definition.

Definition 4.4. *Given a group G and a set A, a group action from G to A assigns a function $g : A \to A$ for every element g of group G. A valid group action satisfies the following properties.*

- *Identity takes any element $a \in A$ to a itself, i.e., $e(a) = a$ for every $a \in A$.*
- *For any two group elements $g_1, g_2 \in G$, their functions are consistent with the group composition,*

$$g_1(g_2(a)) = (g_1 g_2)(a).$$

Using this definition and group structure of G, it can be shown that action of g is a permutation on the elements of A.

This gives us another representation of group elements. For any group action on A of size m, we have a permutation representation for any element $g \in G$ in terms of a permutation on m elements. In the following sections we will keep this representation in mind.

Note 4.5. Actually a slightly stronger theorem holds. It is called *Cayley's theorem* and is given below. We will not show the proof of this theorem.

Theorem 4.6. *Cayley's theorem: Every group of order n is isomorphic to some subgroup of S_n.*

4.2 Orbits

Suppose we are given action of group G on a set A. Lets define a relation between the elements of A. If $\exists\ g :\ g(x) = y$ then we will say that x, y are related ($x \sim y$). We can easily prove that this relation is equivalence relation.

- Reflexive: Why?
- Symmetric: Suppose $x \sim y$ because $g(x) = y$. Then consider $x = ex = (g^{-1}g)(x) = g^{-1}y$, implying $y \sim x$.
- Transitive: Show it as an exercise.

Hence this equivalence relation will partition the set A into distinct equivalence classes. The equivalence class corresponding to $x \in A$ is the orbit ($G(x)$) of element x. In other words,

$$G(x) = \{g(x) :\ g \in G\}.$$

Now we will look at two counting questions,

1. What is the size of these orbits?
2. How many distinct orbits are there?

Why are we interested in these questions. Let us look at this concept from the example of necklaces. If a necklace x can be obtained from another necklace y using a symmetry then they are related (in the necklace case indistinguishable).

Exercise 4.7. Convince yourself that the number of distinct necklaces is the same as the number of distinct orbits (equivalence classes) under the dihedral group D_{2n}.

We will answer both the counting questions under the general group-theoretic framework. As a special case, this will solve the necklace problem.

4.2.1 stabilizers

Remember that the orbit of $x \in A$ under the action of G can be defined as,

$$G(x) = \{g(x) : \; g \in G\}.$$

If every $g \in G$ took x to a different element, the size of the orbit would be $|G|$. But this is too much to expect. If we consider any example, there will be lots of $g \in G$ which will take x to a single element y. Lets define this set as G(x,y),

$$G(x, y) = \{g \in G : \; g(x) = y\}.$$

Exercise 4.8. Does the set $G(x, y)$ form a subgroup of G? Under what condition will it form a subgroup?

The answer to the previous exercise is when $x = y$. The set $G_x := G(x, x)$ is called the stabilizer of x,

$$G_x = \{g \in G : \; g(x) = x\}.$$

Exercise 4.9. If you were not able to solve the previous exercise, prove that G_x is a subgroup of G.

Once we have the subgroup G_x, the natural question to ask is, what are the cosets? This is where we get lucky. Suppose y is an element of the orbit $G(x)$. So there exist an $h \in G$, s.t., $h(x) = y$. Then $G(x, y)$ is precisely the coset hG_x.

Lemma 4.10. *Given a $y \in A$, s.t., $h(x) = y$. The coset hG_x is same as the set $G(x, y)$.*

Proof. $\Rightarrow$: An element of hG_x is of the form $hg, \; g \in G_x$. Then $hg(x) = h(x) = y$. So $hG_x \subseteq G(x, y)$.

$\Leftarrow$: Suppose $g \in G(x, y)$, i.e., $g(x) = y$. Then show that,

Exercise 4.11. $h^{-1}g \in G_x$

But $h^{-1}g \in G_x$ implies $g \in hG_x$.

Exercise 4.12. Show the above implication. Be careful, It is not just the same as multiplying by h on both sides.

From the previous exercise $G(x, y) \subseteq G_x$. □

Hence, for every element y in the orbit $G(x)$, there is a coset. It is an easy exercise to convince yourself that every coset will correspond to a single element in the orbit $G(x)$. So the number of elements in the orbit is equal to the number of cosets. But we know that the number of cosets can be calculated from Lagrange's theorem. Hence,

$$|G| = |G_x||G(x)|.$$

Note 4.13. G_x is a subset of G, but $G(x) \subseteq A$. The equation works because we show a one to one relation between $G(x)$ (orbit) and the cosets.

4.2.2 Burnside's lemma

We now know the size of the orbit. Given an element x with stabilizer G_x, the number of elements in its orbit is $\frac{|G|}{|G_x|}$. Can this help us in counting the number of distinct orbits.

Lets give every element on A a weight of $\frac{1}{|G(x)|}$. The number of distinct orbits is now the sum of weights of all elements of A.

$$\text{Number of distinct orbits } = \sum_{x \in A} \frac{1}{|G(x)|} = \frac{1}{|G|} \sum_{x \in A} |G_x|.$$

Lets concentrate on the summation. The total summation is equal to the number of pairs $(g \in G, x \in A)$, s.t., $g(x) = x$. Suppose we make a matrix with rows indexed by elements of G and columns indexed by elements of A. The entry (g, x) is one if $g(x) = x$ and 0 otherwise.

$$\begin{array}{r|cccc}
 & x_1 & x_2 & \cdots & x_{|A|} \\
\hline
g_1 & 0 & 1 & \cdots & 1 \\
g_2 & 0 & 0 & \cdots & 1 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
g_{|G|} & 1 & 0 & \cdots & 0
\end{array}$$

Then $\sum_x |G(x)|$ is the number of 1's in the matrix above. Each term in the summation, $|G_x|$ is the number of 1's in the column corresponding to x. We can count the number of 1's in the matrix by taking the sum row-wise too. Suppose, $S(g)$ is set of elements of A fixed by g.

$$S(g) = \{x : \; g(x) = x, \; x \in A\}.$$

Using $S(g)$ we get the orbit-counting (Burnside's) lemma.

Lemma 4.14. *Burnside's lemma (Orbit-counting): Given a group action of G over A. The number of distinct orbits can be written as,*

$$\textit{Number of distinct orbits } = \frac{1}{|G|} \sum_{g \in G} |S(g)|.$$

Note 4.15. The summation is now over G instead of A.

One natural question you might ask is, How did this benefit us? Previously we were summing over all possible $x \in A$ and now we are summing up over all $g \in G$. The reason is, in general, the size G will be much smaller than size of A.

Lets look at an example where Orbit-counting lemma will help us in answering the question about necklaces. Try to solve this exercise yourself first and later you can look at the solution given below.

Exercise 4.16. How many necklaces can be formed with 2 black and 6 white beads?

Arrange the beads on the vertices of a regular 8-gon. Since the necklaces are obtained by fixing the position of 2 black beads, there are 28 elements in A. The symmetry group is D_{16} with 16 elements.

For different elements of G we can calculate the number of elements fixed by it.

- Identity e: Fixes 28 elements.
- Out of 7 other rotations, only one of them fixes 4 elements. What is the angle of that rotation? Rest do not fix anything.
- All reflections fix exactly 4 elements. It is easy to see by looking at the cycle structure of the permutation. All beads of the same color should fall in the same cycle.

So by Orbit-counting lemma,

$$\text{Number of distinct orbits } = \frac{1}{16}(28 + 4 + 8 \times 4) = 4.$$

Exercise 4.17. What are the four configurations? Can you characterize them?

For other examples of application of Burnside's lemma, please look at section 21.4 of Norman Biggs book. There is a nice example in Peter Cameron's notes on Group Theory too (section 1.3).

4.3 Group representations (advanced)

We looked at the permutation representation of every group. There is a matrix representation of every group too. The study of that representation is called the group representation theory. Group representation theory is one of the main tools to understand the structure of a group. We will only give a very basic idea of this field. Interested students can look at book *Algebra* by Artin.

Define GL_n to be the group of invertible matrices of size $n \times n$ with complex entries. We can also think of these matrices as linear operators over $\mathbb{C}^n$.

A linear representation (matrix representation) is a homomorphism from group G to the group GL_n (say $R : G \to GL_n$). That means, we map every element of group G to an invertible matrix, s.t., it obeys the group composition,

$$R(gh) = R(g)R(h).$$

If there is a subspace of $\mathbb{C}^n$ which is fixed by every group element, then the representation is reducible. In other words, for an irreducible representation, there is NO subspace which is fixed by every element of group G.

Exercise 4.18. What does it mean that the subspace is fixed?

It can be shown that every representation can be broken down into irreducible representations. Another quantity of interest is the *character*. The character of the representation is a function $\chi : G \to \mathbb{C}$ defined by $\chi(g) = trace(R(g))$.

The characters of irreducible representations are orthonormal to each other and satisfy various other nice properties. Many theorems in group theory are derived by studying the characters of the group. Again, interested students can find more information in the book *Algebra* by Artin.

4.4 Assignment

Exercise 4.19. Show that action of dihedral group on the set of necklaces is a group action.

Exercise 4.20. Show that G is isomorphic to a subgroup of $S_{|A|}$. Remember that S_n is the group of permutations of $[n]$.

Hint: First show that action of g on A is a permutation and then use the consistency of group action with the group composition.

Exercise 4.21. Suppose we want to find the number of necklaces with m black and n white beads. What is the size of G and what is the size of A in terms of m, n?

Exercise 4.22. Prove that the average number of elements fixed by an element of group G under group action is an integer.

Exercise 4.23. Prove that GL_n is a group. What composition rule did you use?

Exercise 4.24. Biggs: Let G be a group of permutations of set A. If u, v are two elements in the same orbit of G, show that $|G_u| = |G_v|$.

Exercise 4.25. Biggs: Let A denote the set of corners of the cube and let G denote the group of permutations of A which correspond to rotation of the cube. Show that,

- G has just one orbit.
- For any corner x, $|G_x| = 3$.
- $|G| = 24$

Exercise 4.26. Biggs: Suppose you manufacture an identity card by punching two holes in an 3×3 grid. How many distinct cards can you produce. Look at the figure given below.

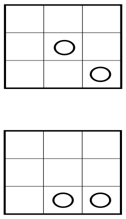


Fig. 4.3. Different identity cards with circles showing the holes.

Hint: The group to consider here is D_8.

5

Quotient group

We have seen that the cosets of a subgroup partition the entire group into disjoint parts. Every part has the same size and hence Lagrange's theorem follows. If you are not comfortable with cosets or Lagrange's theorem, please refer to earlier notes and refresh these concepts.

So we have information about the size of the cosets and the number of them. But we lack the understanding of their structure and relations between them. In this lecture, the concept of *normal subgroups* will be introduced and we will form a group of cosets themselves !!

5.1 Normal subgroup

Suppose we are given two elements g, n from a group G. The *conjugate* of n by g is the group element gng^{-1}.

Exercise 5.1. When is the conjugate of n equal to itself?

Clearly the conjugate of n by g is n itself iff n and g commute.

We can similarly define the conjugate of a set $N \subseteq G$ by g,

$$gNg^{-1} := \{gng^{-1} : \ n \in N\}.$$

Definition 5.2. *Normal subgroup: A subgroup N of G is* normal *if for every element g in G, the conjugate of N is N itself.*

$$gNg^{-1} = N \quad \forall g \in G.$$

We noticed that $gng^{-1} = n$ iff g, n commute with each other.

Exercise 5.3. When is $gNg^{-1} = N$?

In this case the left and right cosets are the same for any element g with respect to subgroup N. Hence, a subgroup is normal if its left and right cosets coincide.

Exercise 5.4. Show that following are equivalent. So you need to show that each of them applies any other.

1. N is a normal subgroup.
2. The set $S = \{g : gN = Ng\}$ is G itself.
3. For all elements $g \in G$, $gNg^{-1} \subseteq N$.

Hint: Instead of showing all $2 \times \binom{3}{2}$ implications, you can show $1) \Rightarrow 2) \Rightarrow 3) \Rightarrow 1$.

5.2 Quotient group

We have introduced the concept of normal subgroups without really emphasizing why it is defined. Lets move to our original question. What can be said about the set of cosets, do they form a group?

Suppose G is a group and H is a subgroup. Denote by S, the set of cosets of G with respect to H. For S to be a group it needs a law of composition. The most natural composition rule which comes to mind is,

$$(gH)(kH) = (gk)H.$$

Here gH and kH represent two different cosets. The problem with this definition is that it might not be *well-defined*. It might happen that $g' \in gH$ and $k' \in kH$ when multiplied give a totally different coset $(g'k')H$ then $(gk)H$.

Exercise 5.5. Show that this operation is well-defined for commutative (abelian) groups.

What about the general groups? Here comes the normal subgroup to the rescue.

Theorem 5.6. *Suppose G is a group and H is its subgroup, the operation,*

$$(gH)(kH) = (gk)H,$$

is well defined if and only if H is a normal subgroup.

Note 5.7. Every subgroup of a commutative group is normal.

Proof. $\Rightarrow$): We need to show that if the operation is well defined then $ghg^{-1} \in H$ for every $g \in G, h \in H$. Consider the multiplication of H with $g^{-1}H$. Since $e, h \in H$, we know $eH = hH$. Since the multiplication is well defined,

$$(eg^{-1})H = (eH)(g^{-1}H) = (hH)(g^{-1}H) = (hg^{-1})H \quad \Rightarrow \quad g^{-1}H = (hg^{-1})H.$$

Again using the fact that $e \in H$, $hg^{-1} \in g^{-1}H$. This implies $hg^{-1} = g^{-1}h' \quad \Rightarrow ghg^{-1} = h'$ for some $h' \in H$.

$\Leftarrow$): Suppose N is a normal subgroup. Given $g' = gn$ and $k' = kn'$, where $g, g', k, k' \in G$ and $n, n' \in N$, we need to show that $(gk)N = (g'k')N$.

$$(g'k')N = (gnkn')N.$$

Exercise 5.8. Show that there exist $m \in N$, s.t., $nk = km$. Hence complete the proof.

□

With this composition rule we can easily prove that the set of cosets form a group (exercise).

Definition 5.9. *Given a group G and a normal subgroup N, the group of cosets formed is known as the* quotient group *and is denoted by $\frac{G}{N}$.*

Using Lagrange's theorem,

Theorem 5.10. *Given a group G and a normal subgroup N,*

$$|G| = |N||\frac{G}{N}|$$

5.3 Relationship between quotient group and homomorphisms

Let us revisit the concept of homomorphisms between groups. The homomorphism between two groups G and H is a mapping $\phi : G \rightarrow H$ that preserves composition.

$$\phi(gg') = \phi(g)\phi(g')$$

For every homomorphism ϕ we can define two important sets.

- Image: The set of all elements h of H, s.t., there exists $g \in G$ for which $\phi(g) = h$.

$$Img(\phi) = \{h \in H : \ \ \exists g \in G \ \ \phi(g) = h\}$$

Generally, you can restrict your attention to $Img(\phi)$ instead of the entire H.

- Kernel: The set of all elements of G which are mapped to identity in H.

$$Ker(\phi) = \{g \in G : \ \ \phi(G) = e_H\}$$

Notice how we have used the subscript to differentiate between the identity of G and H.

Note 5.11. $Img(\phi)$ is a subset of H and $Ker(\phi)$ is a subset of G.

Exercise 5.12. Prove that $Img(\phi)$ and $Ker(phi)$ form a group under composition with respect to H and G respectively.

Exercise 5.13. Show that $Ker(\phi)$ is a normal subgroup.

There is a beautiful relation between the quotient groups and homomorphisms. We know that $Ker(\phi)$ is the set of elements of G which map to identity. What do the cosets of $Ker(\phi)$ represent. Lets take two elements g, h of a coset $gKer(\phi)$. Hence $h = gk$ where $\phi(k) = e_H$. Then by the composition rule of homomorphism $\phi(g) = \phi(h)$.

Exercise 5.14. Prove that $\phi(g) = \phi(h)$ if and only if g and h belong to the same coset with respect to $Ker(\phi)$.

The set of elements of G which map to the same element in H are called the fibers of ϕ. The previous exercise tell us that fibers are essentially the cosets with respect to $Ker(\phi)$ (the quotient group).

The fibers are mapped to some element in $Img(\phi)$ by ϕ. Hence there is a one to one relationship between the quotient group $\frac{G}{Ker(\phi)}$ and $Img(\phi)$. Actually the relation is much stronger.

It is an easy exercise to show that the mapping between quotient group $\frac{G}{Ker(\phi)}$ and $Img(\phi)$ is an isomorphism.

Exercise 5.15. Prove that the above mapping is an isomorphism.

Again applying the Lagrange's theorem,

$$|G| = |Ker(\phi)||Img(\phi)|.$$

The figure 5.11 depict that every element of quotient group is mapped to one element of image of ϕ. Now we know that this mapping is *well-behaved* with respect to composition too.

$$\phi((gKer(\phi))(hKer(\phi))) = \phi(gKer(\phi))\phi(hKer(\phi))$$

There is an abuse of notation which highlights the main point also. The notation $\phi(gKer(\phi))$ represents the value of ϕ on any element of $gKer(\phi)$. We know that they all give the same value. The study of homomorphism is basically the study of quotient group. The study of quotient group can be done by choosing a representative for every coset and doing the computation over it (instead of the cosets).

We have shown that $Ker(\phi)$ is normal. It can also be shown that any normal subgroup N is a kernel of some homomorphism ϕ (exercise).

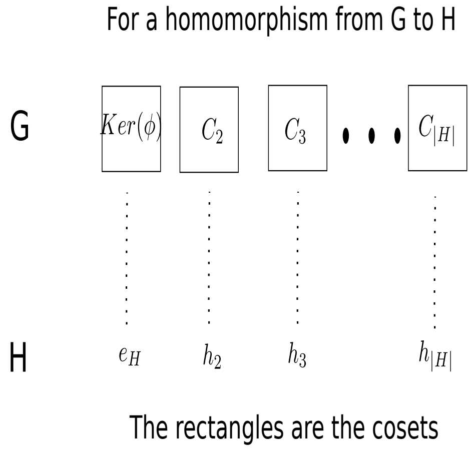


Fig. 5.1. Relationship between the quotient group and the image of homomorphism

5.4 Assignment

Exercise 5.16. Given a subgroup H of G, two elements $x, y \in G$ are related ($x \sim y$) if $x^{-1}y \in H$. Prove that this relation is an equivalence relation. What are the equivalence classes of this relation?

Exercise 5.17. Given a group G and a normal subgroup N. Say the set of cosets is called S and has composition operation $(gH)(kH) = (gk)H$. Show that,

- Identity exists in this set.
- Inverses exist in this set.
- Associativity is satisfied.

Since Closure is obvious we get that S is a group with respect to the above mentioned composition rule.

Exercise 5.18. Given a group G and a subgroup N as a set. Write a program to find if N is normal or not. Assume that you are given a function $mult(x, y)$, which can compute the binary operation of the group G between any two elements x, y of G.

Exercise 5.19. What is the quotient group of D_{2n} with respect to the subgroup generated by reflection?

Exercise 5.20. Suppose G is an abelian group and H is a subgroup. Show that $\frac{G}{H}$ is abelian.

Exercise 5.21. Given N is a normal subgroup, prove that $g^k(N) = (gN)^k$.

Exercise 5.22. Suppose N is normal in G, show that for a subgroup H, $H \cap N$ is a normal subgroup in H.

Exercise 5.23. Show that a subgroup N is normal in G iff it is the kernel of a homomorphism from G to some group H.

6

Rings

We have shown that $\mathbb{Z}_n$ is a group under addition and $\mathbb{Z}_n^+$ is a group under multiplication (set of all numbers co-prime to n in $\mathbb{Z}_n$). Till now, the two operations + and × have been treated differently. But from our experience with integers and even matrices, these operations satisfy properties like "distribution" ($a(b+c) = ab + ac$).

Hence, after success in defining an abstract structure with one operation (group), now we define another abstract structure with 2 operations. The first question is, what should be the defining properties of this new abstract structure. We will be inspired by integers again and define the concept of *Rings*.

6.1 Rings

Consider two operations + and × in a set R.

Definition 6.1. *The set R with the two operations + and × is a* ring*, if,*

- *R is a commutative group under +.*
- *R is associative, closed and has an identity with respect to the operation ×.*
- *The two operations + and × follow the distributive law, i.e.,*

$$a \times (b + c) = a \times b + a \times c \text{ and } (a + b) \times c = a \times c + b \times c.$$

Note 6.2. We will always assume that the multiplicative identity is different from additive identity. The additive identity will be denoted by 0 and multiplicative identity by 1. For brevity, we will denote $a \times b$ as ab.

Exercise 6.3. Are the two conditions under the distributive law same?

Exercise 6.4. Why did we assume commutativity under addition for a ring?

There are many examples of rings, many of these sets we have encountered before.

- The sets $\mathbb{Z}$,$\mathbb{Q}$,$\mathbb{R}$,$\mathbb{C}$ are rings with addition and multiplication.
- The set of integers modulo m, $\mathbb{Z}_m$, is a ring with addition and multiplication.
- The set of 2×2 matrices with integer entries is a ring. Actually if R is a ring then set of 2×2 matrices with entries in R is also a ring.

Another ring which will be of our particular interest is the ring of polynomials. The set $R[x]$ is the set of all polynomials with coefficients from ring R. If the multiplication in R is commutative then $R[x]$ is also a commutative ring.

Note 6.5. The addition and multiplication of polynomials is defined in the same way as in regular polynomials.

Exercise 6.6. Check that you can define these operations on polynomials with entries from a ring R. Why do we need that multiplication is commutative in the original ring?

Hence we have polynomial rings $\mathbb{Z}[x], \mathbb{Q}[x], \mathbb{R}[x], \mathbb{C}[x]$ having commutative multiplication.

6.1.1 Units of a ring

The ring is not a group with respect to multiplication. That is because inverses need not exist in a ring (e.g., integers). The elements of rings which have inverses inside the ring with respect to multiplication are called *units* or *invertible elements*.

The set of units for $\mathbb{Z}$ are just ± 1.

Exercise 6.7. Prove that the set of units form a group under multiplication.

6.1.2 Characteristic of a ring

Rings have two identities $e_\times$ and e_+ (we will denote them by 1 and 0 respectively). For a ring an important criteria is the additive group generated by 1. The elements of that group are 1, $1+1$, $1+1+1$ and so on. The smallest number of times we need to sum 1 to get 0 is called the *characteristic* of the ring.

For some cases, like reals, the sum never reaches the additive identity 0. In these cases we say that the characteristic is *zero*.

Exercise 6.8. Prove that $1 \times 0 = 0$ in a ring.

6.1.3 Homomorphism for a ring

We have already defined the homomorphism for a group. How should we define the homomorphism for a ring?

Exercise 6.9. Try to come up with a definition of ring homomorphism. Remember that the mapping should be well behaved with respect to both the operators.

When not clear from the context, we specify if it is a group homomorphism or a ring isomorphism.

We can define the *kernel* of a homomorphism $\phi : R \to S$ from a ring R to ring S as the set of elements of R which map to the additive identity 0 of S. A bijective homomorphism is called an isomorphism.

We showed in previous lectures that the kernel of a group homomorphism is a normal subgroup. What about the kernel of a ring homomorphism? For this, the concept of ideals will be defined.

6.1.4 Ideal

The ring R is a group under addition. A subgroup I of R under addition is called an *ideal* if

$$\forall x \in I, r \in R : \quad xr, rx \in I$$

For example, the set of all elements divisible by n is an ideal in $\mathbb{Z}$.

Exercise 6.10. Show that $n\mathbb{Z}$ is an ideal of $\mathbb{Z}$.

Ideal is similar to the normal subgroup, but belongs to a ring. Suppose I is an ideal. Then we can define the set of cosets of I with respect to R as $\frac{R}{I}$. We denote the elements of the set by $r + I$.

We know that $\frac{R}{I}$ is a group (why?), but it can be shown that it is a ring under the following operations too.

$$(r + I) + (S + I) = (r + s) + I \quad (r + I) \times (s + I) = (rs) + I$$

Exercise 6.11. Show that the kernel of a ring homomorphism is an ideal.

Kernel of a any ring homomorphism is an ideal and every ideal can be viewed this way. We can define quotient ring using ideals as we defined quotient group using normal subgroup. It turns out,

Theorem 6.12. *Given a homomorphism $\phi : R \to S$,*

$$\frac{R}{Ker(\phi)} \cong Img(\phi)$$

Given a set $S \subseteq I$, we can always come up with the ideal generated by the set. Suppose the multiplication is commutative, then

$$I = \{r_1x_1 + r_2x_2 + \cdots + r_nx_n : \ \forall i \ \ r_i \in R, x_i \in S\},$$

is the ideal generated by S.

Exercise 6.13. Prove that it is an ideal.

6.2 Chinese remainder theorem

One of the most important ways to create a big ring using two small rings is called *direct product.* Suppose the two given rings are R and S. The direct product $T = R \times S$ is a ring with first element from R and second element from S.

$$T = \{(r, s) : \ \ r \in R \text{ and } s \in S\}$$

The two binary operations in ring T are defined by taking the operations component-wise in R and S.

$$(r_1, s_1) + (r_2, s_2) = (r_1 + r_2, s_1 + s_2) \text{ and } (r_1, s_1)(r_2, s_2) = (r_1r_2, s_1s_2)$$

The motivation for *Chinese remainder theorem* is to break the ring $\mathbb{Z}_m$ into smaller parts (rings modulo smaller numbers).

Exercise 6.14. Come up with an isomorphism between $\mathbb{Z}_6$ and $\mathbb{Z}_2 \times \mathbb{Z}_3$.

It might seem that we can break $\mathbb{Z}_{mn}$ to $\mathbb{Z}_m \times \mathbb{Z}_n$.

Exercise 6.15. Show that there is no isomorphism between $\mathbb{Z}_4$ and $\mathbb{Z}_2 \times \mathbb{Z}_2$.

It turns out, in the last exercise, 2 and 3 being co-prime to each other is important. We need to define when two ideals are "co-prime" to each other.

Definition 6.16. *The ideals A and B are said to be comaximal if $A + B = R$. Here $A + B = \{a + b : \ a \in A \text{ and } b \in B\}$.*

The definition of comaximal basically says that there exist $x \in A$ and $y \in B$, s.t., $x + y = 1$.

Note 6.17. Similarly we can define AB to be the ideal with *finite sums* of kind ab where $a \in A$ and $b \in B$.

Exercise 6.18. Notice that $S = \{ab : a \in A, b \in B\}$ need not be an ideal. Show that AB as defined above is an ideal.

Exercise 6.19. If $A_1, A_2, \cdots, A_k$ are pairwise comaximal then show that A_1 and $A_2 \cdots A_k$ are comaximal too.

With all these definitions (direct product, comaximal) we are ready to state the Chinese remaindering theorem. We will assume that the ring is commutative.

Theorem 6.20. *Chinese remainder theorem (CRT): Let $A_1, A_2, \cdots, A_k$ be ideals in ring R. The natural map which takes $r \in R$ to $(r + A_1, r + A_2, \cdots, r + A_k) \in \frac{R}{A_1} \times \frac{R}{A_2} \times \cdots \times \frac{R}{A_k}$ is a ring homomorphism. If all pairs A_i, A_j are comaximal then the homomorphism is actually surjective (onto) and,*

$$\frac{R}{A_1 A_2 \cdots A_k} \cong \frac{R}{A_1} \times \frac{R}{A_2} \times \cdots \times \frac{R}{A_k}.$$

Proof. We will first show this for $k = 2$ and then it can be extended by induction (the exercise that A_1 and $A_2 \cdots A_k$ are comaximal will prove it).

The proof can be broken down into three parts.

1. The map ϕ which takes r to $r + A_1, r + A_2$ is a homomorphism.
2. The kernel ϕ is $A_1 A_2 \cdots A_k$.
3. The image is $\frac{R}{A_1} \times \frac{R}{A_2} \times \cdots \times \frac{R}{A_k}$. In other words the map ϕ is onto(surjective).

The first part is an exercise. It follows from the fact that the individual maps ($\delta_i : R \to \frac{R}{A_i}$) which take r to $r + A_i$ are homomorphisms.

The kernel for this individual maps are A_i's and hence for the combined map ϕ, it is $A_1 \cap A_2$. The second part of the proof requires us to prove that if A_1, A_2 are comaximal then $A_1 \cap A_2 = A_1 A_2$.

Suppose A_1 and A_2 are comaximal. Hence, there exist $x \in A_1, y \in A_2$ for which $x + y = 1$. Even without the comaximal condition $A_1 A_2 \subseteq A_1 \cap A_2$. For the opposite direction, say $c \in A_1 \cap A_2$, then $c = c1 = cx + cy \in A_1 A_2$ (there exist $x \in A_1, y \in A_2$ for which $x + y = 1$). Hence $A_1 \cap A_2 = A_1 A_2$.

Now we only need to prove the third part, to show that the map $\phi : r \to (r + A_1, r + A_2)$ is surjective. Since $x + y = 1$, $\phi(x) = (0, 1)$ and $\phi(y) = (1, 0)$. For any element $(r_1 + A_1, r_2 + A_2)$ of $\frac{R}{A_1} \times \frac{R}{A_2}$, we can prove $\phi(r_2 x + r_1 y) = (r_1 + A_1, r_2 + A_2)$. Hence ϕ is surjective.

$$\phi(r_2 x + r_1 y) = \phi(r_2 x) + \phi(r_1 y) = (A_1, r_2 + A_2) + (r_1 + A_1, A_2) = (r_1 + A_1, r_2 + A_2).$$

□

We will see various applications of Chinese remaindering theorem throughout this course. The most important one is, given a number $n = p_1^{a_1} \cdots p_r^{a_r}$,

$$\mathbb{Z}_n \cong \mathbb{Z}_{p_1^{a_1}} \mathbb{Z}_{p_2^{a_2}} \cdots \mathbb{Z}_{p_r^{a_r}}.$$

The proof is left as an exercise.

This isomorphism and its proof will enable us to answer one of the questions posted earlier. Suppose we need to find a number r which leaves remainder r_1 modulo n_1 and remainder r_2 modulo n_2. Chinese remainder theorem tells us that such a r *always exists* if n_1 and n_2 are co-prime to each other. Through the proof of CRT,

$$r = r_1 n_2 (n_2^{-1} \mod n_1) + r_2 n_1 (n_1^{-1} \mod n_2).$$

Exercise 6.21. Check that the above solution works.

The same can be generalized to more than 2 numbers. How (try to give the explicit formula)?

Now, we will consider two abstract structures which are specialization of rings, integral domains and fields.

6.3 Integral domain

Our main motivation was to study integers. We know that integers are rings but they are not fields. We also saw (through exercise) that integers are more special than rings. The next abstract structure is very close to integers and is called *integral domain*.

An *integral domain* is a commutative ring (multiplication is commutative) where product of two non-zero elements is also non-zero. In other words, if $ab = 0$ then either $a = 0$ or $b = 0$ or both.

Exercise 6.22. Give some examples of an integral domain. Give some examples of rings which are not integral domains.

We said that integral domain is closer to integers than rings. The first thing to notice is that integral domains have cancellation property.

Exercise 6.23. If $ab = ac$ in an integral domain, then either $a = 0$ or $b = c$.

Now we will see that the properties of divisibility, primes etc. can be defined for integral domains.

Given two elements $a, b \in R$, we say that a *divides* b (b is a *multiple* of a) if there exist an $x \in R$, s.t., $ax = b$.

Exercise 6.24. If a divides b and b divides a then they are called *associates*. Show,

- Being associates is an equivalence relation.
- a and b are associates iff $a = ub$ where u is a unit.

You can guess (from the example of integers), the numbers 0 and units (± 1) are not relevant for divisibility. A non-zero non-unit x is *irreducible* if it can't be expressed as a product of two non-zero non-units. A non-zero non-unit x is *prime* if whenever x divides ab, it divides either a or b.

Notice that for integers the definition of irreducible and prime is the same. But this need not be true in general for integral domain. For examples, look at any standard text.

Exercise 6.25. What is the problem with defining divisibility in ring?

6.4 Fields

If you look at the definition of rings, it seems we were a bit unfair towards *multiplication*. R was a commutative group under addition but for multiplication the properties were very relaxed (no inverses, no commutativity). *Field* is the abstract structure where the set is *almost* a commutative group under multiplication.

Definition 6.26. *The set F with the two operations $+$ and $\times$ is a* field*, if,*

- *F is a commutative group under $+$.*
- *$F - \{0\}$ is a commutative group under $\times$ (it has inverses).*
- *The two operations $+$ and $\times$ follow the distributive law, i.e.,*

$$a \times (b + c) = a \times b + a \times c \textit{ and } (a + b) \times c = a \times c + b \times c.$$

Exercise 6.27. Why are we excluding the identity of addition when the multiplicative group is defined?

As you can see Field has the strongest structure (most properties) among the things (groups, rings etc..) we have studied. Hence many theorems can be proven using Fields. Fields is one of the most important abstract structure for computer scientists.

Note 6.28. The notion of divisibility etc. are trivial in fields.

Let us look at some of the examples of fields.

- $\mathbb{Z}$ is NOT a field.
- $\mathbb{Q}$, $\mathbb{R}$ and $\mathbb{C}$ are fields.
- $\mathbb{Z}_m$ is a field iff m is a _______. Ex: Fill in the blank.

The last example is of fields which have finite size. These fields are called *finite fields* and will be of great interest to us.

6.5 The chain of abstract structures (advanced)

We have studied three different abstract structures this week, ring, integral domain and fields. Actually there are a lot of abstract structures which can arise in between rings and fields. They are defined by the properties which have been fundamental in the study of number theory. Take a look at the definition of all of these structures and the relation (order) between the properties.

Exercise 6.29. For how many of them can you guess the defining properties?

Rings $\supset$ Commutative Rings $\supset$ Integral domain $\supset$ Unique factorization domain $\supset$ Principal ideal domain $\supset$ Euclidean domain $\supset$ Field

This list is taken from Wikipedia. You can interpret this chain of inclusion as the fact that Euclidean gcd algorithm (Euclidean domain) implies the every any number of the form $ax + by$ can be written as $dgcd(x, y)$ (principal ideal domain). And principal ideal domain implies unique factorization. Then unique factorization implies, $ab = ac \;\Rightarrow\; b = c$ assuming $a \neq 0$.

Exercise 6.30. Prove all the above assertions.

6.6 Assignment

Exercise 6.31. Give a rule that is satisfied by Integers but need not be satisfied by rings in general.

Exercise 6.32. Find the set of units in the ring $\mathbb{Z}_8$.

Exercise 6.33. If all the ideals in the ring can be generated by a single element then it is called a *principal ideal domain*. Show that $\mathbb{Z}$ is a principal ideal domain.

Exercise 6.34. Show that if $ab = 0$ for a, b in a field F then show that either $a = 0$ or $b = 0$.

Exercise 6.35. What are the units of a field?

Exercise 6.36. Show that a finite integral domain is a field.

Exercise 6.37. Show that the characteristic of a finite field is always a prime.

Exercise 6.38. Find a number n which leaves remainder 23 with 31, 2 with 37 and 61 with 73.

Exercise 6.39. Given a number $n = p_1^{a_1} \cdots p_r^{a_r}$, show that,

$$\mathbb{Z}_n \cong \mathbb{Z}_{p_1^{a_1}} \mathbb{Z}_{p_2^{a_2}} \cdots \mathbb{Z}_{p_r^{a_r}}.$$

Where $\cong$ denotes that two rings are isomorphic.

Exercise 6.40. Find a number n which leaves remainder 3 when divided by 33 and 62 when divided by 81.

Hint: Trick question.

Exercise 6.41. Suppose $\phi(n)$ is the number of elements co-prime to n. Prove that if m and n are co-prime, then $\phi(mn) = \phi(m)\phi(n)$.

Hint: Chinese remainder theorem.

Exercise 6.42. Show that $m\mathbb{Z}$ and $n\mathbb{Z}$ are comaximal in $\mathbb{Z}$.

7

Polynomials

You have seen polynomials many a times till now. The purpose of this lecture is to give a formal treatment to constructing polynomials and the rules over them. We will re derive many properties of polynomials with the only assumption that the coefficients arise from a ring or a field.

We are used to thinking of a polynomial (like $4x^2 + 2x + 6$) as an expression of coefficients (in $\mathbb{Z}, \mathbb{R}$ etc.) and variables (x, y etc.). Mostly the purpose is to solve equations and find out the value of the variable or indeterminate. But the polynomials are useful not just to figure out the value of the variable but as a structure itself. The values $x, x^2, \cdots$ should be thought of as placeholder to signify the position of the coefficients. Using this view lets define a *formal polynomial*.

7.1 Polynomials over a ring

A polynomial over a ring R is a formal sum $a_n x^n + \cdots + a_1 x + a_0$, where the coefficients come from the ring R. The set of all polynomials (in one variable) over a ring R are denoted by $R[x]$. The *degree* of the polynomial is the highest power of x with a non-zero coefficient.

We can define the addition and multiplication over polynomials (in $R[x]$) so as to match the definitions learned till now. Given two polynomials $a(x) = a_n x^n + \cdots + a_1 x + a_0$ and $b(x) = b_n x^n + \cdots + b_1 x + b_0$, their sum is defined as,

$$a(x) + b(x) = (a_n + b_n)x^n + \cdots + (a_1 + b_1)x + (a_0 + b_0).$$

If the degree of two polynomials is not equal, we can introduce extra zero coefficients in the polynomial with the smaller degree. For multiplication, given two polynomials $a(x) = a_n x^n + \cdots + a_1 x + a_0$ and $b(x) = b_m x^m + \cdots + b_1 x + b_0$, their product is defined as,

$$p(x) = a(x)b(x) = (a_n b_m)x^{n+m} + \cdots + (a_2 b_0 + a_1 b_1 + a_0 b_2)x^2 + (a_0 b_1 + a_1 b_0)x + (a_0 b_0).$$

More formally, the product is defined using distribution and the fact that $(ax^i)(bx^j) = (ab)x^{i+j}$.

Exercise 7.1. What is the degree of $a(x)b(x)$ if the degree of $a(x)$ is n and $b(x)$ is m?

Hint: It need not be $n + m$. why?

We mentioned while giving examples of rings that if R is a commutative ring then $R[x]$ is a commutative ring too. Using the definition above, the polynomials in multiple variables can be defined using induction. We can consider $R[x_1, x_2, \cdots, x_k]$ to be the ring of polynomials whose coefficients come from $R[x_1, x_2, \cdots, x_{k-1}]$.

Another definition of interest is the *monic* polynomial whose leading coefficient (non-zero coefficient of the highest degree) is 1. A polynomial is *constant* iff the only non-zero coefficient is the degree 0 one (a_0).

The most important polynomial rings for us would be $\mathbb{Z}_m[x]$ and $\mathbb{Z}[x], \mathbb{R}[x]$ etc..

Exercise 7.2. Suppose $a(x) = 2x^3 + 2x^2 + 2$ is a polynomial in $\mathbb{Z}_4[x]$, what is $a(x)^2$?

7.2 Polynomials over fields

After defining addition and multiplication we would like to define division and gcd of polynomials. It turns out that these definitions make sense when R is a field. For this section, we will assume that we are given a field F and the polynomials are in $F[x]$. We know that $F[x]$ is an integral domain since F is one.

Exercise 7.3. Show that $F[x]$ is an integral domain if F is an integral domain.

Exercise 7.4. For the remaining section, note where we use that the underlying ring of coefficients F is a field.

Theorem 7.5. *Division: Given two polynomials $f(x)$ and $g(x)$, there exist two* unique *polynomials called quotient $q(x)$ and remainder $r(x)$, s.t.,*

$$f(x) = q(x)g(x) + r(x).$$

where the degree of $r(x)$ is less than the degree of $g(x)$.

Proof. Existence: Suppose the degree of $f(x)$ is less than degree of $g(x)$ then $q(x) = 0$ and $r(x) = f(x)$. This will be the base case and we will apply induction on the degree of $f(x)$.

Say $f(x) = f_n x^n + \cdots + f_1 x + f_0$ and $g(x) = g_m x^m + \cdots + g_1 x + g_0$ with $m \leq n$. Multiply g by $f_n g_m^{-1} x^{n-m}$ and subtract it from f.

$$f(x) - f_n g_m^{-1} x^{n-m} g(x) = (f_{n-1} - g_{m-1} f_n g_m^{-1}) x^{n-m-1} + \cdots = l(x).$$

So l is a polynomial with lower degree and by induction it can be written as $l(x) = q'(x)g(x) + r(x)$. This implies $f(x) = (f_n g_m^{-1} x^{n-m} + q'(x))g(x) + r(x)$. So we can always find $q(x)$ and $r(x)$ with the condition given above. This method is called *long division* and is the usual method of dividing two numbers.

Exercise 7.6. What is the relation between the usual division between two integers you learnt in elementary classes and long division.

Uniqueness: Suppose there are two such decompositions $f = q_1 g + r_1$ and $f = q_2 g + r_2$ (notice that we have suppressed x for the sake of brevity). Then subtracting one from another,

$$0 = (q_1 - q_2)g + (r_1 - r_2).$$

Exercise 7.7. Show that this implies q and r are unique.

□

Using the division algorithm, we can define the Euclidean GCD algorithm.

Exercise 7.8. Read and understand the Euclidean gcd algorithm for two positive numbers.

Lets define *greatest common divisor (gcd)* first. Given two polynomials f, g, their greatest common divisor is the highest degree polynomial which divided both f, g. The important observation for Euclidean gcd is, if $f = gq_1 + r_1$ then

$$gcd(f, g) = gcd(g, r_1).$$

Without loss of generality we can assume that f has higher degree than g and hence r has lower degree than g and f. We can continue this process, say $g = q_2 r_1 + r_2$. Then the task reduces to finding the gcd of r_1 and r_2. Ultimately we get two polynomials, s.t., $r_n \mid r_{n-1}$. Then r_n is the gcd of f and g.

Exercise 7.9. Show that any polynomial which divides both f and g will also divide r_n mentioned above. Show that r_n divides both f and g.

From the previous exercise it is clear that r_n is one of the gcd (it divides both and has highest degree).

Exercise 7.10. Why is gcd unique?

Exercise 7.11. Imp: Show that using Euclidean algorithm for gcd, if $gcd(f, g) = d$ then there exist two polynomials p, q, s.t., $d = pf + qg$.

Lets define *primes* in the ring of polynomials. They are called *irreducible* polynomials (irreducible elements of integral domain $F[x]$). A polynomial f is *irreducible* iff it is not constant and there does NOT exist two non-constant polynomials g and h, s.t., $f = gh$.

Exercise 7.12. Given that a monic polynomial g is irreducible, show, any polynomial f is divisible by g or their gcd is 1. This property can be re-stated, any irreducible polynomial can't have a non-trivial gcd (trivial gcd: 1 or the polynomial itself).

With this definition we can start finding the factors of any polynomial f. Either f is irreducible or it can be written as gh. If we keep applying this procedure to g and h. We get that any polynomial f can be written as,

$$f = cg_1g_2 \cdots g_k.$$

Where g_i's are irreducible monic polynomials and c is a constant in the field F.

Can two such factorizations exist? It turns out, like in the case of natural numbers, this factorization is unique up to ordering of polynomials. For the contradiction, suppose there are two such factorizations $cg_1 \cdots g_k$ and $ch_1 \cdots h_l$.

Exercise 7.13. Why can we assume that the constant is the same for both factorizations?

We know that since g_1 is irreducible it can't have a non-trivial gcd with either $h = h_1 \cdots h_{l-1}$ or h_l. We will also show that it can't have gcd 1 with both. Suppose $gcd(h_l, g_1)$ is 1. Then using Euclidean gcd,

$$1 = ph_l + qg_1 \Rightarrow h = pf + qg_1h.$$

Since g_1 divides both terms on the R.H.S, it divides h. Hence the gcd of h and g_1 is g_1. So g_1 either divides h_l or $h = h_1 \cdots h_{l-1}$.

If it divides $h_1 \cdots h_{l-1}$, we can further divide it and ultimately get that g_1 divides h_i for some i. But since g_1 and h_i both are irreducible and monic, hence $g_1 = h_i$. This gives the theorem,

Theorem 7.14. *Unique factorization: Given a polynomial f it can be written in a unique way as a product of irreducible monic polynomials up to ordering.*

$$f(x) = cg_1(x)g_2(x) \cdots g_k(x)$$

Where c is a constant in F (the leading coefficient of f) and g_i's are irreducible monic polynomials.

Exercise 7.15. The order of going from division algorithm to Euclidean GCD to unique factorization is important. Where else have you seen this?

There is an easy way to find out whether a degree 1 polynomial $x - a$ divides a polynomial f or not. Substitute a in the polynomial f (we call the evaluation f(a)), if it evaluates to zero then $x - a$ divides f otherwise not. If $f(a) = 0$, we say that a is a *root* of f. The proof of this is given as an exercise.

Using the factorization theorem we can show that any polynomial of degree d can have at most d roots. The proof of this theorem is left as an exercise.

Theorem 7.16. *Given a polynomial p of degree d over a field F. There are at most d distinct roots of p.*

7.3 Field extension

Exercise 7.17. Why do we need complex numbers?

It might seem a weird question given the context. But lets look at the answer first. Mathematician didn't have the roots of polynomial $x^2 + 1$ in the field $\mathbb{R}$. So they came up with another field $\mathbb{C}$ where the solution existed.

Can we do it for other fields and other polynomials? This can be done and is known as *field extension*. Lets try to construct such a field extension.

Suppose we are given a field F and a polynomial p in it. We can look at the set of all polynomial in $F[x]$ modulo the polynomial p. This set is known as $\frac{F[x]}{(p)}$. The reason for this is that (p) is an ideal generated by polynomial p (it contains all multiples of p).

Exercise 7.18. Show that (p) is an ideal.

So $\frac{F[x]}{(p)}$ is just the quotient ring generated by the ideal (p).

A more intuitive way to understand this ring is, it is the set of polynomials in $F[x]$ assuming that two polynomials are equal if they only differ by a multiple of p. Using the division algorithm we can always reduce any polynomial f to a polynomial r, s.t., $f = qp + r$. Then by above discussion $r = f$ in $\frac{F[x]}{(p)}$. In the algebraic language, r is the representative of the additive coset of $F[x]$ containing f.

Exercise 7.19. Show that the elements of $\frac{F[x]}{(p)}$ are basically all the polynomials with degree less than $deg(p)$.

The ring $\frac{F[x]}{(p)}$ is a field iff p is irreducible (proof is left as an exercise). This field $\frac{F[x]}{(p)}$ is called the field extension of F. It is easy to see that an isomorphic copy of F is a subfield of $\frac{F[x]}{(p)}$.

The great thing is that there is a root of p in this new field. If you think for a minute the root is x !!

The field $\frac{F[x]}{(p)}$ can also be viewed as a vector space over F.

Exercise 7.20. What is the dimension of that vector space?

We are interested in these field extensions because they will help us characterize finite fields.

7.3.1 Another way to look at field extension

More general way to define field extensions is through subfields. Suppose K is a subset of a field L, s.t., K is field itself. Then we say that K is a sub-field of L or L is an extension of K.

Suppose s is an element of L not in K. We can extend K to include s. The smallest subfield of L which contains K as well as s is called $K(s)$. We can similarly define $K(S)$, where S is a set.

Why are the two interpretations given above similar. If s was a root of an irreducible polynomial p in K then the field $K(s)$ is precisely $\frac{K[x]}{(p)}$. All the elements of $K(s)$ can be viewed as polynomials in $K[x]$ with degree less than $deg(p)$ (substituting s for x).

Exercise 7.21. Show that $\mathbb{Q}(\sqrt{2}) = \{a + b\sqrt{2} : \quad a, b \in \mathbb{Q}\}$ is a subfield of $\mathbb{R}$.

Notice that complex numbers can be viewed as $\mathbb{R}(i)$ or as $\frac{\mathbb{R}[x]}{x^2+1}$.

Suppose L is a field extension of K. Then L can be seen as a vector space over K. The dimension of the vector space is known as the *degree* of the extension.

Note 7.22. You might have seen vector spaces over reals and complex numbers. They can be defined over arbitrary field F by assuming that the coefficients (scalars) come from F and any addition and multiplication of scalars can be done in accordance with the field.

Exercise 7.23. What is the degree of extension $\frac{K[x]}{p}$.

Note that the subfield perspective of field extension is more general than the field extension using polynomials. Reals are an extension of rationals. It can be shown that such an extension cannot be obtained by any irreducible polynomial over rationals.

7.4 Assignment

Exercise 7.24. Write a program to compute the coefficient of x^i in $a(x)b(x)$ given two polynomials $a(x)$ and $b(x)$.

Exercise 7.25. Compute the product of $7x^3 + 2x^2 + 2x + 4$ and $2x^2 + 5x + 1$ in $\mathbb{Z}_{14}$.

Exercise 7.26. Show that in $\mathbb{Z}_p$ (p is a prime), $(x + y)^p = x^p + y^p$.

Exercise 7.27. Show that if R is an integral domain then so is $R[x]$.

Exercise 7.28. Show that $f(x)$ in $F[x]$ has an inverse iff $f(x)$ is a constant polynomial (zero excluded).

Exercise 7.29. Find out the quotient and remainder when $x^3 + 5x^2 + 2x + 3$ is divided by $x^2 + 1$ in $\mathbb{Z}_7$.

Exercise 7.30. If F is a field, is $F[x]$ also a field?

Exercise 7.31. What is the gcd of $x^n + 1$ and $x^m - 1$ in $\mathbb{Z}_2[x]$.

Exercise 7.32. Show that $x - a$ divides f iff $f(a) = 0$.

Exercise 7.33. Hard: Prove that the ring $\frac{F[x]}{(p)}$ is a field iff p is irreducible.

Exercise 7.34. Prove the theorem 7.16.

8

Finite Fields

We have learnt about groups, rings, integral domains and fields till now. Fields have the maximum required properties and hence many nice theorems can be proved about them. For instance, in previous lectures we saw that the polynomials with coefficients from fields have unique factorization theorem.

One of the important sub case of fields is when they are finite. In this case the fields can be completely characterized up to isomorphism and have lot of applications in computer science. We will cover the characterization and an application in these lecture notes.

8.1 Characteristic of a field

We have seen how the characteristic of a ring was defined.

Exercise 8.1. What is the characteristic of a ring?

Since field is a special case of rings, the definition can be applied to fields too. The characteristic of a field F is the minimum $n \in \mathbb{N}$, s.t., $n1 = 0$. Here $n1$ denotes the addition of multiplicative identity n times,

$$\underbrace{1 + 1 + \cdots + 1}_{n \text{ times}}.$$

In general, the characteristic might not exist for field (say $\mathbb{R}$). In that case we say that characteristic is zero. For the case of finite field though, the characteristic is always a positive number. Why?

Suppose n is a characteristic of a finite field. If n is composite, say $n = pq$, then $(p1)(q1) = 0$. But F does not have a zero divisor (it is a field) and hence either $p1 = 0$ or $q1 = 0$, establishing contradiction. So we get the theorem,

Theorem 8.2. *The characteristic of a finite field is always a prime.*

Note 8.3. $p1 = 0$ implies that $pf = 0$ for all $f \in F$, the proof is given as an exercise.

How does a field with characteristic p looks like? We can look at the additive structure. It turns out that it can be seen as a vector space over $\mathbb{Z}_p$.

Exercise 8.4. Review the definition of a vector space over a field.

Theorem 8.5. *A finite field F of characteristic p is a vector space over $\mathbb{Z}_p$. Hence, if there are r basis elements then $|F| = p^r$.*

Proof. Define nf to be $\underbrace{f + f + \cdots + f}_{n \text{ times}}$. From the previous discussion, the only relevant values of n are $\{0, 1, \cdots, p-1\}$.

Let us look at the set generated by $S = \{f_1, f_2, \cdots, f_k\}$. We call it the *span*,

$$span(S) = \{n_1 f_1 + n_2 f_2 + \cdots + n_k f_k : \ n_i \in \{0, 1, \cdots, p-1\} \ \forall i\}.$$

Exercise 8.6. Show that $span(S)$ is the smallest additive group containing S.

Clearly one set exist for which span is the entire field (the field itself).

Exercise 8.7. Show that F is a vector space over $\mathbb{Z}_p$.

Say a basis $B = \{b_1, b_2, \cdots, b_r\}$ is the *minimal* set of elements such that $span(B) = F$. We have assumed that B has r elements. Then,

$$span(B) = \{n_1 b_1 + n_2 b_2 + \cdots + n_r b_r : \ n_i \in \{0, 1, \cdots, p-1\} \ \forall i\}.$$

We claim that no two elements of the above set are same. If they are then some element of B can be written as a linear combination of others, violating the minimalness of B. Hence $span(B)$ has no duplicates and it is equal to F. So the cardinality of F is p^r.

□

Note 8.8. The theorem shows that as an additive group, a field of size p^r, is isomorphic to $(\mathbb{Z}_p)^r$.

By the previous theorem we have proved that every finite field has characteristic some prime p and number of elements are p^r, some power of its characteristic. Hence the number of elements in a finite field can only be a prime power.

Does there exist a finite field for every prime power. Clearly for every p, $\mathbb{Z}_p$ is a field.

8.1.1 Finite fields of order p^r

To construct fields of cardinality p^r, we use the concept of field extension. Suppose g is an irreducible polynomial in $\mathbb{Z}_p$. Then we know that $\frac{\mathbb{Z}_p[x]}{(g)}$ is a field (from field extensions).

Exercise 8.9. Show that $\frac{\mathbb{Z}_3[x]}{x^2+1}$ is a field. What is its cardinality? What is the characteristic?

It is clear that in such a field $p1 = 0$. That shows that characteristic of the field is p. The different elements of this field are all the remainder polynomials modulo g. In other words, all the polynomials of degree $deg(g) - 1$ with coefficients from $\mathbb{Z}_p$. So the number of elements in this field are $p^{deg(g)}$.

This shows that to construct a finite field of size p^r, we need to find an irreducible polynomial of degree r. It is known that such an irreducible polynomial always exist. The proof of this statement will not be covered in this class.

So there always exist at least one field of size p^r. It can actually be shown that all such fields of size p^r are isomorphic and we call them $\mathbb{F}_{p^r}$. For $r = 1$, this field is $\mathbb{Z}_p$, we will also call it $\mathbb{F}_p$.

Exercise 8.10. What is the difference between vector space $\mathbb{Z}_3^2$ and field $\frac{\mathbb{Z}_3[x]}{x^2+1}$?

We won't prove that there exist a unique field of size p^r up to isomorphism. But we will provide a partial justification. We have seen that the additive group of any field of size p^r is isomorphic to $(\mathbb{Z}_p)^r$. In the next section we will show that their multiplicative group is also isomorphic to $\mathbb{Z}_{p^r-1}$ (it is cyclic). So for any two finite fields of same size, their additive groups and multiplicative groups are isomorphic.

Exercise 8.11. Why is this a partial and not full proof that two fields of the same size are isomorphic?

8.1.2 Primitive element

We need to show that the multiplicative group of any field is cyclic. That means, there exist an element $f \in F$, s.t., the order of f is $|F| - 1$ (why did we subtract 1?). Such an element generates the whole group $F - \{0\} = \{f^0, f^1, \cdots, f^{|F|-2}\}$.

Definition 8.12. *Primitive element: An element f of F which generates the multiplicative group of the field F is called the* primitive element *of F.*

To show that any field's multiplicative group is cyclic, we just need to show the existence of a primitive element.

Theorem 8.13. *For any finite field F, there always exist a primitive element of F.*

Proof. Lets call the multiplicative group $F^* = F - \{0\}$ and $|F^*| = n$. Since F^* has order n, for all elements x of F^*,

$$x^n - 1 = 0$$

So there are exactly n roots of the above equation (why exactly n?).

For any element x, the order d divides n, hence x is a solution of $p(d) = x^d - 1$ for some $d \mid n$. Notice that the polynomial $p(d)$ has at most d roots.

For the sake of contradiction, suppose there are no primitive elements. Then every element has order strictly less than n. We would like to show that there are not enough roots (n) for the polynomial $x^n - 1$.

So we would like to show,

$$\sum_{d<n, d|n} d < n \tag{8.1}$$

Note 8.14. There is a strict inequality $d < n$ in the summation index as well as the inequality.

Exercise 8.15. Show that this is not true for some n.

The reason why the above strategy does not work is that we are counting lot of elements multiple times. A solution of $p(d)$ will be a solution of $p(2d), p(3d), \cdots$. There is a decent chance that some of numbers $2d, 3d, \cdots$ might be divisors of n too.

So say $e(d)$ is the number of elements with order *exactly* d. Hence instead of Eq. 8.1, the contradiction will be shown by proving the equation,

$$\sum_{d<n, d|n} e(d) < n \tag{8.2}$$

This equation follows from the following two claims. The proof of first one is left as an exercise, other will be proved here.

Note 8.16. $\phi(d)$ is number of elements co-prime (gcd 1) to d.

Claim. For a number n, $\sum_{d|n} \phi(d) = n$.

Proof hint: For any number $k \leq n$, look at $gcd(k, n)$ and $\frac{k}{gcd(k,n)}$.

Claim. If there exist an element of order d then $\phi(d) = e(d)$.

Proof. Suppose the element with order d is x. Then the d roots for $x^d - 1$ are precisely $x^0, x^1, \cdots, x^{d-1}$ (these are d roots and there are at most d roots). The order of x^k is $\frac{d}{gcd(d,k)}$.

Exercise 8.17. Suppose the order of x in a group G is d. Show that for x^k, the order is $\frac{d}{gcd(d,k)}$.

Hence the elements with order d are precisely x^k, s.t., $gcd(d, k) = 1$. So $e(d) = \phi(d)$.

□

Using the claims,

$$n = \sum_{d|n} \phi(d) > \sum_{d|n} e(d).$$

The inequality follows because $e(d) \leq \phi(d)$ and we have assumed $e(n) = 0$. So the equation 8.2 follows from non-existence of primitive element and hence we get the contradiction.

Note 8.18. By definition of $e(d)$, $\sum_{d|n} e(d) = n$. Hence there should be equality in the above equation. That means there are exactly $\phi(d)$ elements of order d in a field n where $d \mid n$. Specifically, there are $\phi(n)$ primitive elements for a field F with size $n+1$.

□

Since $\mathbb{Z}_p$ is a field, by previous theorem, $\mathbb{F}_p = \mathbb{Z}_p$ is cyclic as a multiplicative group. This can be generalized to show that even $\mathbb{Z}_{p^k}^{\times}$ is cyclic.

Exercise 8.19. Show that $\mathbb{Z}_{p^k}^{\times}$ is NOT isomorphic to the multiplicative group of $\mathbb{F}_{p^k}$ for $k > 1$.

Theorem 8.20. *If $n = p^k$ for some power k of an odd prime p then $G = \mathbb{Z}_n^{\times}$ is cyclic.*

Note 8.21. This is not true for even prime, we have seen that $\mathbb{Z}_8^{\times}$ is not cyclic.

Exercise 8.22. Find out where did we use the fact that p is odd.

Proof. Assume that $t = p^{k-1}(p-1)$, the order of the group G.

We know that $\mathbb{F}_p$ is cyclic and hence have a generator g. We will use g to come up with a generator of G. First notice that,

$$(g+p)^{p-1} = g^{p-1} + (p-1)g^{p-2}p \neq g^{p-1} \mod p^2.$$

So either $(g+p)^{p-1}$ or g^{p-1} is not $1 \mod p^2$. We can assume the latter case, otherwise replace g by $g+p$ in the argument below.

So $g^{p-1} = 1 + k_1 p$ where $p \nmid k_1$. So using binomial theorem,

$$g^{p(p-1)} = (1 + k_1 p)^p = 1 + k_2 p^2.$$

Where $p \nmid k_2$

Exercise 8.23. Continuing this process, show that,

$$g^{p^{e-1}(p-1)} = 1 + k_e p^e,$$

with $p \nmid k_e$.

From the previous exercise $g^t = 1 \mod p^k$ but $g^{t/p} \neq 1 \mod p^k$. The only possible order of g then is $p^{k-1}d$ where d is a divisor of $p-1$ (because the order has to divide t, Lagrange's theorem).

If the order is $p^{k-1}d$, then

$$g^{p^{k-1}d} = 1 \mod p^k = 1 \mod p.$$

But $g^p = g \mod p$ (why?). That implies $g^d = 1 \mod p$. Since $p-1$ is the order of g modulo p (g is the generator), implies $d = p-1$. Hence proved.

□

8.2 Application: The classical part of quantum algorithm for factorization

One of the most important achievements of quantum computing has been to solve factorization in polynomial time. There is no known *efficient* classical algorithm to factorize a number. The problem is easy to state, given a number n, find the factorization of n.

Note 8.24. An efficient algorithm for factorization runs in time polynomial in $\log n$, since the input size is $\log n$ (the number of bits needed to specify n).

The quantum algorithm works by reducing the problem classically to something known as the *hidden subgroup problem (HSP)*. Shor's factorization algorithm (1994) can be reduced to giving an efficient algorithm to solve HSP on a quantum computer.

The quantum algorithm for HSP is out of scope of this course. But we will present the classical reduction from factorization to HSP, a neat application of many things we learnt in this course.

8.2.1 Hidden subgroup problem (HSP)

In the hidden subgroup problem, we are given a group G and a function $f : G \to \mathbb{R}$ which *hides* a subgroup H. By hiding a subgroup means that the functions assign the same value to two elements from the same coset and different values to elements from a different coset. The subgroup H is not known and the task is to find this subgroup.

Note 8.25. For this case, we assume that a black-box is given which computes the value of a function on group elements. In practice, if we can compute the function efficiently then the algorithm for finding hidden subgroup is efficient too.

The interest in this problem is because many problems like order-finding, discrete logarithm can be thought of as HSP's over finite abelian groups. Their is a quantum algorithm for solving HSP over any finite abelian group. If we can solve HSP on non-abelian groups then it can be used to solve important problems like graph isomorphism and shortest vector problem in a lattice.

The problem of order-finding is that given an element g in a group G, find the order of g in G (smallest r, s.t., $g^r = 1$). Lets see how order-finding can be thought of as an example of HSP in $\mathbb{Z}$.

Suppose the order is r (the quantity we need to find). The set of multiples of r form a subgroup of $\mathbb{Z}$ known as $r\mathbb{Z}$. The cosets are the residue classes modulo r. Given an element $x \in \mathbb{Z}$, the function $a^x = a^{x \mod r}$ is constant on cosets and distinct on different cosets.

Exercise 8.26. Prove the above assertion.

This function can be computed efficiently (repeated squaring) and hence order-finding can be posed as a hidden subgroup problem.

Note 8.27. Above discussion shows that order-finding is an HSP over an abelian group ($\mathbb{Z}$, which is not finite). The quantum algorithm for finite abelian groups can be modified to handle this case too.

8.2.2 Factorization to order-finding

In this section we will reduce the factorization of n to order-finding in the group $\mathbb{Z}_n^\times$. Hence, complete the reduction from factorization to hidden subgroup problem.

We will first get rid of the trivial cases, it can be easily checked if the number is even or if $n = m^k$ (take the square root, cubic root etc. up to $\log n$). So it can be assumed that n is a number of type kk' where k and k' are co-prime and odd. We are interested in finding a non-trivial factor of n (not 1 or n). Once found one factor, we can repeat the procedure to find the complete factorization.

Look at the square roots of $1 \mod n$, i.e., b for which $b^2 = 1 \mod n$. Clearly there are two solutions $b = \pm 1 \mod n$. Suppose there exist a $b \neq \pm 1 \mod n$. Then $b^2 - 1$ is divisible by n and $b \pm 1$ is not. So the $gcd(b \pm 1, n)$ will give non-trivial factors of n.

The reduction from factorization to order-finding basically searches for such a b. It can be shown using Chinese remainder theorem that such a b always exists (exercise).

Exercise 8.28. In the if statement of the algorithm why didn't we check that $b = 1 \mod n$?

The only thing we need to show is that there are enough a's for which $b = a^{r/2} \neq \pm 1 \mod n$ is a square-root of $1 \mod n$.

Note 8.29. The quantum algorithm is a probabilistic algorithm, hence showing that there are enough "good" a's works.

Theorem 8.30. *Suppose n is a product of two co-prime numbers $k, k' > 1$. For a randomly chosen a, the probability that a has an even order r and $a^{r/2} \neq -1 \mod n$ is at least $1/4$.*

```
Check if $n$ is even or of the form $n = m^k$ ;
Pick an $a$, s.t., $gcd(a, n) = 1$ (else we have already found a non-trivial factor of $n$) ;
for $i = 1, \cdots$ do
    Find the order of $a$ and call it $r$ (use the quantum algorithm for order-finding) ;
    if $r$ is odd or $a^{r/2} = -1 \mod n$ then
        Pick another $a$ co-prime to $n$ ;
    else
        Found $b = a^{r/2} \neq \pm 1 \mod n$, square root of 1 ;
        Find the non-trivial factors from $gcd(b \pm 1, n)$ ;
        Break;
    end
end
```

Algorithm 1: Algorithm for factorization using order-finding

Proof. This proof is taken from the book Quantum computing and Quantum information by Nielsen and Chuang. We introduce a notation, $pow2(z)$, the highest power of 2 that divides any number z.

First we prove a lemma for a number $q = p^k$, which is a prime power. Say $m = \phi(q) = p^{k-1}(p-1)$ (exercise). By theorem 8.20, $\mathbb{Z}_q^\times$ is cyclic, say g is the generator (m is the least number, s.t., $g^m = 1 \mod q$).

Suppose $l = pow2(m)$ (m is even and hence $l \geq 1$).

Lemma 8.31. *Say, we choose a random element from* $\mathbb{Z}_q^\times$. *With probability* $1/2$, *the order* r *satisfies* $pow2(r) = l$.

Proof. We know that g^t has order $\frac{m}{gcd(m,t)}$. Then it can be easily seen that $pow2(r) = l$ iff t is odd.

□

Now consider the prime factorization $n = p_1^{i_1} \cdots p_s^{i_s}$. By Chinese remainder theorem,

$$\mathbb{Z}_n^\times \cong \mathbb{Z}_{p_1^{i_1}}^\times \times \cdots \times \mathbb{Z}_{p_s^{i_s}}^\times .$$

So, to randomly chose a, we can pick random $a_1, \cdots, a_s$ from the respective $\mathbb{Z}_{p^i}^\times$'s. Say r_j are the orders of a_j modulo $p_j^{i_j}$.

Claim. Suppose the order r of a is odd or $a^{r/2} = -1 \mod n$. Then $pow2(r_j)$ is same for all j.

Proof. The order is odd iff all r_j's are odd. Otherwise, if $a^{r/2} = -1 \mod p_j^{i_j}$ then none of r_j divide $r/2$ (we use the fact that p_i's are not 2).

All the r_j's divide r but not $r/2$, so $pow2(r_j)$ is the same. □

From lemma 8.31, with half the probability, The order r_j of a_j will be such that $pow2(r_j) = l_j$ (where $l_j = pow2(p_j^{i_j - 1}(p_j - 1))$). Call the case when $pow2(r_j) = l_j$ as the "first" case and other the "second" case. We know that both cases happen with probability $1/2$.

Notice that l_j's only depend on n. If all l_j are equal, pick a_1's from first case and a_2 from the second case. If they are unequal, say $l_1 \neq l_2$, then pick the a_1, a_2 from the first case. So in either scenario, r_j's can't be all equal. Which implies r is even and $a^{\frac{r}{2}} \neq 1 \mod n$ (by claim). Since we have only fixed at most 2 cases out of s, the probability is at least $1/4$.

□

Hence the reduction from factorization to order finding is complete.

8.3 Assignment

Exercise 8.32. Biggs: Prove that the set of all elements of type $\underbrace{1+1+\cdots+1}_{n \text{ times}}$ form a subfield.

Exercise 8.33. If $p1 = 0$, prove that $pf = 0$ for all $f \in F$.

Exercise 8.34. Suppose in a field F, $p1 = 0$ for a prime p. Show that the characteristic of that field is p.

Exercise 8.35. Show that any field of size p is isomorphic to $\mathbb{Z}_p$.

Hint: 0 and 1 should exist in that field. Now construct the obvious isomorphism.

Exercise 8.36. Find a primitive element in field $\mathbb{F}_{23}$

Exercise 8.37. Write a program to find if a degree 2 polynomial is irreducible or not in $\mathbb{F}_p$ for a prime p.

Exercise 8.38. Construct the field $\mathbb{F}_{49}$.

Hint: Look at square roots modulo 7.

Exercise 8.39. Prove the claim 8.16.

Hint: Look at any number m less than n as $m = gcd(m, n).m'$.

Exercise 8.40. Discrete logarithm: Given an element a and a generator g in the group $G = \mathbb{Z}_m^\times$, the discrete log is the problem of finding least l, s.t., $g^l = a$. Show that it can be cast as an HSP.

Hint: Use the function $a^x g^y$ where $x, y \in \mathbb{Z}_{|G|}$.

Exercise 8.41. Show that for $n = kk'$ where k, k' are co-prime, there exist a square root of $1 \mod n$ which is not $\pm 1 \mod n$.

Exercise 8.42. If $n = p^k$, show that $\phi(n) = p^{k-1}(p-1)$.