Preface

Many of the challenges and opportunities facing citizens in the twenty-first century require some level of mathematical proficiency. Some obvious ones are optimization problems in business, managing your household's budget, weighing the economic policies and proposals of political candidates, and of course the ever-important quest to build the best fantasy sports team possible and, if not winning your local NCAA basketball pool, at least doing well enough to avoid embarrassment! As important as these are, there are many other applications of mathematics going on quietly around us all the time. In this book we concentrate on issues arising from cryptography, which we'll see is far more than soldiers and terrorists trying to communicate in secret. We use this as the vehicle to introduce you to a lot of good, applicable mathematics; for much of the book all you need is high school algebra and some patience. These are not cookbook problems to help you perfect your math skills, but rather the basis of modern commerce and security! Equally important, you'll gain valuable experience in how to think about and approach difficult problems. This is a highly transferable skill and will serve you well in the years to come.

Cryptography is one of the oldest studies, and one of the most active and important. The word **cryptography** comes from two Greek words: *κρυτττὸςς* (kryptos), meaning secret, and *γρὰφω* (grapho), meaning to write. As these roots imply, it all began with the need for people to communicate securely. The basic setup is that there are two people, and they must be able to quickly, easily, and securely exchange information, often in the presence of an adversary who is actively attempting to intercept and decipher the messages.

In the public mind, the most commonly associated images involve the military. While war stories make for dramatic examples and are very important in both the development of the field and its applications, they are only part of the picture. It's not just a subject for soldiers on the battlefield. Whenever you make an online purchase, you're a player. This example has many of the key features.

The first issue is the most obvious. You need to authorize your credit card company or bank to transfer funds to the merchant; however, you're not face-to-face with the seller, and you have to send your information through a probably very insecure channel. It's imperative that no one is able to obtain your personal information and pretend to be you in future transactions!

There are, however, two other very important items. The process must be fast; people aren't willing to wait minutes to make sure an order has been confirmed. Also, there's always the problem of a message being corrupted. What if some of the message is mistransmitted or misread by the party on the other end? These questions lead us to the study of efficient algorithms and error detection and correction codes. These have found a wealth of applications not just in cryptography, but also in areas where the information is not secret.

Two great examples are streaming video and Universal Product Codes (UPC). In streaming video the information (everything from sports highlights to CSPAN debates) is often unprotected and deliberately meant to be freely available to all; what matters is being able to transmit it quickly and play it correctly on the other end. Fruits and vegetables are some of the few remaining items to resist getting a UPC barcode; these black and white patterns are on almost all products. It may shock you to realize how these are used. It's far more than helping the cashier charge you the proper amount; they're also used to help stores update their inventory in real time as well as correlate and analyze your purchases to better target you in the future! These are both wonderful examples of the need to detect and correct errors.

These examples illustrate that problems and solutions arising from cryptography often have applications in other disciplines. That's why we didn't title this book as an introduction to cryptography, but rather to encryption. Cryptography is of course important in the development of the field, but it's not the entire story.

The purpose of this book is to introduce just enough mathematics to explore these topics and to familiarize you with the issues and challenges of the field. Fortunately, basic algebra and some elementary number theory is enough to describe the systems and methods. This means you can read this book without knowing calculus or linear algebra; however, it's important to understand what "elementary" means. While we don't need to use powerful theorems from advanced mathematics, we do need to be very clever in combining our tools from algebra. Fortunately we're following the paths of giants, who have had numerous "aha moments" and have seen subtle connections between seemingly disparate subjects. We leisurely explore these paths, emphasizing the thought processes that led to these remarkable advances.

Below is a quick summary of what is covered in this book, which we follow with outlines for semester-long courses. Each chapter ends with a collection of problems. Some problems are straightforward applications of material from the text, while others are quite challenging and are introductions to more advanced topics. These problems are meant to supplement the text and to allow students of different levels and interests to explore the material in different ways. Instructors may contact the authors (either directly or through the AMS webpage) to request a complete solution key.

- Chapter 1 is a brief introduction to the history of cryptography. There is not much mathematics here. The purpose is to provide the exciting historical importance and background of cryptography, introduce the terminology, and describe some of the problems and uses.
- Chapter 2 deals with classical methods of encryption. For the most part we postpone the attacks and vulnerabilities of these methods for later chapters, concentrating instead on describing popular methods to encrypt and decrypt messages. Many of these methods involve procedures to replace the letters of a message with other letters. The main mathematical tool used here is modular arithmetic. This is a generalization of addition on a clock (if it's 10 o'clock now, then in five hours it's 3 o'clock), and this turns out to be a very convenient language for cryptography. The final section on the Hill cipher requires some basic linear algebra, but this section may safely be skipped or assigned as optional reading.
- Chapter 3 describes one of the most important encryption methods ever, the Enigma. It was used by the Germans in World War II and thought by them to be unbreakable due to the enormous number of possibilities provided. Fortunately for the Allies, through espionage and small mistakes by some operators, the Enigma was successfully broken. The analysis of the Enigma is a great introduction to some of the basic combinatorial functions and problems. We use these to completely analyze the Enigma's complexity, and we end with a brief discussion of Ultra, the Allied program that broke the unbreakable code.
- Chapters 4 and 5 are devoted to attacks on the classical ciphers. The most powerful of these is frequency analysis. We further develop the theory of modular arithmetic, generalizing a bit more operations on a clock. We end with a discussion of one-time pads. When used correctly, these offer perfect security; however, they require the correspondents to meet and securely exchange a secret. Exchanging a secret via insecure channels is one of the central problems of the subject, and that is the topic of Chapters 7 and 8.
- In Chapter 6 we begin our study of modern encryption methods. Several mathematical tools are developed, in particular binary expansions (which are similar to the more familiar decimal or base 10 expansions) and recurrence relations (which you may know from the Fibonacci numbers, which satisfy the recursion $F_{n+2} = F_{n+1} + F_n$).

We encounter a problem that we'll face again and again in later chapters: an encryption method which seems hard to break is actually vulnerable to a clever attack. All is not lost, however, as the very fast methods of this chapter can be used in tandem with the more powerful methods we discuss later.

- Chapters 7 and 8 bring us to the theoretical and practical high point of the book, a complete description of RSA (its name comes from the initials of the three people who described it publicly for the first time—Rivest, Shamir, and Aldeman). For years this was one of the most used encryption schemes. It allows two people who have never met to communicate quickly and securely. Before describing RSA, we first discuss several simpler methods. We dwell in detail on why they seem secure but are, alas, vulnerable to simple attacks. In the course of our analysis we'll see some ideas on how to improve these methods, which leads us to RSA. The mathematical content of these chapters is higher than earlier in the book. We first introduce some basic graph theory and then two gems of mathematics, the Euclidean algorithm and fast exponentiation. Both of these methods allow us to solve problems far faster than brute force suggests is possible, and they are the reason that RSA can be done in a reasonable amount of time. Our final needed mathematical ingredient is Fermat's little Theorem. Though it's usually encountered in a group theory course (as a special case of Lagrange's theorem), it's possible to prove it directly and elementarily. Fermat's result allows the recipient to decrypt the message efficiently; without it, we would be left with just a method for encryption, which of course is useless. In addition to describing how RSA works and proving why it works, we also explore some of the implementation issues. These range from transmitting messages quickly to verifying the identity of the sender.
- In Chapter 9 we discuss the need to detect and correct errors. Often the data is not encrypted, and we are just concerned with ensuring that we've updated our records correctly or received the correct file. We motivate these problems through some entertaining riddles. After exploring some natural candidates for error detecting and correcting codes, we see some elegant alternatives that are able to transmit a lot of information with enough redundancy to catch many errors. The general theory involves advanced group theory and lattices, but fortunately we can go quite far using elementary counting.
- We describe some of the complexities of modern cryptography in Chapter 10, such as quantum cryptography and steganography.
- Chapter 11 is on primality testing and factorization algorithms. In the RSA chapters we see the benefits of the mathematicalization of messages. To implement RSA, we need to be able to find two large

primes; for RSA to be secure, it should be very hard for someone to factor a given number (even if they're told it's just the product of two primes). Thus, this advanced chapter is a companion to the RSA chapter, but is not needed to understand the implementation of RSA. The mathematical requirements of the chapter grow as we progress further; the first algorithms are elementary, while the last is the only known modern, provably fast way to determine whether a number is prime. As there are many primality tests and factorization algorithms, there should be a compelling reason behind what we include and what we omit, and there is. For centuries people had unsuccessfully searched for a provably fast primality test; the mathematics community was shocked when Agrawal, Kayal, and Saxena found just such an algorithm. Our goal is not to prove why their algorithm works, but instead to explain the ideas and notation so that the interested reader can pick up the paper and follow the proof, as well as to remind the reader that just because a problem seems hard or impossible does not mean that it is! As much of cryptography is built around the assumption of the difficulty of solving certain problems, this is a lesson worth learning well.

Chapters 1–5 and 10 can be covered as a one semester course in mathematics for liberal arts or criminal justice majors, with little or no mathematics background. If time permits, parts of Chapters 9 and 11 can be included or sections from the RSA chapters (Chapters 7 and 8). For a semester course for mathematics, science, or engineering majors, most of the chapters can be covered in a week or two, which allows a variety of options to supplement the core material from the first few chapters.

A natural choice is to build the semester with the intention of describing RSA in complete detail and then supplementing as time allows with topics from Chapters 9 and 11. Depending on the length of the semester, some of the classical ciphers can safely be omitted (such as the permutation and the Hill ciphers), which shortens several of the first few chapters and lessens the mathematical prerequisites. Other options are to skip either the Enigma/Ultra chapter (Chapter 3) or the symmetric encryption chapter (Chapter 6) to have more time for other topics. Chapters 1 and 10 are less mathematical. These are meant to provide a broad overview of the past, present, and future of the subject and are thus good chapters for all to read.

Cryptography is a wonderful subject with lots of great applications. It's a terrific way to motivate some great mathematics. We hope you enjoy the journey ahead, and we end with some advice:

- Wzr fdq nhhs d vhfuhw li rqh lv ghdg.
- Zh fdq idfwru wkh qxpehu iliwhhq zlwk txdqwxp frpsxwhuv.
  Zh fdq dovr idfwru wkh qxpehu iliwhhq zlwk d grj wudlqhg
  wr edun wkuhh wlphv.
- Jlyh xv wkh wrrov dqg zh zloo ilqlvk wkh mre.