Cryptography and the cyclic group $\mathbb{Z}_p^\times$
Rob Donnelly
Notes from the MSU course MAT 421 Intro to Algebraic Structures, Fall 2002

By now I think we have a fairly good understanding of the "modular arithmetic groups" $\mathbb{Z}_n$. Here are some things we know:

- Each $\mathbb{Z}_n$ is a (cyclic) group under addition. I'll use $\overline{a}$ to denote a typical element of $\mathbb{Z}_n$; this is the set of all integers equivalent to the integer a modulo n. The two equations $a \equiv b \mod n$ and $\overline{a} = \overline{b}$ in $\mathbb{Z}_n$ mean the same thing.
- Each $\mathbb{Z}_n$ has a multiplication operation, thus making it a ring.
- Define $\mathbb{Z}_n^\times$ to be the set of all elements in $\mathbb{Z}_n$ which are invertible under multiplication, i.e. all $\overline{a}$ for which there is an element $\overline{b}$ in $\mathbb{Z}_n$ such that $\overline{a}\overline{b} = \overline{1}$. It is not hard to prove that $\mathbb{Z}_n^\times$ is a group under multiplication.
- When $n = p$ is prime, then $\mathbb{Z}_p^\times$ consists of all non-zero elements of $\mathbb{Z}_p$. In particular, $\mathbb{Z}_p^\times$ is a (multiplicative) group of order $p - 1$, and $\mathbb{Z}_p$ is a field.

Most of this handout is aimed at understanding and applying the following theorem:

Theorem *Let p be a prime. Then $\mathbb{Z}_p^\times$ is a cyclic group of order $p - 1$.*

Guided Discovery Proof. Instead of offering an explicit proof, I'll give you a "guided discovery" proof in which you'll be asked to fill in details at each step.

1. Let F be any field, and let $f(x) = a_k x^k + a_{k-1}x^{k-1} + \cdots + a_1 x + a_0$ be a polynomial with coefficients a_i taken from F. Suppose that $f(x)$ has degree k (so $a_k \neq 0$). Prove that $f(x)$ has at most k distinct roots in F. HINT: Use induction on k and the fact that if $f(c) = 0$, then $x - c$ is a factor of $f(x)$.
2. Now let m be a positive integer and consider the polynomial $f(x) = x^m - \overline{1}$, where we think of the coefficients of f as elements of $\mathbb{Z}_p$. Prove that $f(x)$ has at most m distinct roots from $\mathbb{Z}_p^\times$.
3. Now use exercise #46 from §2.3 (p. 128) of Fraleigh's Abstract Algebra (6th ed.) to conclude that $\mathbb{Z}_p^\times$ is cyclic. □

AN IMPORTANT NOTE: It should be noted that this proof is "non-constructive" since it does not produce a generator for the group $\mathbb{Z}_p^\times$, but only proves the existence of such a generator. However, there are fairly fast algorithms for finding generators for $\mathbb{Z}_p^\times$.

Before proceeding, it might be worthwhile to try a couple of computational problems.

Exercise Find a generator for $\mathbb{Z}_{17}^\times$.

Exercise For the prime $p = 6133$, the number $b = 2507$ generates $\mathbb{Z}_{6133}^\times$. What is 2507^{855} in $\mathbb{Z}_{6133}$?

Application 1 *Diffie-Hellman key exchange*

So here's a problem: suppose Ann and Bob wish to communicate a secret message over an insecure line. They must assume that everything that they say can be overheard by the enemy. Even with the enemy listening in, is it possible for them to (1) agree on an encryption method and then (2) exchange encoded messages which only the two of them can decode?

One approach is for Ann and Bob to exchange a "key" which they can then use to encode and decode messages. This key might be a number which would specify a particular code to be used. So how can they exchange this key with the enemy listening in? The Diffie-Hellman answer is to perform certain computations in the group $\mathbb{Z}_p^\times$.

1. Ann and Bob publicly agree on a large prime p and a generator $\overline{b}$ for $\mathbb{Z}_p^\times$.
2. Ann privately picks an integer m between 0 and $p-1$ (inclusive) which she keeps to herself. Bob picks an integer n between 0 and $p-1$ (inclusive) which he keeps to himself.
3. Ann sends Bob the integer $\overline{b}^m$, and Bob sends Ann $\overline{b}^n$. They must assume the enemy can intercept this information.
4. Ann computes $(\overline{b}^n)^m$. Bob computes $(\overline{b}^m)^n$. Then $\overline{b}^{mn}$ is the key.

Exercise For the prime $p = 7919$, the number $b = 1003$ generates $\mathbb{Z}_{7919}^\times$. If Ann chooses $m = 20$ and Bob chooses $n = 32$, what is the Diffie-Hellman key?

Where is the security in this key exchange? The enemy knows the numbers $\overline{u} = \overline{b}^m$ and $\overline{v} = \overline{b}^n$. Clearly if enemy elements could solve these equations for m and n, then they would be able to determine the key. Here's a little more perspective on these two equations: we have an isomorphism of groups

$$\phi : \mathbb{Z}_{p-1} \longrightarrow \mathbb{Z}_p^\times \qquad \text{given by} \qquad \phi(\overline{x}) = \overline{b^x}.$$

Here, $\mathbb{Z}_{p-1}$ is a group under addition and $\mathbb{Z}_p^\times$ is a group under multiplication. *The reader should confirm that this "exponential function" is indeed an isomorphism!* Since this "exponential function" is a bijection, we can refer to its inverse as a "discrete logarithm." The inverse ϕ^{-1} is normally denoted mlog_b. That is,

$$\begin{aligned} \overline{u} = \overline{b}^m \quad &\Longleftrightarrow \quad \text{mlog}_b \overline{u} = m \\ \overline{v} = \overline{b}^n \quad &\Longleftrightarrow \quad \text{mlog}_b \overline{v} = n \end{aligned}$$

The difficulty is that the function ϕ is known to be a "one-way" function. That is, it is easy to compute in the forward direction, but is difficult to compute in the inverse direction. This is not uncommon: the squaring function $x \mapsto x^2$ is easier to calculate than its inverse $x \mapsto \sqrt{x}$. How difficult is it to calculate discrete logarithms? Of course, the difficulty is dependent on the size of the prime number p. The best algorithm right now requires

$$e^{\sqrt{\ln(p)\ln(\ln(p))}}$$

steps to compute $\text{mlog}_b \overline{y}$.

Exercise How long would it take to compute the discrete logarithm if we used a prime p with 100,000 digits and we used a computer which could compute 100,000,000 mips? (One "mip" is one million instructions per second. Assume each "step" in the algorithm for computing discrete logarithms corresponds to one computer "instruction.")

It is not known if it is possible to find a faster algorithm, although it is widely believed that the problem of computing discrete logarithms is "intractable."

A Historical Note: The Diffie-Hellman public key encryption scheme was invented in the mid-1970's. Diffie was a subversive 60's radical, and (along with Al Gore?) was an early user and proponent of the internet. He envisioned privacy of information exchange, free from the prying eyes of government.

Another Application of One-way Functions: Let me describe one other possible use of one-way functions, i.e. functions that are easy to compute in the forward direction but difficult to compute in the inverse direction. Suppose you have a computer network with access limited to "members only." Each member has a personal password which they enter in order to log in. It would not be wise for the network manager to store the passwords in a file because if a hacker found the password file, he would be able to have access to the network and to individual accounts. Instead, the network manager only keeps a file of "encoded" passwords, which can be made publicly available.

So how should passwords be encoded? At log-in, the computer can take the password and then apply a one-way function to "encode" the password. If the encoded version of the password appears in the list of encoded passwords, the user is allowed in. Now even if a hacker knows the encoded passwords, the one-way function prevents him from figuring out the original passwords, and hence keeps him out.

Application 2 *The RSA encryption scheme*

Suppose you are an internet vendor and you want to have a secure way for customers to exchange financial information with you. You could use a secure public key exchange method, but such key exchanges do involve some risk. So here's the idea: rather than sending out keys, you will send out "padlocks." You freely distribute padlocks to anyone, including potential enemies, because there is no risk in losing a padlock. The customer takes the padlock and uses it to secure the information he will send. Only you, the vendor, have the key to the padlock, so you are the only one who can unlock the customer's information. A mathematical method that accomplishes this was invented by MIT mathematicians Rivest, Shamir, and Adleman in 1977.

Vendor

1. For large primes p and q (100 digits or more), compute $n := pq$.
2. Set $m := LCM(p-1, q-1)$.
3. Pick $\overline{r} \in \mathbb{Z}_m^\times$, and let $\overline{s} := \overline{r}^{-1}$.
4. Publicly announce $\overline{r}$ and n.

Customer

1. Convert the message to be sent into a sequence of non-negative integers $M_1, M_2, \ldots, M_k$ each less than n such that M_i is relatively prime to n for each i. This is usually easy since n is very large and has only two prime factors.
2. Compute $\overline{R_i} := \overline{M_i}^r$ in $\mathbb{Z}_n$.
3. Send the numbers $R_1, R_2, \ldots, R_k$.

Vendor

1. Compute $\overline{R_i}^s$ in $\mathbb{Z}_n$. Remarkably, this will equal $\overline{M_i}$.
2. Assemble the M_i's to recover the message.

In the appendix we work through an example where we have $p = 37$, $q = 73$, and $r = 5$, and we use the RSA method to send the message "BAD." We use MAPLE to facilitate the computations.

Exercise With $p = 71$, $q = 131$, and $r = 43$, use the RSA method to send the message "WOW."

Of course, at this point you should be asking yourself two questions:

1. Why does the method work?
2. Why is it secure?

To answer the first question, we will prove that for the M_i's and R_i's described above, we have $\overline{R_i}^s = \overline{M_i}$ in $\mathbb{Z}_n$.

Theorem *Let p and q be prime, and let $m := LCM(p-1, q-1)$. Set $n := pq$. Suppose $\overline{rs} = \overline{1}$ in $\mathbb{Z}_m$, and pick M relatively prime to n. Then $\overline{M}^{rs} = \overline{M}$ in $\mathbb{Z}_n$.*

Guided Discovery Proof. Again, we'll prove this in steps.

1. Use the fact that M is relatively prime to p and q (why?) to see that $\overline{M}$ is in both $\mathbb{Z}_p^\times$ and $\mathbb{Z}_q^\times$. Now say how to use the fact that $\mathbb{Z}_p^\times$ and $\mathbb{Z}_q^\times$ are cyclic to obtain integers b_1, x_1, b_2, and x_2 such that

$$\begin{aligned} M &\equiv b_1^{x_1} \mod p \\ M &\equiv b_2^{x_2} \mod q \end{aligned}$$

2. Observe that $(p-1)|mx_1$ and $(q-1)|mx_2$. Now show that $\overline{M}^m = \overline{1}$ in $\mathbb{Z}_n$.
3. Use this to help conclude that $\overline{M}^{rs} = \overline{M}$ in $\mathbb{Z}_n$. □

To answer the second question, think about how an enemy might be able to take the "padlock" of RSA (namely, the numbers $\overline{r}$ and n) and determine the "key." If the enemy can factor n to determine p and q, then the enemy can easily find m and therefore the "key" $\overline{s}$. Where is the security? It turns out that multiplication is another example of a "one-way" function. That is, it

is much easier to multiply than it is to factor—at least, that is what we believe. Given the best algorithms for factoring, the time it would take to factor the large number n would be prohibitive. It is possible, however, that we just don't yet know how to think about factoring in the right way, and that someday it might be possible to factor a number just as easily as it is to multiply two numbers together.

Exercise Knowing that it is hard to factor a large number, and assuming that p and q are large primes, how do you think one should go about computing $m = LCM(p-1, q-1)$?

A Historical Note: It has recently been learned that Rivest, Shamir, and Adleman were not the first to discover the encryption technique that now bears their name. In the late 1960's, a man named James Ellis worked for the British Government Communications Headquarters (GCHQ), the British equivalent of the American National Security Agency (NSA), and began to think about this problem of "padlock" distribution. In 1973, British mathematicians Cox and Williamson came to GCHQ, and Ellis enlisted their help on the problem, which they quickly solved. However, their work was classified and remained classified for years, well after the RSA method was being widely used to secure internet communications. Ellis became gravely ill in 1997, and a move was made at GCHQ to declassify the work of Ellis, Cox, and Williamson, so that they could get the credit they deserved for their work on the solution to one of the most important problems of the Information Age. Sadly, Ellis died three weeks before any public announcement was made. More of the details of this story and of the history of cryptography can be found in Simon Singh's The Code Book.

A Note on Algorithms: The problem of understanding the "complexity" or "efficiency" of algorithms has been a subject of intense study since the 1960's and the advent of computers. A problem which can be solved with a "polynomial-time" algorithm is said to be in the class **P**. A problem is **NP** if you can check whether a proposed solution is actually a solution in polynomial time. Certainly the integer factoring problem is **NP**. It is known that $\mathbf{P} \subset \mathbf{NP}$, but it is not known whether $\mathbf{NP} \subset \mathbf{P}$. If so, then it follows that many seemingly "intractable" problems — like integer factoring — could then be resolved in polynomial time. *Thus, for example, RSA would no longer be secure.* In fact, in the 1998 article "Mathematical Problems for the Next Century" appearing in the *Math Intelligencer*, Steve Smale (a prominent name in the study of chaos and fractals) proposed that the $\mathbf{P} = \mathbf{NP}$ question is one of the seven most important mathematical problems of the 21st century. This is a hugely important question, because as I understand it, most secure internet communications use the RSA encryption method. For more information about algorithms and the **P** vs. **NP** problem, visit `www.claymath.org/prize_problems/p_vs_np.htm` and look for the expository paper "The **P** versus **NP** Problem" by Stephen Cook. By the way, Smale's article is a sequel of sorts. At the turn of the last century, the German mathematician David Hilbert famously proposed a list of 23 problems as among the most important problems for the 20th century, and these continue to spur many great and important advances in mathematics.