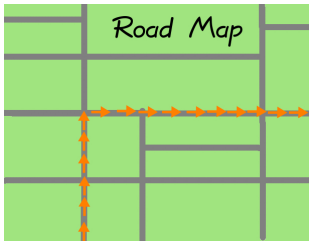
Module 10.2: The Basics of Modular Arithmetic

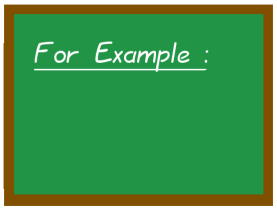


This module will introduce the bizarre but fascinating and powerful world of modular arithmetic. Exploring a new topic is often disorienting, and modular arithmetic will not be an exception. The key is to take the topic seriously, and engage it with the spirit of problem solving.

Warning: at first, modular arithmetic is going to look really, really stupid. However, modular arithmetic is spectacularly powerful. This is how cryptography works, this is how error-detecting codes work, and this is how a lot of the greatest and most fascinating theorems in number theory have been proven.

I have taught this course many times. Typically, there are a few students who discover that they already know the first quarter of this module, and therefore they conclude that they must already know everything that there is to know about modular arithmetic. Very often these students will ignore all the readings and the problems, get all the related questions wrong on the final exam, fail the entire semester, and repeat the course.

Like any other chapter, it is wise to read every word of every reading, and to solve every problem. However, the solving of the problems is absolutely indispensable. The one true test of whether or not you understand a concept in mathematics is whether or not you can solve a problem that has been designed to measure your understanding of that concept.



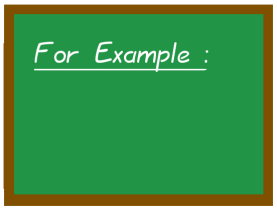
10-2-1

In our ordinary lives, we perform something called "clock arithmetic" all the time, without any stress or confusion. Here are two examples of this extremely straightforward process.

Imagine that you've got a date for a movie followed by lunch. Suppose the movie starts at 11 AM and it is two hours long. What time does it end? Of course, it ends at $11 + 2 = 13 = 1$ PM.

Suppose you're watching another movie with friends at home, and it starts at 11 PM but lasts 3 hours. What time does it end? Of course, it ends at $11 + 3 = 14 = 2$ AM.

The idea is that you can add or subtract 12 at any moment, in order to get back to a number in the expected set, which is $\{1, 2, 3, \ldots, 12\}$.



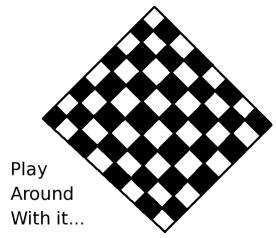
10-2-2

Here's an example of subtraction in ordinary clock arithmetic, using a 12-hour clock.

Suppose Helga has packed for a fun trip to see her family in Norway, and she is all ready to go the night before. She wakes up at 8 AM, has breakfast, and is bored. She knows that she wants to leave the house for the airport at 1 PM. She has a four-hour opera that she can watch on her laptop. When is the last safe time that Helga could start it?

Of course,

$$1 - 4 = -3 = -3 + 12 = 9 \text{ AM}$$

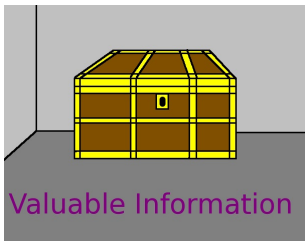


10-2-3

You've probably heard that European train stations and airports use a 24-hour clock, as does the US military. This is also a form of clock arithmetic. In this case, you can add or subtract 24 at any moment, in order to get your answer back into the expected set, which is $\{0, 1, 2, \ldots, 23\}$.

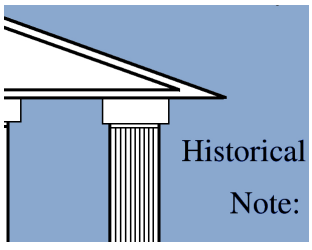
- One flight takes off at 22:00 and lasts 8 hours. When does it land (using the home timezone)?
- Another flight takes off at 19:00 and lasts 9 hours. When does it land (using the home timezone)?
- A third flight lands at 07:00 but lasts 11 hours. When did it take off (using the home timezone)?

The answers are to be found on Page 592 of this module.



- In ordinary clock arithmetic, we have the right to add or subtract 12 at any moment, to get the answer back into the expected set $\{1, 2, 3, \ldots, 12\}$.
- In 24-hour clock arithmetic, we have the right to add or subtract 24 at any moment, to get the answer back into the expected set $\{0, 1, 2, \ldots, 23\}$.
- When we talk about arithmetic in "the integers modulo 17," we have the right to add or subtract 17 at any moment, to get the answer back into the expected set $\{0, 1, 2, \ldots, 16\}$.
- When we talk about arithmetic in "the integers modulo 15," we have the right to add or subtract 15 at any moment, to get the answer back into the expected set $\{0, 1, 2, \ldots, 14\}$.
- When we talk about arithmetic in "the integers modulo m," we have the right to add or subtract m at any moment, to get the answer back into the expected set $\{0, 1, 2, \ldots, m-1\}$.

Some practice computations are important, so that we can be confident that we understand how this is supposed to work. By the way, rather than say "working in the integers *modulo* 17" it is very common to say "working *mod* 17." The number that we can add or subtract at any moment is called the *modulus*. The plural of modulus is *moduli*, from Latin.



Ordinary clock arithmetic uses 12 instead of 0, even in the 21st century, for historical reasons. This point will not concern us further. Nonetheless, a few of my readers might be curious about the historical origins of why 12 is used instead of 0.

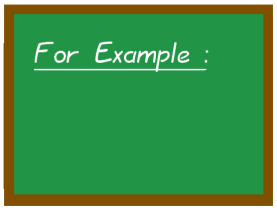
Clocks became popular across Europe during the 12th century, in the era before Roman numerals were replaced by Hindu-Arabic numerals. It was easy to write 12 o'clock as XII in Roman numerals, but very few people in the 12th century had seen 0 represented as a Roman numeral N, for the Latin *nulla*.

In fact, many textbooks from the 20th century erroneously claim that there was no way to write 0 in Roman numerals, even though there are manuscripts extant showing either N or the full word *nulla*.

For example, there is a data table from the year 725 CE, dealing with phases of the moon and the date of Easter, where all the numbers are written in Roman numerals, and zero is shown with N.

The table was written by Bede the Venerable (672–735 CE) or one of his colleagues, as part of a traditional discipline called *computus*. That discipline concerned the making of calendars that synchronized the lunar calendar, the solar calendar, and the civil calendar, for both religious reasons and practical agricultural reasons. It is from this word *computus* that the English language got the words "computation" and "computer." However, in my experience, almost no computer engineers or computer scientists know about the origin of the name of their profession.

Bede the Venerable made other contributions to scholarly topics, and therefore he is often listed as one of the most influential scholars of the period between the fall of the Western Roman Empire (in 476 CE), and the dawn of French culture with the rise of Charlemagne (748–814 CE), a period of over 300 years.



10-2-4

Suppose one sets as a task for you, the computation of $8(9)+11$ in the world of the integers mod 15, but also in the world of the integers mod 17.

Working modulo 15, we should have the right to add or subtract 15 at any moment. We would obtain the following:

$$8(9)+11 = 72+11 = \underbrace{83 \equiv 68 \equiv 53 \equiv 38 \equiv 23 \equiv 8}_{\text{repeatedly subtract } 15} \mod 15$$

where the symbol $\equiv$ is like an equal sign, but with a distinction that will be explained momentarily.

Working modulo 17, we should have the right to add or subtract 17 at any moment. We would obtain the following:

$$8(9)+11 = 72+11 = \underbrace{83 \equiv 66 \equiv 49 \equiv 32 \equiv 15}_{\text{repeatedly subtract } 17} \mod 17$$

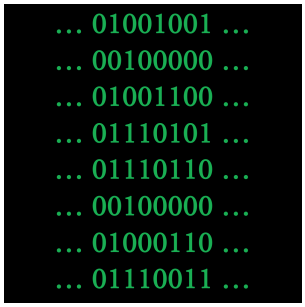
I often like to think of modular arithmetic to be analogous to visiting another planet in a science-fiction novel. On our planet $8(9)+11 = 83$, but on the planet "modulo 15," we have seen that $8(9)+11 \equiv 8$, and on the planet "modulo 17," we have seen that $8(9)+11 \equiv 15$.

Our planet has infinitely many integers. However, on the planet "modulo 15," the only numbers are $\{0, 1, 2, \ldots, 14\}$, and on the planet "modulo 17," the only numbers are $\{0, 1, 2, \ldots, 16\}$.

As you might have figured out, I switch from $=$ to $\equiv$ at the moment when the equality is no longer true in the ordinary integers.

There's actually a lot of variation among textbooks. The consensus is that $\equiv$ indicates that an equation is operating in the world of modular arithmetic, not the ordinary integers. The moment when one switches from $=$ to $\equiv$ varies from textbook to textbook. Some follow the convention that I follow, and others use $\equiv$ throughout any computation that takes place in modular arithmetic. Some textbooks even use $=$ for everything in modular arithmetic.

Of course, this should not concern you at all, unless you are thinking of writing your own textbook.



Notice that there is a difference between the syntax of computer languages such as Java and `C++`, and discrete mathematics. Almost all my readers will have seen the `%` operator from the computer languages Java, `C`, Python, or `C++`. If you haven't seen the `%` operator, then skip this box and the next box.

In discrete mathematics, we do indeed write

$$8(9) + 11 \equiv 8 \mod 15$$

as well as

$$8(9) + 11 \equiv 15 \mod 17$$

However, both of the following would evaluate to `false` in the computer languages Java, `C`, Python, and `C++`.

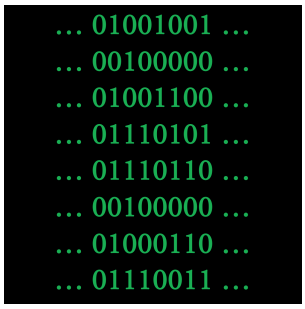
```
8*9 + 11 == 8 % 15
8*9 + 11 == 15 % 17
```

In reality, when a mathematician writes $x \equiv y \mod m$, this is equivalent to

```
(x % m) == (y % m)
```

in those computer languages.

In other words, the "mod" applies to the whole equation—to the entire line—not just to the number written nearest to "mod."



The previous box is extremely important. When students skip the first lecture on modular arithmetic, they are often baffled beyond description by the second and third lectures, because they incorrectly assume that the mathematical keyword "mod" operates like the `%` operator in the computer languages Java, `C`, Python, and `C++`.

I should add that this difference in notation is not a difference between two academic disciplines. When PhD students and faculty in computer science or computer engineering write theoretical papers about new algorithms, if modular arithmetic comes up, then they use the same notation as mathematicians do.

This shouldn't be too shocking. For example, computer scientists and computer engineers would write $2(3)x = 6x$ in an academic paper, just as mathematicians do. They would never write `2*3*x == 6*x` in such a context.

Let's return to the world of arithmetic mod 15. Of course, since I have the right to subtract or add 15 at any moment, then I should have the right to subtract or add any multiple of 15 as well, such as 30, 45, 60, 75, or any $n(15)$, so long as n is an integer.

With that in mind, instead of

$$8(9) + 11 = 72 + 11 = 83 \equiv 68 \equiv 53 \equiv 38 \equiv 23 \equiv 8$$

we should write instead

$$8(9) + 11 = 72 + 11 = 83 = 75 + 8 = 5(15) + 8 \equiv 8$$

When we move to larger moduli, this will be a much better way to write things, because it takes up less space. You can still see that I subtracted 15 a total of five times. Imagine if I had to subtract m several hundred times. It would be wasteful to list all the intermediate steps.

The space-saving and time-saving trick explained in the previous box is rather useful. It will be important when working with 4-digit numbers or larger—otherwise, we'd need huge pieces of paper.

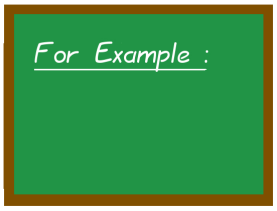
Similarly, in the world of arithmetic mod 17, instead of

$$8(9) + 11 = 72 + 11 = 83 \equiv 66 \equiv 49 \equiv 32 \equiv 15$$

we should write instead

$$8(9) + 11 = 72 + 11 = 83 = 68 + 15 = 4(17) + 15 \equiv 15$$

to save space. You can still see that I subtracted 17 a total of four times.



10-2-5

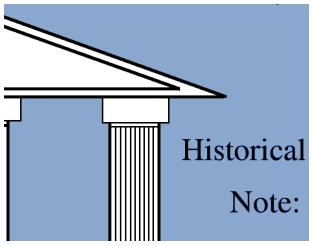
If you think of modular arithmetic like traveling to another planet, then you might imagine what the children on those planets have for their multiplication tables (or "times tables" as they were once called), and addition tables.

For working modulo five, we have the addition table:

+	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

For working modulo five, we have the multiplication table:

×	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1



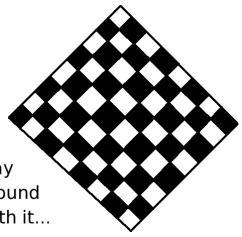
In previous generations, the professor for discrete mathematics would assign to the student the task of making tables for addition and multiplication, working modulo a handful of moduli. In the late 20th century, the professor might still have assigned that task, using MS-Excel or similar spreadsheet software.

These tables are called Cayley tables after Arthur Cayley (1821–1895), who invented several important bits of mathematics. He also popularized the "Theory of Groups," which is the gateway to a subject often called *Abstract Algebra* or *Modern Algebra*, but which really should be called *The Theory of Groups, Rings, and Fields*.

I regret that I do not have space for a biography of Arthur Cayley here.

To prevent a student uprising, I have provided you with some tables. They are for working mod 11, 13, 15, 17, 25, and 26. However, because of space restrictions, I provide only the multiplication table for mod 26, not the addition table. Those tables are located on `http://www.discrete-math-hub.com/textbook-in-progress.html` between Module 10.1 and Module 10.2.

Moreover, the provided Cayley tables aren't labelled. This is not an accident. You can tell from the fact that the columns are (for example) labelled 0, 1, 2, . . . , 16 that this table is for working mod 17. (That's because the only numbers in the world of arithmetic mod 17 are 0, 1, 2, . . . , 16.) It should also be very easy to see which are for addition, and which are for multiplication.



10-2-6

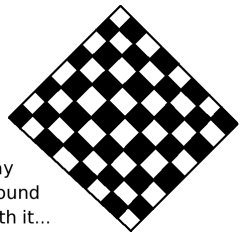
Moments ago, we computed that

$$8(9) + 11 \equiv 15 \mod 17$$

and also

$$8(9) + 11 \equiv 8 \mod 15$$

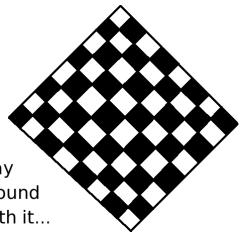
You should take a moment and confirm the computations above, but using the Cayley tables for mod 17 and for mod 15. This is a good way to get acquainted with using those tables.



10-2-7

Let's practice a bit before introducing new concepts. It might be fun to solve these with mental arithmetic, commit to what you think the answers are, and then compute it a second time using the Cayley tables. After that, you can check your answers, which are given on Page 592.

- What is $(8)(7) \mod 11$?
- What is $(9)(11) \mod 25$?
- What is $(14)(13) \mod 17$?
- What is $7(6) + 10 \mod 13$?
- What is $9(5) + 11 \mod 13$?



10-2-8

I know that some students will skip this box, but it really is a good idea to do a bit of drill when learning a new mathematical concept. It helps cement the knowledge in your brain.

Compute the answers to these questions using your pencil, and then compute them a second time using the Cayley tables. After that, you can check your answers, which are given on Page 592.

- What is $11(6) + 10 \mod 13$?
- What is $7(9) + 3 \mod 11$?
- What is $8(17) - 5 \mod 25$?
- What is $5(9) - 11 \mod 17$?

I should mention that the Cayley tables are like training wheels on a bicycle. They are temporary aids to assist the beginner.

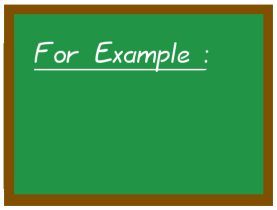
In cryptography, a modulus that is 100 digits long is definitely too small to be useful. The moduli that are used are always far larger than that. The Cayley tables for such moduli would require pieces of paper that are larger than our solar system.

Therefore, it is okay to use the Cayley tables for this module, and perhaps a bit in the next module. After that, they should be set aside and never used again.

Nonetheless, Cayley tables have some cool properties. In the addition tables, each symbol occurs exactly once in each row, and exactly once in each column. That's because the function $f(x) = x + 3$ is a bijection and that remains true if you replace 3 with any other member of the integers modulo m. Recall, a bijection merely "renames" the members of a set. This is called "the Latin Square" property or "the Magic Square" property, and was known to fascinate Benjamin Franklin (1706–1790), but in a slightly different context. You can read about that in the book *Benjamin Franklin's Numbers—An Unsung Mathematical Odyssey*, by Paul Pasles, published by Princeton University Press in 2008.

For the multiplication tables modulo a prime, if you remove the column of zeros, then the Latin Square property is there too. However, for composite moduli, this does not work. We do not have enough tools in our toolbox to explain why, just yet. It turns out this "Latin Square" property is extremely important in the "Theory of Groups."

Another neat property is that the tables are symmetric, both for addition and multiplication. If you've taken a course in matrix algebra or linear algebra, then you might know the definition of a symmetric matrix, which is a matrix that is equal to its own transpose. If you have not, then it just means that if you write the nth row as a sequence of symbols, and the nth column as a sequence of symbols, then you get exactly the same sequence.



10-2-9

One view of modular arithmetic is the clock arithmetic, given earlier. Another view comes from the Cayley tables. Still another view has to do with division and remainder.

When we first learn about division in elementary school, we learn that

$$22 \div 5 = 4R2$$

because $4(5) + 2 = 22$.

Similarly, we learn that

$$46 \div 7 = 6R4$$

because $6(7) + 4 = 46$.

We used this technique when we learned about converting between different number bases.

The following very simple idea comes up surprisingly often in number theory. It even has a name, either "the division algorithm" or "the division rule," depending on which textbook you read.

Informally, if you want to divide by an integer d but want to avoid using fractions, then you can take any integer z and rewrite it as $z = qd + r$, where q is the quotient, and r is the remainder. The remainder is always going to be in the range $0 \leq r < d$ and is an integer. While q can be any integer whatsoever, d cannot be zero.

Semi-formally, for any $z \in \mathbb{Z}$ and $d \in \mathbb{Z}$ with $d \neq 0$, we can write $z = qd + r$, where $q \in \mathbb{Z}$, $r \in \mathbb{Z}$, and $0 \leq r < d$.

Formally,

$$\forall z \in \mathbb{Z}, \forall d \in \mathbb{Z} - \{0\}, \exists q \in \mathbb{Z}, \exists r \in \mathbb{Z}, (z = qd + r) \wedge (0 \leq r < d)$$

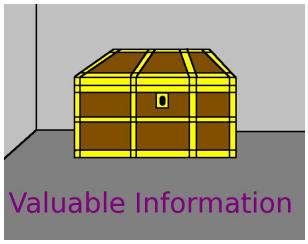
In practice, we do not normally consider $d < 2$.

Applying this idea to modular arithmetic, we can write the following in discrete mathematics.

- We write $52 \equiv 1 \mod 17$, because $52 = 3(17) + 1$.
- We write $54 \equiv 3 \mod 17$, because $54 = 3(17) + 3$.
- We write $78 \equiv 10 \mod 17$, because $78 = 4(17) + 10$.

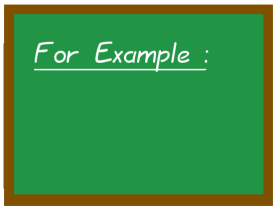
Whereas in elementary school, we would have written these lines:

- $52 \div 17 = 3R1$.
- $54 \div 17 = 3R3$.
- $78 \div 17 = 4R10$.



Let's summarize the last three boxes.

- We write $a \equiv b \mod m$ in modular arithmetic if and only if $a \div m = qRb$ for some integer q.
- We write $a \equiv b \mod m$ in modular arithmetic if and only if there exists an integer q such that $a = qm + b$.



10-2-10

Sometimes, it is useful to perform these computations on a hand calculator, especially with three-digit and four-digit moduli. Since the majority of hand calculators do not have a mod button, I should show you the standard procedure, which would even work on any four-function calculator. Let's confirm our findings that $78 \equiv 10 \mod 17$, and $46 \equiv 4 \mod 7$.

I press $78 \div 17$ on my hand calculator, and see

$$4.5882352\ldots$$

so I subtract 4 to remove the "integer part" q. Now, I see the "fractional part"

$$0.58823529\ldots$$

so I multiply by 17 to recover the remainder r. At this point, I see

$$10.0000000\ldots$$

therefore $78 \equiv 10 \mod 17$.

I press $46 \div 7$ on my hand calculator, and see

$$6.57142857\ldots$$

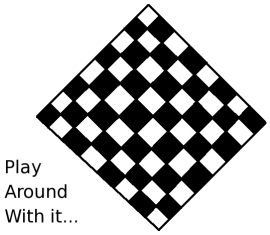
so I subtract 6 to remove the "integer part" q. Now, I see the "fractional part"

$$0.571428571\ldots$$

so I multiply by 7 to recover the remainder r. At this point, I see

$$4.00000000\ldots$$

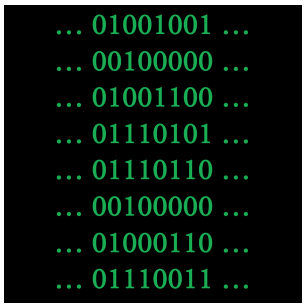
therefore $46 \equiv 4 \mod 7$.



10-2-11

Please take just a few seconds to confirm these facts using a hand-calculator:

- Confirm $52 \equiv 1 \mod 17$.
- Confirm $54 \equiv 3 \mod 17$.

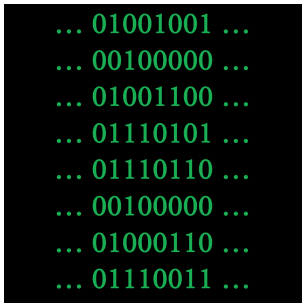


Suppose you have an internship with a company that hosts informational videos, or news articles. There are two major threats to the cybersecurity of such an enterprise.

The first is the "flash-crowd effect." If a celebrity dies, wins a major award, or says something embarrassing on nation-wide television, it is entirely possible that a URL might be sent out by someone with lots of followers to some fairly arbitrary old video or old article about that celebrity. Your company couldn't possibly predict in advance that the server hosting this file will be overwhelmed by millions more requests than it can handle.

The second is the "denial of service" attack. That's when millions or billions of superfluous requests bombard your company, so that the servers become overloaded, and cannot deliver the content actually requested by your paying customers. Government agencies suffer this attack very frequently. Medium-sized businesses also get targeted as a form of blackmail, to extract "protection fees."

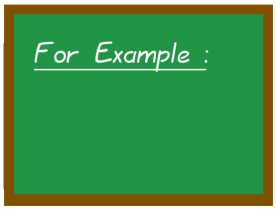
Modular arithmetic can help mitigate the damage from these attacks.



Suppose that your company has 285 servers. Simply take the serial number of each article or video, and compute a modular reduction by 285. The resulting number is the server for that article or video. (If the videos or articles are not serial numbered, then there's something called "a hash function" that can be used to do the same thing.)

In this manner, if a denial-of-service attack goes out with requests for 1 or 2 videos/articles, then those 1 or 2 servers will still get flooded. However, the remaining 284 or 283 servers will be totally unaffected, so that 99.6491% to 99.2982% of the servers are still up and running. It is bad luck for the articles/videos that have the misfortune to reside on the 1–2 servers that are targeted, but the vast majority of your company's content is still visible and completely unaffected.

Similarly, if a celebrity's old video or old article "goes viral," then the one server containing that one video or article will still get flooded. However, the remaining 284 servers will be totally unaffected, etc..., etc....



10-2-12

It turns out that we can also use linear functions in this world of modular arithmetic, so we'll start with

$$f(x) = 7x + 10$$

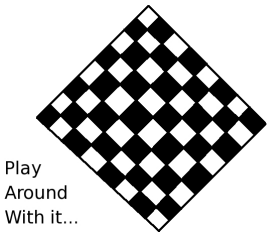
as our opening example. Suppose someone asked you what $f(6)$ is, working mod 17.

Similar to our previous computations, we can compute

$$f(6) = 7(6) + 10 = 52 = 51 + 1 = 3(17) + 1 \equiv 1$$

Therefore, we write that

$$f(6) \equiv 1 \mod 17$$



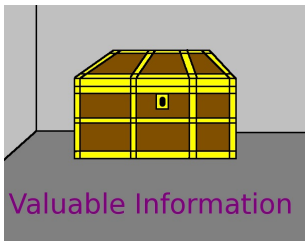
10-2-13

Now suppose that

$$g(x) = 9x + 11$$

- Working mod 17, what is $g(5)$?
- Working mod 17, what is $g(9)$?
- Working mod 15, what is $g(5)$?
- Working mod 15, what is $g(9)$?

The answers will be given on Page 593.

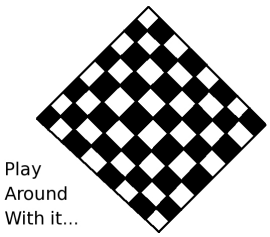


Another view of modular arithmetic is that $a \equiv b \mod m$ if and only if $a - b \in \text{multiples}(m)$.

This leads to an extremely rapid and reliable way of checking one's work, after computing a modular reduction. For example, let's check the first and last subproblems from the previous box.

- We computed $g(5) \equiv 5 \mod 17$. We know $g(5) = 9(5) + 11 = 56$ in the ordinary integers. To confirm that $56 \equiv 5 \mod 17$, we need to ask if $56 - 5$ is a multiple of 17 or not. Therefore, we compute $56 - 5 = 51$ and divide by 17 to get 3. Since 3 is an integer, we know 51 is a multiple of 17, and our modular reduction was correct.
- We computed $g(9) \equiv 2 \mod 15$. We know $g(9) = 9(9) + 11 = 92$ in the ordinary integers. To confirm that $92 \equiv 2 \mod 15$, we need to ask if $92 - 2$ is a multiple of 15 or not. Therefore, we compute $92 - 2 = 90$ and divide by 15 to get 6. Since 6 is an integer, we know 90 is a multiple of 15, and our modular reduction was correct.

This method of checking is so rapid that it should always be used.

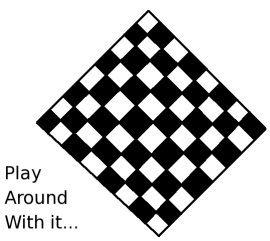


10-2-14

Some students might like further practice with modular arithmetic before proceeding. Others might choose to skip this box, if they feel confident already.

- Evaluate $a(x) \equiv 9x + 4 \mod 17$ at $x = 11$.
- Evaluate $b(x) \equiv 2x + 5 \mod 11$ at $x = 4$.
- Evaluate $c(x) \equiv 17x + 19 \mod 25$ at $x = 5$.

The answers will be given on Page 593.



10-2-15

I'd like everyone to solve these questions. Suppose that $h(x) = 11x + 12$.

- What is $h(4) \mod 17$?
- What is $h(6) \mod 17$?
- What is $h(11) \mod 17$?
- What is $h(4) \mod 11$?
- What is $h(6) \mod 11$?
- What is $h(11) \mod 11$?

There's a special property of $h(x) \mod 11$, so the answers will be given in the next box.

Here are the answers to the questions that were asked in the previous box.

- What is $h(4)$ mod 17? [Answer: $h(4) \equiv 5 \mod 17$.]
- What is $h(6)$ mod 17? [Answer: $h(6) \equiv 10 \mod 17$.]
- What is $h(11)$ mod 17? [Answer: $h(11) \equiv 14 \mod 17$.]
- What is $h(4)$ mod 11? [Answer: $h(4) \equiv 1 \mod 11$.]
- What is $h(6)$ mod 11? [Answer: $h(6) \equiv 1 \mod 11$.]
- What is $h(11)$ mod 11? [Answer: $h(11) \equiv 1 \mod 11$.]

Modular arithmetic is full of surprises! If you put all the integers from -50 to 50 into $h(x)$, working modulo 11, the answer will always be 1. The Sage code below this box is a great way to test this claim. Actually, you could easy check all the integers from -200 to 200, or any other interval, and you would see that working modulo 11, the answer will always be 1.

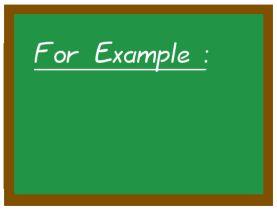
The key issue is that $11 \equiv 0 \mod 11$, and $12 \equiv 1 \mod 11$. Therefore

$$h(x) = 11x + 12 \equiv 0x + 1 \equiv 1 \mod 11$$

which means that, in the world of arithmetic modulo 11, $h(x) \equiv 1$.

Below is some Sage code for checking this fact. If you cut-and-paste this code, then take care to adjust the indentation. That's because Sage is built on top of Python, and Python is sensitive about indentation.

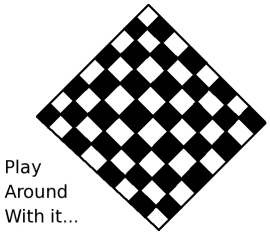
```
for x in range(-50, 51):
    y = (11*x + 12) % 11
    print("f(", x, ") =", y)
```



10-2-16

As it turns out, exponentiation in modular arithmetic will be fundamentally crucial in our study of modern cryptography, We will start with a simple example. Let's consider the powers of 2 mod 15. This means successively multiplying by 2. To be precise, $2^{n+1} = 2(2^n)$ for all positive integers n.

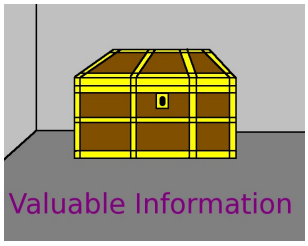
$$\begin{aligned}
2^2 = 2(2) &= 4 \\
2^3 = 2(4) &= 8 \\
2^4 = 2(8) &= 16 = 15 + 1 \equiv 1 \\
2^5 \equiv 2(1) &= 2 \\
2^6 \equiv 2(2) &= 4 \\
2^7 \equiv 2(4) &= 8 \\
2^8 \equiv 2(8) &= 16 = 15 + 1 \equiv 1 \\
2^9 \equiv 2(1) &= 2 \\
2^{10} \equiv 2(2) &= 4 \\
2^{11} \equiv 2(4) &= 8 \\
2^{12} \equiv 2(8) &= 16 = 15 + 1 \equiv 1
\end{aligned}$$



10-2-17

Now compute the powers of 3, working modulo 15, from 3^2 to 3^{12}. Follow the pattern of the previous box.

The answer will be given on Page 593.



Over the past sixty years, the toy cipher called "The Affine Cipher," has been extremely popular in teaching the rudiments of cryptography. It's also a great way to explore modular arithmetic.

Before we begin, I'd like to emphasize that the affine cipher is a toy cipher meant to train us in cryptography. It is less secure than the secret-decoder ring in a cracker-jack box. Never use the affine cipher in practice, in the real world!

- The plaintext is converted into a sequence of integers mod 26, using the table found below.
- The sequence of numbers is encrypted using a simple function, such as

$$c = f(p) = 17p + 19 \mod 26$$

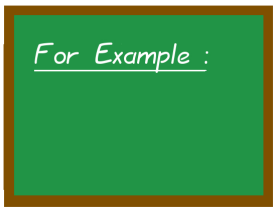
  where p indicates the plaintext and c indicates the ciphertext.
- The new sequence of numbers is converted from numbers back into letters, using the same table.

Of course, there are many possible functions of the form $c = f(p) = mp + b \mod 26$. In our opening example, we've chosen one possible function, where $m = 17$ and $b = 19$. There is also a "mod 27" version of the Affine cipher, which we will not explore.

The following is a bijection between the letters of the English alphabet and the integers modulo 26.

$A \leftrightarrow 0$	$B \leftrightarrow 1$	$C \leftrightarrow 2$	$D \leftrightarrow 3$
$E \leftrightarrow 4$	$F \leftrightarrow 5$	$G \leftrightarrow 6$	$H \leftrightarrow 7$
$I \leftrightarrow 8$	$J \leftrightarrow 9$	$K \leftrightarrow 10$	$L \leftrightarrow 11$
$M \leftrightarrow 12$	$N \leftrightarrow 13$	$O \leftrightarrow 14$	$P \leftrightarrow 15$
$Q \leftrightarrow 16$	$R \leftrightarrow 17$	$S \leftrightarrow 18$	$T \leftrightarrow 19$
$U \leftrightarrow 20$	$V \leftrightarrow 21$	$W \leftrightarrow 22$	$X \leftrightarrow 23$
	$Y \leftrightarrow 24$	$Z \leftrightarrow 25$	

The best way to explain how the affine cipher works is through some examples, and we'll look at those momentarily.



10-2-18

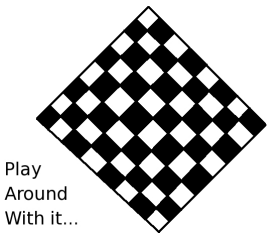
Using

$$f(p) = 17p + 19 \mod 26$$

as the encryption function for the affine cipher, encrypt the message "FROG."

- Step One: (F, R, O, G) becomes (5, 17, 14, 6) using the table in the previous box.
- Step Two: $(5, 17, 14, 6)$ encrypts to $(f(5), f(17), f(14), f(6)) \equiv (0, 22, 23, 17)$
- Step Three: $(0, 22, 23, 17)$, becomes (A, W, X, R) or "AWXR."

Therefore, the plaintext message "FROG" becomes the ciphertext "AWXR."



10-2-19

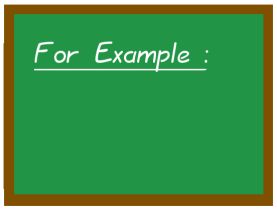
Let's continue to use

$$f(p) = 17p + 19 \mod 26$$

as the encryption function for the affine cipher.

- Encrypt the message "BEER."
- Encrypt the message "STATS."

The answers (including the intermediate steps) will be given on Page 594 of this module.



10-2-20

Of course, to be a usable cryptosystem, there must be some way of decrypting the encrypted messages. As it turns out,

$$p = g(c) = 23c + 5 \mod 26$$

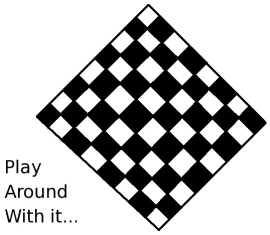
is the decryption function for $f(p) = 17p + 19 \mod 26$, but we're not yet entirely ready to compute the decryption function from the encryption function. Nonetheless, we can practice using the decryption function.

For example, to decrypt AWXR, (A, W, X, R) becomes (0, 22, 23, 17) and $(g(0), g(22), g(23), g(17)) \equiv (5, 17, 14, 6)$, which is (F, R, O, G) or "FROG."

Similarly, to decrypt KJJW, (K, J, J, W) becomes (10, 9, 9, 22) and $(g(10), g(9), g(9), g(22)) \equiv (1, 4, 4, 17)$, which is (B, E, E, R) or "BEER."

Once we learn a bit about modular inverses, we can easily compute the affine decryption function for any affine encryption function that we are shown, assuming that a decryption function exists at all. We'll do that in Module 10-4, as an exercise.

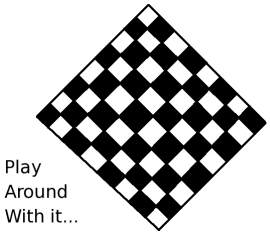
We will also be able to determine, in advance, what encryption functions have a decryption function.



10-2-21

Using the decryption function $p = g(c) = 21c + 8 \mod 26$, how would you decrypt "BEQ"?

The answer (including the intermediate steps) will be given on Page 594.

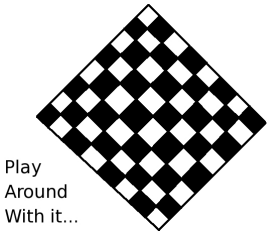


10-2-22

Using the decryption function $c = g(p) = 19p + 17 \mod 26\ldots$

- ... decrypt the ciphertext "XTIWVIV."
- ... decrypt the ciphertext "VABVILVL."

The answers will be given on Page 594.

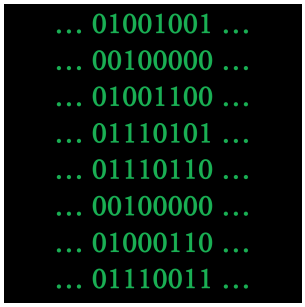


10-2-23

Let's return now to encryption.

- Using $c = f(p) = 13p + 3 \mod 26$, encrypt "BEERS."
- Using $c = f(p) = 13p + 3 \mod 26$, encrypt "BOATS."
- Based on your answer to the first two subproblems, do you think $f(p) = 13p + 3 \mod 26$ is a good encryption function?
- Based on your answer to the first two subproblems, do you think $f(p) = 13p + 3 \mod 26$ is injective?

The answers (with intermediate steps) will be given on Page 594.

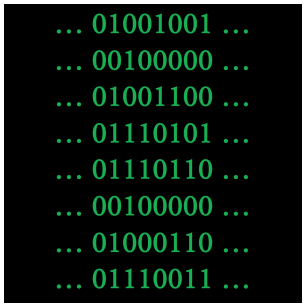


What we've just stumbled across is called "unique decodability."

For any cryptosystem, it is important that the encryption function be injective. If not, then there would be two plaintext messages, $m_1 \neq m_2$, such that they encrypt to the same ciphertext. When the receiver gets this ciphertext, there would be no way of determining if m_1 or m_2 was the intended message. We just saw that $f(p) = 13p + 3$ is not uniquely decodable.

For uniquely decodable encryption functions, another property is called "correctness." We say that an encryption function is correct if and only if, for all possible messages m, it is the case that m equals the decryption of the encryption of m.

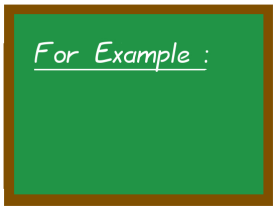
When new cryptosystems are proposed, they must be proven to be uniquely decodable, and correct, among many other important properties.



Another twist in cryptography is that one must keep track of what is public and what is private. The classification of the variables into public and private is not usually found in other branches of mathematics or computer science, so this presents students with a bit of a surprise.

For example, it's clear that the choice of m and b in the affine cipher must remain private. Suppose Dave is using the affine cipher, and Dave's choices of m and b became public. With only 26 function evaluations, an adversary could encrypt the entire alphabet. Then, this data could serve as a dictionary, enabling extremely rapid decryption of any encrypted message that the adversary might intercept.

In fact, just finding the 20 most common letters, and skipping the 6 rare letters: V, K, X, J, Q, and Z, would be plenty. These six rare letters have probability 0.99%, 0.67%, 0.19%, 0.16%, 0.11%, and 0.09%, respectively, totaling only 2.21% of modern English prose. These 20 function evaluations, to discover the ciphertexts for the 20 non-rare letters, would require much less work than checking all $(26)^2 = 676$ possible choices of m and b.



10-2-24

Modern cryptanalysis, however, is a much more subtle art than most people imagine. Suppose that Alice makes a physical device to carry out the affine cipher, using

$$c = f(p) = 15p + 9 \mod 26$$

Of course, she has to keep the function private, for reasons that were explained in the previous box. She decides to give a demo of her device. Bob asks her to encrypt "ABC." The ciphertext is "JYN." Who would guess that knowing the ciphertext for "ABC" will allow Alice's choices for m and b to become known?!

- Bob certainly knows that "A" encrypts to "J." This means that he knows 0 encrypts to 9, or $f(0) = 9$. If we think of $c = f(p) = mp + b$, then Bob now knows that $b = 9$, since
$$9 = f(0) = m0 + b = 0 + b = b$$
- Next, Bob knows that "B" encrypts to "Y." This means that he knows 1 encrypts to 24, or $f(1) = 24$. If we think of $f(1) = m1 + b = m + b$, Bob now knows that $24 = m + b$.
- However, because Bob also knows $b = 9$, then Bob knows $m = 24 - b = 24 - 9 = 15$. Now Bob knows the encryption function, $c = f(p) = mp + b = 15p + 9$.
- Bob can also use the last letter to check his work. Since "C" is 2, if he is correct, he expects
$$f(2) = 15(2) + 9 = 30 + 9 = 39 = 26 + 13 \equiv 13$$
and indeed, 13 is "N." Now Bob is certain that $c = f(p) = 15p + 9$ is Alice's encryption function.

Our exploration of cryptography will reveal many, many more surprises.

This module is now complete. All that follows will be the solutions to the checkerboard boxes from earlier in this module.

Here are the answers to the clock arithmetic questions from Page 578.

- One flight takes off at 22:00 and lasts 8 hours. When does it land (using the home timezone)?
  [Answer: at 06:00, because $22+8 = 30$, which is out of the expected set, but $30-24 = 6$.]
- Another flight takes off at 19:00 and lasts 9 hours. When does it land (using the home timezone)?
  [Answer: at 04:00, because $19+9 = 28$, which is out of the expected set, but $28-24 = 4$.]
- A third flight lands at 07:00 but lasts 11 hours. When did it take off (using the home timezone)?
  [Answer: at 20:00, because $7 - 11 = -4$, which is out of the expected set, but $-4 + 24 = 20$.]

The point is that I have the right to add or subtract the 24 at any moment, to get my answer back into the expected set $\{0, 1, 2, 3, \ldots, 23\}$.

Here are the answers to the computations (from Page 582) that I asked you to perform using both mental arithmetic and the Cayley tables.

- What is $(8)(7) \bmod 11$?
  [Answer: 1, because $(8)(7) = 56 = 55 + 1 = 5(11) + 1 \equiv 1 \mod 11$.]
- What is $(9)(11) \bmod 25$?
  [Answer: 24, because $(9)(11) = 99 = 75 + 24 = 3(25) + 24 \equiv 24 \mod 25$.]
- What is $(14)(13) \bmod 17$?
  [Answer: 12, because $(14)(13) = 182 = 170 + 12 = 10(17) + 12 \equiv 12 \mod 17$.]
- What is $7(6) + 10 \bmod 13$?
  [Answer: 0, because $7(6) + 10 = 42 + 10 = 52 = 4(13) + 0 \equiv 0 \mod 13$.]
- What is $9(5) + 11 \bmod 13$?
  [Answer: 4, because $9(5) + 11 = 45 + 11 = 56 = 52 + 4 = 4(13) + 4 \equiv 4 \mod 13$.]

Here are the answers to the computations (from Page 582) that I asked you to perform using both mental arithmetic and the Cayley tables.

- What is $11(6) + 10 \bmod 13$?
  [Answer: 11, because $11(6) + 10 = 66 + 10 = 76 = 65 + 11 = 5(13) + 11 \equiv 11 \mod 13$.]
- What is $7(9) + 3 \bmod 11$?
  [Answer: 0, because $7(9) + 3 = 63 + 3 = 66 = 66(11) \equiv 0 \mod 11$.]
- What is $8(17) - 5 \bmod 25$?
  [Answer: 6, because $8(17) - 5 = 136 - 5 = 131 = 125 + 6 = 5(25) + 6 \equiv 6 \mod 25$.]
- What is $5(9) - 11 \bmod 17$?
  [Answer: 0, because $5(9) - 11 = 45 - 11 = 34 = 2(17) + 0 \equiv 0 \mod 17$.]

Here is the answer to the question from Page 586, where we were asked to evaluate the function

$$g(x) = 9x + 11$$

working mod 17 and working mod 15.

- Working mod 17, what is $g(5)$? [Answer: $g(5) \equiv 5 \mod 17$.]
- Working mod 17, what is $g(9)$? [Answer: $g(9) \equiv 7 \mod 17$.]
- Working mod 15, what is $g(5)$? [Answer: $g(5) \equiv 11 \mod 15$.]
- Working mod 15, what is $g(9)$? [Answer: $g(9) \equiv 2 \mod 15$.]

In general, knowing the value of a function mod x tells you very little about the value of a function mod y, unless some special coincidence occurs. For example, if y is a divisor of x, or if y is a multiple of x. Those special coincidences can be fairly neat, but regrettably, we don't have the space or the time to explore those special cases.

Here are the answers to the three optional practice questions from Page 586 about evaluating functions using modular arithmetic.

- Evaluate $a(x) \equiv 9x + 4 \mod 17$ at $x = 11$.
  [Answer: $a(11) = 9(11) + 4 = 99 + 4 = 103 = 102 + 1 = 6(17) + 1 \equiv 1 \mod 17$.]
- Evaluate $b(x) \equiv 2x + 5 \mod 11$ at $x = 4$.
  [Answer: $b(4) = 2(4) + 5 = 8 + 5 = 13 = 11 + 2 \equiv 2 \mod 11$.]
- Evaluate $c(x) \equiv 17x + 19 \mod 25$ at $x = 5$.
  [Answer: $c(5) = 17(5) + 19 = 85 + 19 = 104 = 100 + 4 = 4(25) + 4 \equiv 4 \mod 25$.]

Here is the answer to the question from Page 588, where we were asked to compute the powers of 3 mod 15. Remember, each new power is 3 times the answer of the previous line.

$$\begin{aligned}
3^2 = 3(3) &= 9 \\
3^3 = 3(9) &= 27 = 15 + 12 \equiv 12 \\
3^4 = 3(12) &= 36 = 30 + 6 = 2(15) + 6 \equiv 6 \\
3^5 \equiv 3(6) &= 18 = 15 + 3 \equiv 3 \\
3^6 \equiv 3(3) &= 9 \\
3^7 \equiv 3(9) &= 27 = 15 + 12 \equiv 12 \\
3^8 \equiv 3(12) &= 36 = 30 + 6 = 2(15) + 6 \equiv 6 \\
3^9 \equiv 3(6) &= 18 = 15 + 3 \equiv 3 \\
3^{10} \equiv 3(3) &= 9 \\
3^{11} \equiv 3(9) &= 27 = 15 + 12 \equiv 12 \\
3^{12} \equiv 3(12) &= 36 = 30 + 6 = 2(15) + 6 \equiv 6 \\
\vdots \quad \vdots &\quad \vdots
\end{aligned}$$

Here are the answers to the question, from Page 589, where we were asked to encrypt two words using the affine cipher.

- Encrypt the message "BEER." [Answer: "KJJW."]
- Encrypt the message "STATS." [Answer: "EPOPE."]

For the intermediate steps, note that (B, E, E, R) becomes (1, 4, 4, 17) and encrypts to $(f(1), f(4), f(4), f(17)) \equiv (10, 9, 9, 22)$, which is (K, J, J, W) or "KJJW." Similarly, note that (S,T , A,T , S) becomes (18, 19, 0, 19, 18) and encrypts to $(f(18), f(19), f(0), f(19), f(18)) = (13, 4, 19, 4, 13)$ which is (N, E, T, E, N) or "NETEN."

On Page 590, we were asked to use the decryption function $p = g(c) = 21c + 8 \mod 26$, to decrypt "BEQ."

[Answer: (B,E,Q) becomes (1, 4, 16) and decrypts to $(g(1), g(4), g(16)) = (3, 14, 6)$ which is (D, O, G) or "DOG."]

On Page 590, you were asked to decrypt two ciphertexts using the decryption function $c = g(p) = 19p + 17 \mod 26$.

- For the ciphertext "XTIWVIV," the plaintext is "MONTANA."
- For the ciphertext "VABVILVL," the plaintext is "ARKANSAS."

Here are the answers to the question from Page 590, where we encrypted using $c = f(p) = 13p + 3 \mod 26$.

- Using $c = f(p) = 13p + 3 \mod 26$, encrypt "BEERS." [Answer: (B, E, E, R, S) becomes (1, 4, 4, 17, 18) and $(f(1), f(4), f(4), f(17), f(18)) \equiv (16, 3, 3, 16, 3)$, which is (Q, D, D, Q, D) or "QDDQD."]
- Using $c = f(p) = 13p + 3 \mod 26$, encrypt "BOATS." [Answer: (B, O, A, T, S) becomes (1, 14, 0, 19, 18) and $(f(1), f(14), f(0), f(19), f(18)) \equiv (16, 3, 3, 16, 3)$, which is (Q, D, D, Q, D) or "QDDQD."]
- Based on your answer to the first two subproblems, do you think $f(p) = 13p + 3 \mod 26$ is a good encryption function? [Answer: Hell no. If a military unit is in urgent need of boats, and they get beers instead, then that's a major problem!]
- Based on your answer to the first two subproblems, do you think $f(p) = 13p + 3 \mod 26$ is injective? [Answer: Clearly not. We had several cases of differing inputs getting the same output. For example, $f(4) \equiv 3 \equiv f(18)$ and $f(1) \equiv 16 \equiv f(17)$.]