Math 187A Notes

Introduction to Cryptography

Winter 2023

Taught by Professor Shishir (Sunny) Agrawal

Table of Contents

1 Introduction to Cryptography

We begin with some common definitions.

1.1 Terminology

Definition 1.1: Cipher

A **cipher**, or cryptosystem, is a cryptographic method for confidential communication.

Generally, a cryptographic method includes algorithms for *encryption* and *decryption*, which are inverse processes that convert between plainly readable information called *plaintext*[1] and unintelligible information called *ciphertext*.

Definition 1.2: Sender

A **sender**, often named "Alice" in abstract cryptographic discussions, *encrypts* her plaintext into ciphertext.

Definition 1.3: Receiver

A **receiver**, often named "Bob," *decrypts* (or deciphers) the ciphertext back into plaintext.

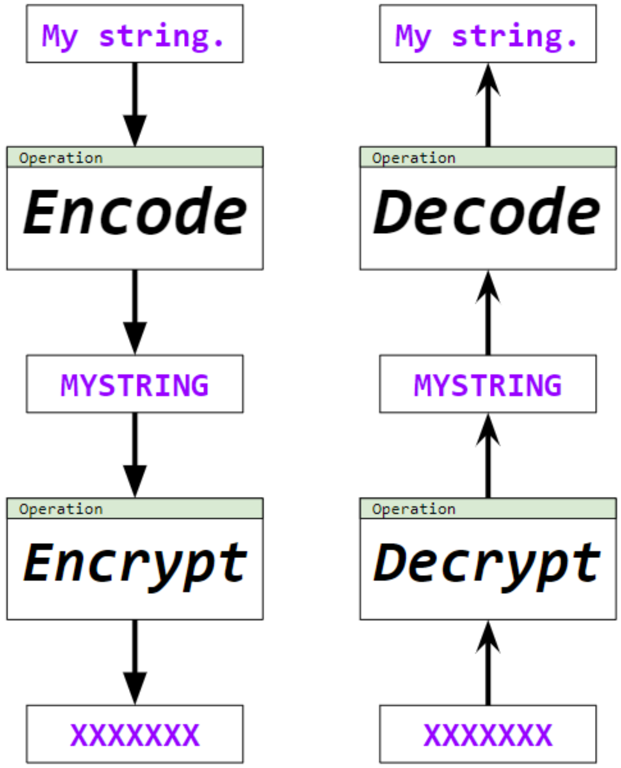
Often times, Bob will use a *key* to decrypt the message. This is sometimes known as a private key or decryption key.

Definition 1.4: Encoding

The (usually) preliminary step where a message is converted into a format which can then be encrypted is called **encoding**.

Note that encoded text is not secure; it is only secure after encryption. So, we can think of encoding as the pre-processing step. In other words, before we encrypt something, we might *encode* the text so it's easier to encrypt. It should also be noted that if a message had to be encoded before encryption, then it will also need to be decoded after decryption.

[1] In cryptography, we use *plaintext* and *ciphertext* instead of *plain text* and *cipher text*.



Definition 1.5: Adversary

An **adversary**, often named "Eve," is one whose aim is to prevent the users of a cryptosystem from achieving their goal.

In our case here, an adversary can intercept a ciphertext. Thus, the adversary will not have Bob's decryption key at the beginning. The idea is that, even if the adversary knows what cryptosystem was used to encrypt the message, if the adversary doesn't have this decryption key, she should ideally not be able to decrypt the message. If she does manage to figure out the plaintext, she has *broken* the code.

Definition 1.6: Attack Model

An **attack model** specifies what Eve is allowed to do in order to break the code.

Some common attack models includes:

- Ciphertext-only attack: Eve must recover the plaintext using only the ciphertext.
- Known-plaintext attack: Eve may have access to some information about the plaintext (e.g., knowledge of portions of the plaintext), which can be used to recover the plaintext entirely.
- Chosen-plaintext attack: Eve can request or generate ciphertexts corresponding to any plaintext message of her choosing, and she can use this information to recover the plaintext.

Classical cryptography was mostly concerned with assuring security against the first two. Modern cryptography tries to assure security against the last.

2 Classical Cryptosystems

We begin with a definition:

Definition 2.1: n-gram

An n-gram is a sequence of n letters.

For example, a 1-gram is just a single letter; a 2-gram (i.e., *bigram*) is a pair of letters; and so on. Generally, we can group many classical cryptosystems into a few different encryption strategies.

<table>
<tr><th>Strategy</th><th>Description</th></tr>
<tr><td>Transposition</td><td>Involves rearranging units of plaintext according to some pattern. We'll see just one example of this type of cipher: rectangular transposition.</td></tr>
<tr><td>Substitution</td><td>Involves replacing units of plaintext with units of ciphertext. We can further group substitution ciphers into some subtypes:
<table>
<tr><th>Subtype</th><th>Description</th></tr>
<tr><td>Simple Substitution</td><td>In these ciphers, single letters of plaintext are replaced by ciphertext. The substitution scheme stays the same over the course of the entire message. Some examples we'll see include:
• Masonic cipher
• Caesar cipher
• Affine cipher
• Polybius square
In essence, though, there is a 1-1 relationship between the letters of the plaintext and the ciphertext alphabets.</td></tr>
<tr><td>Polygraphic Substitution</td><td>In these ciphers, groups of letters in the plaintext are replaced by ciphertext (a group of n letters is called an n-gram).The substitution scheme stays the same over the entire message. Some examples we'll see include:
• Hill cipher
• Playfair cipher
So, in essence, polygraphic substitution is just simple substitution but with groups of letters instead of individual letters.</td></tr>
<tr><td>Polyalphabetic Substitution</td><td>In these ciphers, single letters in the plaintext are replaced by ciphertext, and the substitution scheme changes over the course of the message. Some examples include:
• Vignere cipher
• One-time pad</td></tr>
</table>
In practice, however, most cryptosystems employ a combination of these strategies.</td></tr>
</table>

2.1 Rectangular Tranposition

Rectangular tranposition, known also as *regular columnar transposition*, is a tranposition cipher. The ciphertext is obtained by *permuting* the letters of the plaintext in a particular pattern. The pattern is determined by a secret *keyword*.

Roughly speaking, the steps to perform rectangular transposition are as follows:

1. Using the keyword, rank the letters based on alphabetical ranking.
2. Break up the message into groups of n, where n is the length of the keyword.
3. For each group, do the following:
   - Encrypting: If the ith letter of the keyword has rank j, move the ith letter in the group into the jth position.
   - Decrypting: If the ith letter of the keyword has rank j, move the jth letter of each group into the ith position.

Note that keywords with repeat letters do not work by themselves. We either need to agree not to use words with repeat letters, or remove duplicate letters from the keyword[2].

(Example: Encryption.) Suppose that Alice and Bob share the keyword `GUARD`, and that Alice wants to send the following message to Bob:

```
Hide! The baboons are coming for you.
```

First, we'll **encode** the message so that it's easier to encrypt. In our example, we'll remove all spaces and punctuation.

```
HIDETHEBABOONSARECOMINGFORYOU
```

Now that encoding is done, we still need to encrypt the message. Notice how the keyword `GUARD` has 5 letters; we can break the message up into 5-grams and then stack them into rows:

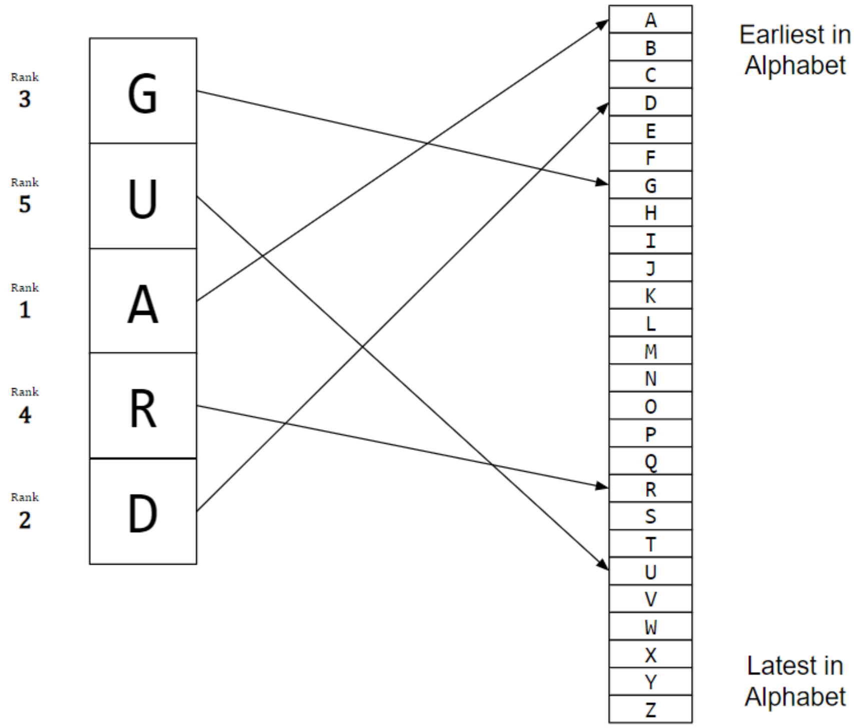
```
HIDET
HEBAB
OONSA
RECOM
INGFO
RYOU
```

We then need to insert some random letters at the end of the message so every row has an equal number of letters. Let's use `Q`:

```
HIDET
HEBAB
OONSA
RECOM
INGFO
RYOUQ
```

Now, we begin the **encryption** process by rearranging the letters in each row based on the alphabetical ranking of the letters of the keyword `GUARD`.

[2] In this course, we won't consider words with repeat letters.



We note that the alphabetical rankings of the letters of this keyword are 3, 5, 1, 4, 2. We can see this as a *permutation*; that is,

$$1 \mapsto 3 \qquad 2 \mapsto 5 \qquad 3 \mapsto 1 \qquad 4 \mapsto 4 \qquad 5 \mapsto 2$$

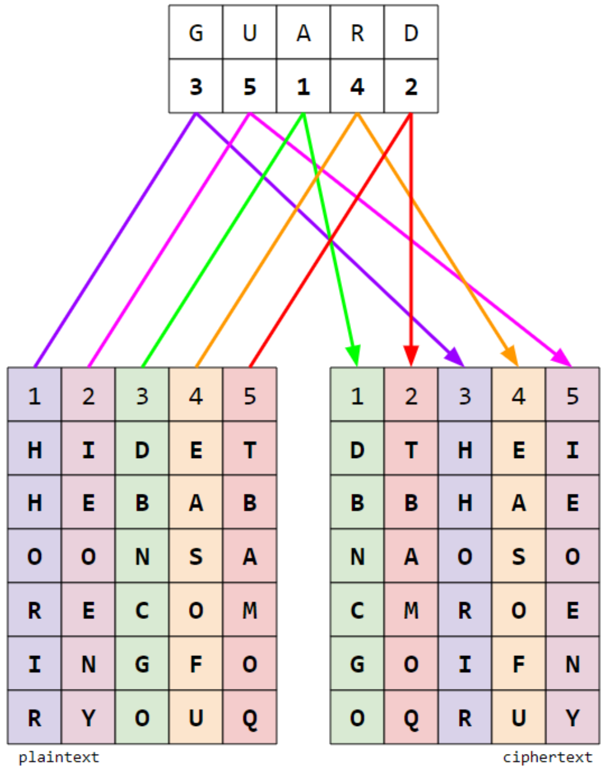
The idea for encryption is that, for each column i, we'll send that column to whatever is mapped by the permutation above. Going back to the stack of letters we have, we can label each individual column:

```
plaintext
position    1 2 3 4 5
            H I D E T
            H E B A B
            O O N S A
            R E C O M
            I N G F O
            R Y O U Q
```

The idea is that

- we can put all letters under position 1 in the plaintext stack to position **3** of the ciphertext stack,
- we can put all letters under position 2 in the plaintext stack to position **5** of the ciphertext stack,
- we can put all letters under position 3 in the plaintext stack to position **1** of the ciphertext stack,
- we can put all letters under position 4 in the plaintext stack to position **4** of the ciphertext stack,
- we can put all letters under position 5 in the plaintext stack to position **2** of the ciphertext stack.

The process, visually, would look like:



Therefore, the ciphertext stack would look like:

```
DTHEI
BBHAE
NAOSO
CMROE
GOIFN
OQRUY
```

Undoing the stacking gives us the ciphertext:

```
DTHEIBBHAENAOSOCMROEGOIFNOQRUY
```

Remark: An easy way to run through the process is to create two "groups," side-by-side. The first group will be the plaintext stack, and the second group will be the ciphertext text. Then, label each column of the first group with the **alphabetical ranking** of the keyword. Label each column of the second group with **12345**. Finally, map each column from the first group to the second group based on the label.

3	5	1	4	2
H	I	D	E	T
H	E	B	A	B
O	O	N	S	A
R	E	C	O	M
I	N	G	F	O
R	Y	O	U	Q

plaintext

1	2	3	4	5
D	T	H	E	I
B	B	H	A	E
N	A	O	S	O
C	M	R	O	E
G	O	I	F	N
O	Q	R	U	Y

ciphertext

Decrypting is merely the inverse of the encryption process.

(Example: Decryption.) Consider the above example again. Suppose Alice successfully sends the following ciphertext to Bob:

```
DTHEIBBHAENAOSOCMROEGOIFNOQRUY
```

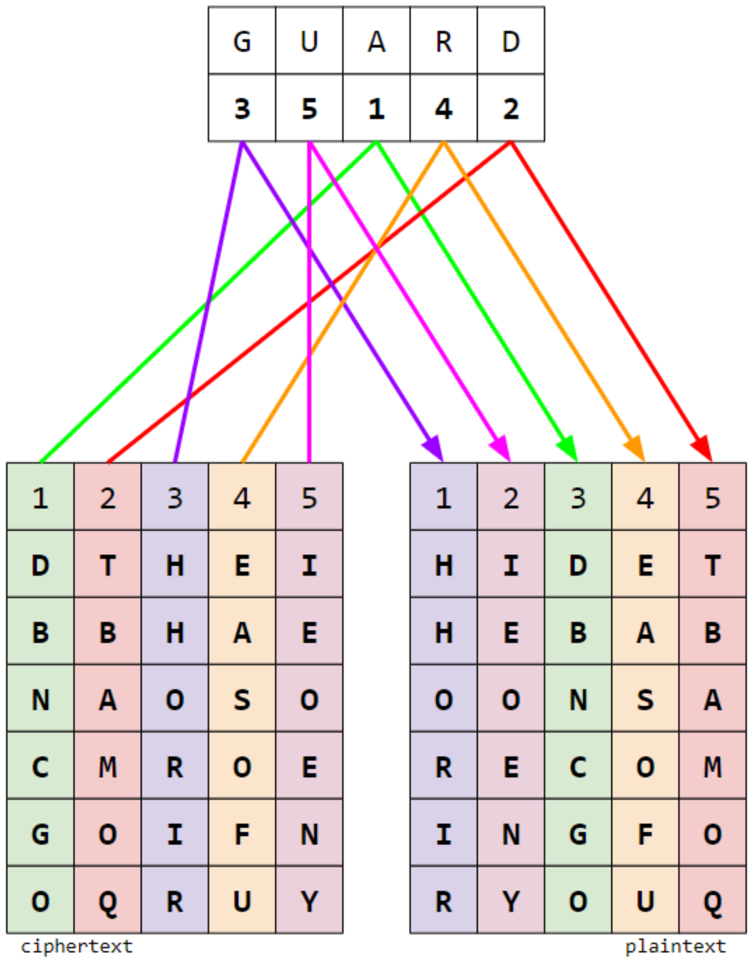
Bob knows that the keyword is `GUARD`. He can use this keyword to decrypt the message. He can begin by taking the letters of the ciphertext and stacking them into rows of 5, since `GUARD` has 5 letters:

```
DTHEI
BBHAE
NAOSO
CMROE
GOIFN
OQRUY
```

Bob also knows the alphabetical ranking of the letters of `GUARD` (which is the same rankings as described above). In particular, the alphabetical ranking is `35142`. So, we need to do the following:

- The letters in position 1 of the ciphertext stack needs to be moved to position **3**,
- the letters in position 2 of the ciphertext stack needs to be moved to position **5**,
- the letters in position 3 of the ciphertext stack needs to be moved to position **1**,
- the letters in position 4 of the ciphertext stack needs to be moved to position **4**,
- the letters in position 5 of the ciphertext stack needs to be moved to position **2**.

The process, visually, would look like:



Undoing the stacking gives us:

```
HIDETHEBABOONSARECOMINGFORYOUQ
```

At this point, Bob needs to make an educated guess as to what the encoded message says (recall that we had to encode the message before encrypting it). By removing the `Q` and correctly punctuating the message, we get

```
Hide! The baboons are coming for you.
```

Remark: We can easily decrypt an encrypted word by doing the inverse of what we did above. Create two "groups," side-by-side. The first group will be the ciphertext stack, and the second group will be the plaintext text. Then, label each column of the first group with **12345**. Label each column of the second group with the **alphabetical ranking** of the keyword. Finally, map each column from the first group to the second group based on the label.

1	2	3	4	5
D	T	H	E	I
B	B	H	A	E
N	A	O	S	O
C	M	R	O	E
G	O	I	F	N
O	Q	R	U	Y

ciphertext

3	5	1	4	2
H	I	D	E	T
H	E	B	A	B
O	O	N	S	A
R	E	C	O	M
I	N	G	F	O
R	Y	O	U	Q

plaintext

(Exercise: Encryption.) *Encrypt the message* `There is always hope.` *using the keyword* *CRASH.*

First, we encode the message so that we can easily encrypt it:

```
THEREISALWAYSHOPE
```

Noting that `CRASH` has length 5, we break the now encoded message into groups of 5 letters (5-grams):

```
THERE
ISALW
AYSHO
PE
```

Let's now add nonsense letters at the end of the last row so every row has 5 letters:

```
THERE
ISALW
AYSHO
PEABC
```

Now, we note the alphabetical ranking of each letter in `CRASH`:

$$C \mapsto 2 \quad R \mapsto 4 \quad A \mapsto 1 \quad S \mapsto 5 \quad H \mapsto 3.$$

Using the streamlined way discussed above, we have

```
2 4 1 5 3 | 1 2 3 4 5
T H E R E | E T E H R
I S A L W | A I W S L
A Y S H O | S A O Y H
P E A B C | A P C E B
```

Unstacking the new rows gives us the ciphertext:

```
ETEHRAIWSLSAOYHAPCEB
```

(Exercise: Decryption.) *Decrypt the message* `ETIHGFREAFRSLAESOXOE` *using the keyword* `CRASH`.

Begin by grouping the letters into 5-grams, since `CRASH` has length 5:

```
ETIHG
FREAF
RSLAE
SOXOE
```

Recall that the alphabetical ranking of each letter in `CRASH` is `24153`. Using the streamlined way discussed above, we have

```
1 2 3 4 5    2 4 1 5 3
E T I H G -> T H E G I
F R E A F -> R A F F E
R S L A E -> S A R E L
S O X O E -> O O S E X
```

Unstacking the new rows gives us the plaintext:

```
THEGIRAFFESARELOOSEX
```

Decoding the message gives us:

```
The giraffes are loose.
```

(Exercise.) Encrypt the message `Meet at the trolley station.` using keyword `UCSD`.

Encoding, grouping the resulting letters into groups of 4, and adding a nonsense letter gives us:

```
MEET
ATTH
ETRO
LLEY
STAT
IONX
```

Noting that the alphabetical ranking of `UCSD` is `4132`, we can use the streamlined way discussed above to get the encrypted result:

```
4 1 3 2    1 2 3 4
M E E T -> E T E M
A T T H -> T H T A
E T R O -> T O R E
L L E Y -> L Y E L
S T A T -> T T A S
I O N X -> O X N I
```

Unstacking the result gives us:

```
/ETEMTHTATORELYELTTASOXNI
```

(Exercise.) Alice and Bob share the keyword `ZEUS`. Alice uses rectangular tranposition to encrypt the following nonsense message:

`MTSQAGXY`

What is the corresponding ciphertext?

Encoding, grouping the resulting letters into groups of 4, and adding a nonsense letter gives us:

```
MTSQ
AGXY
```

Noting that the alphabetical ranking of `ZEUS` is `4132`, we can use the streamlined way discussed above to get the encrypted result:

```
4 1 3 2    1 2 3 4
M T S Q -> T Q S M
A G X Y -> G Y X A
```

Unstacking the result gives us:

```
TQSMGYXA
```

(Exercise.) The following message was encrypted using rectangular transposition with the keyword `SNAKE`. What is the plaintext?

`DSUEMSEDIAJQQDA`

`SNAKE` has alphabetical ranking `54132`. With this in mind, stacking the letters of the encrypted message into groups of 5 and then running the streamlined process gives us:

```
1 2 3 4 5    5 4 1 3 2
D S U E M -> M E D U S
S E D I A -> A I S D E
J Q Q D A -> A D J Q Q
```

Unstacking the result gives us:

```
MEDUSAISDEADJQQ
```

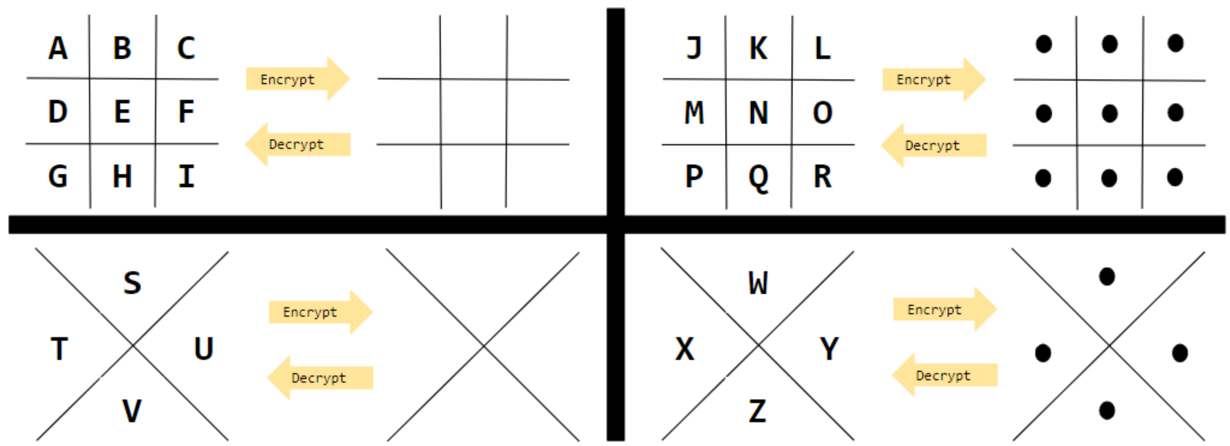
Decoding gives us:

```
Medusa is dead.
```

2.2 Masonic Cipher

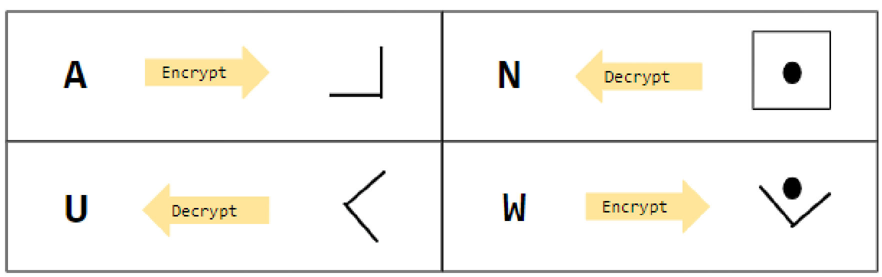
The masonic cipher (also known as the *pigpen cipher* or *tic-tac-toe cipher*) is a simple substitution cipher that replaces individual letters with certain geometric shapes.

For example, consider the following diagram, which represents a Masonic cipher for the English letters:



The idea is that we can replace a letter (e.g., `A`) with a corresponding geometric shape (e.g., the backwards `L` represented by the top-left part of the grid.)

Some other examples based on the above cipher are shown below:



Note that there is *no key* associated with this cipher. There is only a decryption function (which is just mapping the geometric shape back to the letter). Therefore, the adversary, who knows that a message was encrypted using a masonic cipher, can recover the plaintext easily.

2.3 Caesar Cipher

The Caesar cipher, also known as a *shift cipher*, is a simple substitution cipher that *shifts* a letter by some amount n. Hence, the key for this cipher is an integer n. The idea is that we initially assign each letter an integer, perhaps by their alphabetical ranking (e.g., A is 0, B is 1, and so on.) If we want to shift the letters by some number, we can just "move" the letters by that amount. If a letter gets a new integer that's greater than 25, we can "wrap" the letter back.

Consider the following diagram, which shows the correspondence between the plaintext alphabet and the ciphertext alphabet.

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

In this particular diagram, when we apply a shift, we apply the shift to the *plain* row. By doing this, we can translate whatever plaintext we have to ciphertext.

(Example.) If we shift each letter by 3 (i.e., $n = 3$), we have

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

Notice how A now corresponds to 3. Recall that A's original position was 0; if we shift each letter by 3, we essentially add 3 to A's original position to get the new position

$$0 + 3 = 3.$$

The same idea applies to any other letter. One key thing to notice is how X, Y, and Z were *wrapped back* to the beginning. In any case, let's see how translation would work in this case:

- To convert a letter from plaintext to ciphertext, look for the letter in the(shifted) plaintext row and then look at the corresponding ciphertext column. For example, R in plaintext would become U in ciphertext.
- To convert a letter from ciphertext to plaintext, look for the letter in the ciphertext row and then look at the corresponding (shifted) plaintext column. For example, U in ciphertext becomes R in plaintext.

(Example.) If we shift each letter by -2 (i.e., $n = -2$), we have

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

As with rectangular tranposition, we should encode the message by removing any non-alphabetic characters and capitalizing everything.

(Exercise.)

- *Using a shift of 3, encrypt the message* `Meet at La Jolla Shores.`

  Encoding the message gives us `MEETATLAJOLLASHORES`. Then, we can use the example above (with the shift of 3) to give us the proper correspondence.

  ```
  plain     M E E T A T L A J O L L A S H O R E S
  cipher    P H H W D W O D M R O O D V K R U H V
  ```

  This gives us `PHHWDWODMROODVKRUHV`.

- *Using a shift of 3, decrypt the message* `PHHWDWVXQJRGODZQ`

Using the example above (with the shift of 3), we have

```
cipher    P H H W D W V X Q J R G O D Z Q
plain     M E E T A T S U N G O D L A W N
```

Decoding this gives us `Meet at Sun God Lawn.`

(Exercise.) *You are Eve. You have just intercepted the following message that Alice was trying to send to Bob:* `Q TQDM IB QPWCAM`*. You know that Alice used a Caesar cipher, but she didn't remove spaces before encrypting: she left the spaces in her original message as-is. What is the original message?*

`Q` itself could be a word; specifically, it could either be `A` or `I`. We can try to figure out what the message by guessing which word the first word could be.

- If `Q` maps to `A`, then we have

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

  Partially decrypting the ciphertext gives us `A DANW`, but `DANW` is meaningless. Therefore, it cannot be `A`.

- If `Q` maps to `I`, then we have

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

  Decrypting this gives us:

```
I LIVE AT IHOUSE
```

Therefore, the message is `I LIVE AT IHOUSE`. The shift was 8.

(Exercise.) Alice encrypts the following message using a Caesar cipher with a shift of 1.

`Zeus is hiding in a cave`

What is the corresponding ciphertext?

```
plain     ZEUSISHIDINGINACAVE
cipher    AFVTJTIJEJOHJOBDBWF
```

Essentially, we just move all letters forward by 1.

2.4 Interlude: Modular Arithmetic

One fundamental idea in number theory, which is used in cryptography, is modular arithmetic.

2.4.1 Quotients and Remainders

Lemma 2.1: Euclid's Division

For any integer a and positive integer n, there exists a unique pair of integers q and r such that $0 \leq r < n$ and $a = qn + r$. The integers q and r are called the quotient and remainder, respectively. We also write $a \pmod{n}$ to refer to the remainder.

For the proof, the deal is that we can keep subtracting, or adding, n from a until we end up in the range $[0, n)$. Therefore, the number of times we had to subtract, or add, n is the *quotient*, and the number in the range $[0, n)$ that we end up with at the end is the *remainder*.

(Example.) Divide $a = 17$ by $n = 5$. Find the quotient and remainder.

Using the proof idea, we note that:

- Subtracting 5 to a once gives us 12.
- Subtracting 5 to a twice gives us 7.
- Subtracting 5 to a thrice gives us 2.

It took us 3 subtractions to get to a number that's in the range $[0, 5)$, so the quotient is $\boxed{3}$ and the remainder is $\boxed{2}$.

We should note that this is pretty standard when $a \geq 0$. However, for $a < 0$, it might be less familiar, albeit the same process.

(Example.) Divide $a = -7$ by $n = 5$. Find the quotient and remainder.

Using the proof idea, we note that:

- Adding 5 to a once gives us 2.
- Adding 5 to a twice gives us 3.

It took us 2 additions to get to a number that's in the range $[0, 5)$, so the quotient is $\boxed{-2}$ (because we had to *add*, not subtract) and the remainder is $\boxed{3}$.

Remark:

- If we have to **add** n to a x times to get a number that's in the range $[0, n)$, then our final quotient will be negative (that is, $-x$).
- If we have to **subtract** n from a x times to get a number that's in the range $[0, n)$, then our final quotient will be positive (that is, x).

(Exercise.) For each of the following, calculate the quotient and remainder when a is divided by n. Do these calculations by hand.

- $a = 13$, $n = 3$.

  We know that $13/3 = 4$, and $13 - (3 \cdot 4) = 1 \in [0, 3)$. So, the quotient is $\boxed{4}$ and the remainder is $\boxed{1}$.

- $a = 134$, $n = 10$.

We know that $134/10 = 13$ and $134 - (10 \cdot 13) = 4 \in [0, 10)$. So, the quotient is $\boxed{13}$ and remainder is $\boxed{4}$.

- $a = -37$, $n = 10$.

We know that we need to add n to a **4** times to get a number, 3, that is in the range $[0, 10)$. To be precise,
$$-37 + 10 + 10 + 10 + 10 = -37 + 40 = 3 \in [0, 10).$$
Therefore, the quotient is $\boxed{-4}$ and the remainder is $\boxed{3}$.

- $a = -15$, $n = 60$.

We have to add n to a **1** time to get $45 \in [0, 60)$. Therefore, the quotient is $\boxed{-1}$ and the remainder is $\boxed{45}$.

- $a = 13$, $n = 12$.

We know that $13/12 = 1$ and $13 - (12 \cdot 1) = 1$. So, the quotient is $\boxed{1}$ and the remainder is $\boxed{1}$.

(Exercise.) What is $-13 \pmod 5$?

$$-13 + 5 + 5 + 5 = 2 \in [0, 5),$$
so the quotient is -3 (since we had to perform 3 additions) and the remainder is $\boxed{2}$. Therefore,
$$-13 \pmod 5 = 2.$$

Proposition. *Suppose a and n are integers and $n > 0$. All the following statements are equivalent:*

- $a \pmod n = 0$.
- *There is no remainder when a is divided by n.*
- *a is a multiple of n.*
- *a is divisible by n.*
- *n is a divisor of a.*
- *n is a factor of a.*
- *n divides a (in notation[3]: $n|a$).*
- *a/n is an integer.*

[3] Note that | is read as "*divides*."

2.4.2 Congruences

Definition 2.2: Congruence

Fix a positive integer n. If a and b are integers, we say that "a is **congruent** to b mod n," or that "a and b are congruent mod n," if a and b have the same remainder when each is divided by n. This can be denoted in symbols as follows:

$$a \equiv b \pmod{n}.$$

For example, $19 \equiv 7 \pmod{4}$ since 19 and 7 both have remainder 3 when divided by 4. Observe also that $19 - 7 = 12$ is a multiple of 4. This can be generalized:

Lemma 2.2

Fix a positive integer n. Two integers a and b are congruent mod n if and only if $a - b$ is a multiple of n.

Proof. Divide a and b by n to write $a = q_1 n + r_1$ and $b = q_2 n + r_2$. If

$$a \equiv b \pmod{n},$$

this by definition means that $r_1 = r_2$ so

$$a - b = (q_1 n + r_1) - (q_2 n + r_2) = q_1 n - q_2 n = n(q_1 - q_2).$$

So, $a - b$ is a multiple of n. Conversely, suppose $a - b$ is a multiple of n. Then,

$$(a - b) - (q_1 - q_2)n = ((q_1 n + r_1) - (q_2 n + r_2)) - (q_1 - q_2)n = r_1 - r_2$$

is a multiple of n. Since $0 \leq r_1, r_2 < n$, however, we must have $|r_1 - r_2| < n$. The only way that $r_1 - r_2$ can be a multiple of n is if $r_1 - r_2 = 0$, i.e., if $r_1 = r_2$. That means $a \equiv b \pmod{n}$. $\square$

Theorem 2.1: Modular Arithmetic Theorem

Fix a positive integer n. Suppose a, a', b, b' are integers such that

$$a \equiv a' \pmod{n}$$

$$b \equiv b' \pmod{n}$$

and k is any positive integer. Then, all of the following are also true:

$$a + b \equiv a' + b' \pmod{n}$$

$$a - b \equiv a' - b' \pmod{n}$$

$$ab \equiv a'b' \pmod{n}$$

$$a^k \equiv (a')^k \pmod{n}$$

(Exercise.) Use the Modular Arithmetic Theorem to quickly calculate the following.

- $417 \cdot 22 \pmod{10}$.

$$\begin{aligned} 417 \cdot 22 &\equiv 7 \cdot 2 \\ &= 14 \\ &\equiv 4 \pmod{10}. \end{aligned}$$

- $333333 + 666 \pmod 3$.

$$\begin{aligned} 333333 + 666 &\equiv 0 + 0 \\ &\equiv 0 \pmod 3. \end{aligned}$$

- $7^{202320232023} \pmod 6$.

$$\begin{aligned} 7^{202320232023} &= 7 \cdot 7 \cdot \ldots \cdot 7 \\ &\equiv 1 \cdot 1 \cdot \ldots \cdot 1 \\ &= 1 \pmod 6. \end{aligned}$$

- What is $5^{2023202320232023} \pmod 6$?

$$\begin{aligned} 5^{2023202320232023} &= 5 \cdot 5 \cdot \ldots \cdot 5 \\ &\equiv (-1) \cdot (-1) \cdot \ldots \cdot (-1) \\ &= (-1)^{2023202320232023} \\ &\equiv -1 \\ &\equiv 5 \pmod 6. \end{aligned}$$

Therefore, the answer is $\boxed{5}$.

(Exercise.) Fix positive integers k and n. Suppose a and a' are integers such that $a \equiv a' \pmod n$. It is not true in general that $k^a \equiv k^{a'} \pmod n$. Show this by example: in other words, find k, n, a, and a' such that $a \equiv a' \pmod n$ but $k^a \not\equiv k^{a'} \pmod n$.

Let $k = 2$, $n = 5$, $a = 6$, and $a' = 1$ so that

$$6 \equiv 1 \pmod 5.$$

Then, we note that

$$k^a = 2^6 = 64$$

and

$$k^{a'} = 2^1 = 2.$$

From this, it's clear that

$$64 \not\equiv 2 \pmod 5.$$

(Exercise.) *Suppose that the number $273x49y5$, where x and y are unknown digits, is divisible by 495. Find x and y.*

We are asked to solve
$$273x49y5 \equiv 0 \mod 495.$$
We can write $273x49y5$ as
$$20000000 + 7000000 + 300000 + 10000x + 4000 + 900 + 10000y + 5.$$
With this in mind, we have
$$\begin{aligned} 20000000 + 7000000 + 300000 + 10000x + 4000 + 900 + 10y + 5 \\ &\equiv 20 + 205 + 30 + 100x + 40 + 405 + 10y + 5 \\ &= 705 + 100x + 10y \\ &\equiv 210 + 100x + 10y \mod 495. \end{aligned}$$
We note that the next multiple of 495 is 990. So, effectively, we want to find some x and y such that $0 \leq x < 10$ and $0 \leq y < 10$ and
$$210 + 100x + 10y = 990.$$
This gives us
$$100x + 10y = 780.$$
One obvious solution is $x = 7$ and $y = 8$.

2.4.3 Revisiting the Caesar Cipher

Suppose we identify the letters A through Z with the numbers 0 through 25. In other words, we have $A \mapsto 0$, $B \mapsto 1$, and so on. Suppose we want to apply the Caesar cipher with a shift of 5 to encrypt the letter Y. Consider the following
$$E(x) = (x + 5) \pmod{26}.$$
We note that Y corresponds to the number 24. Then, it follows that
$$E(24) = (24 + 5) \pmod{26} = 29 \pmod{26} = 3.$$
The number 3 corresponds to the letter D, the desired result. In other words, if we can identify the letters with numbers, the function E is the encryption function of the Caesar cipher with a shift of 5.

The decryption function is given by
$$D(y) = (y - 5) \pmod{26}.$$
So, if we wanted to decrypt the letter D, which corresponds to the number 3, then
$$D(3) = (3 - 5) \pmod{26} = -2 \pmod{26} = 24,$$
which corresponds to Y.

What we just did is actually a consequence of the Modular Arithmetic Theorem; for a quick little "proof," notice how
$$\begin{aligned} D(E(x)) &= D(y) \\ &\equiv (y - 5) \pmod{26} \\ &\equiv ((x + 5) - 5) \pmod{26} \\ &= x. \end{aligned}$$

(Exercise.) Decipher the message below, which was encrypted using a Caesar cipher with a shift of 3 and then using a rectangular transposition with the keyword `EARLY`.

```
DKSSBUIGLDEBXOX
```

To decrypt this message, we need to work backwards: first, use rectangular transposition to undo the first encryption, and then Caesar cipher to undo the second encryption.

1. For the rectangular transposition, note that the keyword has alphabetical ranking `21435`, so using the streamlined way discussed earlier, we have

```
12345    21435
DKSSB -> KDSSB
UIGLD -> IULGD
EBXOX -> BEOXX
```

   Unstacking gives us `KDSSBIULGDBEOXX`.

2. Next, we need to undo the Caesar cipher encryption on the message that we found from the previous step. Since the encryption used a positive shift of 3, undoing it requires us to use a negative shift of 3. This gives us:

```
encrypted   KDSSBIULGDBEOXX
decrypted   HAPPYFRIDAY....
```

   Note that the last four letters were omitted. In any case, this gives us the decoded message `Happy Friday`.

Remark: You should not assume that these operations are commutative. That is, if we were to decrypt the message by applying the Caesar cipher first and then the rectangular transposition, as opposed to the reverse order, we may get a different answer!

2.5 Interlude: GCDs

Definition 2.3: Greatest Common Divisor

The **greatest common divisor** (or *GCD*) of two integers a and b that are not both zero is denoted $\gcd(a, b)$ and is defined to be the largest integer which is both a divisor of a and a divisor of b.

(Example.) Suppose we wanted to compute $\gcd(14, 21)$.

- The factors of 14 are 1, 2, 7, and 14.
- The factors of 21 are 1, 3, 7, and 21.

Therefore, as 7 is the *largest integer* which is both a divisor of 14 and 21, it follows that $\gcd(14, 21) = 7$.

Note that, while intuitive, this is actually not the best way of finding GCDs. Finding the factors of a number, especially a large one, is difficult. However, there exists algorithms that we can use to quickly calculate GCDs.

(Example.) Suppose a is a nonzero integer. What is $\gcd(a, 0)$?

The answer is $\gcd(a, 0) = |a|$. To see why this is the case, consider the following points.

1. If $a \neq 0$, the largest value that divides a is $|a|$.

   For example, the largest value that divides 100 is $|100| = 100$. Likewise, the largest value that divides -100 is still $|-100| = 100$.

2. If you think about it, all integers divide 0.

   Recall that, if a and b are integers, a divides b if there is an integer c such that

   $$ac = b.$$

   Here, we write that $a|b$ to mean that a divides b.

   With this in mind, we note that
   $$a \cdot 0 = 0$$
   and therefore
   $$a|0.$$

3. Therefore, it follows that $\gcd(a, 0) = |a|$.

   To see this, note that the factors of 10 and -10 are

   $$\{-10, -5, -2, -1, 1, 2, 5, 10\},$$

   and we know that all factors of 0 are effectively all integers. Therefore, it follows that 10 would be the answer here.

2.5.1 Euclidean Algorithm

The Euclidean Algorithm for computing GCDs relies on the following observation, defined as a lemma.

Lemma 2.3

Let n be a positive integer and $a \equiv b \pmod{n}$. Then, $\gcd(a, n) = \gcd(b, n)$.

Proof. Let $c = \gcd(a, n)$ and $d = \gcd(b, n)$. Let k be an integer such that

$$a - b = nk.$$

Since c is a factor of both a and n, it is also a factor of $a - nk = b$. Thus, c is a common factor of both b and n as well, so $c \leq d$ bby definition of d. On the other hand, the same logic shows that d is a common factor of both a and n, so $d \leq c$ and thus $d = c$. □

Corollary 2.1

Let n be a positive integer and let r be the remainder when an integer a is divided by n. Then, $\gcd(a, n) = \gcd(r, n)$.

This brings us to the Euclidean Algorithm:

Suppose a and b are two positive integers, and assume without loss of generality (WLOG) that $b \geq a$. To find $\gcd(a, b)$, we can do the following:

- Divide b by a and let r be the remainder. Then,
  - If $r = 0$, output a.
  - Otherwise, replace b with a and a with r. Then, repeat.

(Example.) Suppose we wanted to compute $\gcd(115, 35)$. We divide the bigger number by the smaller one and get

$$115 = 3 \cdot 35 + 10.$$

The remainder, $r = 10$, is nonzero, so we'll divide again, but this time, we'll divide the dividend (35) by the remainder (10) to get

$$35 = 3 \cdot 10 + 5.$$

The remainder is nonzero again, so we repeat to get

$$10 = 2 \cdot 5 + 0.$$

Since the remainder is 0, we output the dividend: $\boxed{5}$. Therefore,

$$\gcd(115, 35) = 5.$$

(Exercise.) Compute the following GCDs using the Euclidean Algorithm.

- $\gcd(180, 120)$.

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
120	180	$180 = 120q + r$	1	60
60	120	$120 = 60q + r$	2	0

Therefore, the answer must be $\boxed{60}$.

- $\gcd(180, 81)$.

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
81	180	$180 = 81q + r$	2	18
18	81	$81 = 18q + r$	4	9
9	18	$18 = 9q + r$	2	0

Therefore, the answer must be $\boxed{9}$.

- $\gcd(121, 77)$.

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
77	121	$121 = 77q + r$	1	44
44	77	$77 = 44q + r$	1	33
33	44	$44 = 33q + r$	1	11
11	33	$33 = 11q + r$	3	0

Therefore, the answer must be $\boxed{11}$.

2.5.2 Bezout's Theorem

Theorem 2.2: Bezout's Theorem

Suppose a and b are integers not both 0. Then, $\gcd(a, b)$ can be written as an *integer linear combination* of a and b, i.e., it can be written as $ax + by$ for some integers x and y. Integers x and y such that

$$\gcd(a, b) = ax + by$$

are called **Bezout's coefficients**.

We can use the Euclidean Algorithm to find the Bezout coefficients, as seen in the example below.

(Example.) Suppose we want to find the Bezout coefficients for $\gcd(115, 35)$. Recall the sequence of operations we had to do:

$$115 = 3 \cdot 35 + 10.$$

$$35 = 3 \cdot 10 + 5.$$

$$10 = 2 \cdot 5 + 0.$$

Suppose we rearrange the first and second equations, like so:

$$10 = 115 - 3 \cdot 35.$$

$$5 = 35 - 3 \cdot 10.$$

Plugging in the first equation into the second equation gives us

$$5 = 35 - 3 \cdot (115 - 3 \cdot 35).$$

Simplifying this gives us

$$\begin{aligned} 5 &= 35 - 3 \cdot (115 - 3 \cdot 35) \\ &= 35 - 3(115) + 9(35) \\ &= 10(35) - 3(115). \end{aligned}$$

Notice how we wrote $\gcd(115, 35)$ as an integer linear combination of those two numbers.

Essentially, the steps are as follows:

1. Find the GCD using the Euclidean Algorithm.
2. Rewrite the division for the *last nonzero remainder*.
3. Alternate between substitution for the remainder directly above, and then simplify. Alternatively, start from the last equation with a nonzero remainder and then keep using the equations before that equation (e.g., from equation n, the last equation with a nonzero remainder, substitute equation $n - 1$ in the next step. Then, in the next step, substitute equation $n - 2$. Keep doing this until you reach equation 1.)

(Example.) Suppose we want to find the Bezout coefficients for $\gcd(240, 46)$.

1. First, let's compute the GCD, keeping note of the sequence of operations we made.

a	b	b = aq + r	q	r
46	240	$240 = 46q + r$	5	10
10	46	$46 = 10q + r$	4	6
6	10	$10 = 6q + r$	1	4
4	6	$6 = 4q + r$	1	2
2	4	$4 = 2q + r$	2	0

This tells us that $\gcd(240, 46) = 2$. The operations we did were

- (Eq. 1) $240 = 46(5) + 10 \implies 10 = 240 - 46 \cdot 5$
- (Eq. 2) $46 = 10(4) + 6 \implies 6 = 46 - 10 \cdot 4$
- (Eq. 3) $10 = 6(1) + 4 \implies 4 = 10 - 6 \cdot 1$
- (Eq. 4) $6 = 4(1) + 2 \implies 2 = 6 - 4 \cdot 1$
- (Eq. 5) $4 = 2(2) + 0$

2. Rewriting the division for the last equation with the nonzero remainder (Eq. 4) gives us $2 = 6 - 4 \cdot 1$.

3. Starting from the division for the last nonzero remainder, let's rewrite it:

$$\begin{aligned}
2 &= 6 - 4 \cdot 1 && \text{From Eq. 4} \\
&= 6 - (\underbrace{10 - 6 \cdot 1}_{\text{Eq. 3}}) \cdot 1 && \text{Substitute Eq. 3} \\
&= 6 - 10 + 6 && \text{Expand} \\
&= 2 \cdot 6 - 1 \cdot 10 && \text{Rewrite to group like terms} \\
&= 2 \cdot (\underbrace{46 - 10 \cdot 4}_{\text{Eq. 2}}) - 1 \cdot 10 && \text{Substitute Eq. 2} \\
&= 2 \cdot 46 - 2 \cdot 10 \cdot 4 - 1 \cdot 10 && \text{Expand} \\
&= 2 \cdot 46 - 8 \cdot 10 - 1 \cdot 10 && \text{Simplify} \\
&= 2 \cdot 46 - 9 \cdot 10 && \text{Rewrite to group like terms} \\
&= 2 \cdot 46 - 9 \cdot (\underbrace{240 - 46 \cdot 5}_{\text{Eq. 1}}) && \text{Substitute Eq. 1} \\
&= 2 \cdot 46 - 9 \cdot 240 + 46 \cdot 5 \cdot 9 && \text{Expand} \\
&= 2 \cdot 46 - 9 \cdot 240 + 46 \cdot 45 && \text{Simplify} \\
&= 47 \cdot 46 - 9 \cdot 240 && \text{Rewrite to group like terms}
\end{aligned}$$

Notice how the Bezout coefficients are 47 and -9.

(Exercise.) Calculate Bezout's coefficients for the following GCDs using the extended Euclidean Algorithm.

- $\gcd(180, 120)$.

1. First, compute the GCD. We already did this in a previous exercise, but just to reiterate:

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
120	180	$180 = 120q + r$	1	60
60	120	$120 = 60q + r$	2	0

   Therefore, the GCD is 60. The operations that we did were
   - (Eq. 1) $180 = 120(1) + 60 \implies 60 = 180 - 120(1)$
   - (Eq. 2) $120 = 60(2) + 0$
2. Next, we just need to rewrite the last equation with a nonzero remainder.
$$180 = 120(1) + 60 \implies 60 = 180 - 120(1)$$
3. Finally, we need to work backwards, substituting the previous equations. Because we only have one operation which resulted in a non-zero remainder, it follows that we only need to do:
$$60 = 180 - 120(1).$$
   Therefore, the Bezout coefficients are $\boxed{1}$ and $\boxed{-1}$.

- $\gcd(180, 81)$.

1. First, we need to compute the GCD. We already did this in a previous exercise, but to reiterate:

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
81	180	$180 = 81q + r$	2	18
18	81	$81 = 18q + r$	4	9
9	18	$18 = 9q + r$	2	0

   Therefore, the GCD is 9. The operations we did were
   - (Eq. 1) $180 = 81(2) + 18 \implies 18 = 180 - 81(2)$
   - (Eq. 2) $81 = 18(4) + 9 \implies 9 = 81 - 18(4)$
   - (Eq. 3) $18 = 9(2) + 0$
2. Next, we need to rewrite the last equation with a nonzero remainder.
$$81 = 18(4) + 9 \implies 9 = 81 - 18(4).$$
3. Finally, we need to work backwards, substituting the previous equations as needed.
$$\begin{aligned} 9 &= 81 - 18(4) \\ &= 81 - \underbrace{(180 - 81(2))}_{\text{Eq. 1}} \cdot 4 \\ &= 81 - 180(4) + 81(8) \\ &= 81(9) - 180(4) \end{aligned}$$
   Therefore, the Bezout coefficients are $\boxed{9}$ and $\boxed{-4}$.

- $\gcd(121, 77)$.

1. First, compute the GCD. To reiterate:

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
77	121	$121 = 77q + r$	1	44
44	77	$77 = 44q + r$	1	33
33	44	$44 = 33q + r$	1	11
11	33	$33 = 11q + r$	3	0

   Therefore, the GCD is 11. The operations that we did were
   - (Eq. 1) $121 = 77(1) + 44 \implies 44 = 121 - 77(1)$
   - (Eq. 2) $77 = 44(1) + 33 \implies 33 = 77 - 44(1)$
   - (Eq. 3) $44 = 33(1) + 11 \implies 11 = 44 - 33(1)$
   - (Eq. 4) $33 = 11(3) + 0$

2. Next, rewrite the last equation with a nonzero remainder.

$$44 = 33(1) + 11 \implies 11 = 44 - 33(1).$$

3. Finally, work backwards.

$$\begin{aligned} 11 &= 44 - 33(1) \\ &= 44 - \underbrace{(77 - 44(1))}_{\text{Eq. 2}} \cdot 1 \\ &= 44 - 77 + 44(1) \\ &= 44(2) - 77 \\ &= \underbrace{(121 - 77(1))}_{\text{Eq. 1}} \cdot 2 - 77 \\ &= 121(2) - 77(2) - 77 \\ &= 121(2) - 77(3). \end{aligned}$$

   Thereforem the Bezout coefficients are $\boxed{2}$ and $\boxed{-3}$.

(Exercise.) Observe that $\gcd(42, 12) = 6$. Show that the pairs $(-1, 4)$ and $(1, -3)$ are both Bezout coefficients for 42 and 12.

- For the pair $(-1, 4)$, we have

$$42(-1) + 12(4) = -42 + 48 = 6.$$

- For the pair $(1, -3)$, we have

$$42(1) + 12(-3) = 42 - 36 = 6.$$

(Exercise.) Consider $\gcd(150, 90)$.

1. How many divisions do we need to do until we see a remainder of 0 when we use the Euclidean algorithm to compute $\gcd(150, 90)$?

a	b	b = aq + r	q	r
90	150	$150 = 90q + r$	1	60
60	90	$90 = 60q + r$	1	30
30	60	$60 = 30q + r$	2	0

We had to perform **3** divisions.

2. Find Bezout coefficients for $\gcd(150, 90)$.

Noting that $\gcd(150, 90) = 30$ and the equations we worked with are

- (Eq. 1) $150 = 90(1) + 60 \implies 60 = 150 - 90(1)$
- (Eq. 2) $90 = 60(1) + 30 \implies 30 = 90 - 60(1)$
- (Eq. 3) $60 = 30(2) + 0$

Starting with Eq. 2, we have

$$\begin{aligned} 30 &= 90 - 60(1) \\ &= 90 - (\underbrace{150 - 90(1)}_{\text{Eq. 1}})(1) \\ &= 90 - 150 + 90 \\ &= 90(2) + 150(-1). \end{aligned}$$

So, the Bezout coefficients are $\boxed{2}$ and $\boxed{-1}$.

2.5.3 Modular Inversion

Suppose you are asked to solve the equation

$$5z = 7.$$

Intuitively, we can just divide both sides by 5. Stated differently, we can multiply both sides by $\frac{1}{5}$:

$$\left(\frac{1}{5}\right) \cdot 5z = \left(\frac{1}{5}\right) 7 \implies z = \frac{5}{7}.$$

In other words, we're able to "cancel out" the 5 that appears on the left-hand side, thus isolating z.

With modular inversion, we can recreate this process with *congruences*. For example, suppose we want to solve

$$5z \equiv 7 \pmod{11}.$$

We cannot "divide both sides by 5" because congruences only make sense when both sides of the congruence are *integers*. But, if we find an integer x with the property that

$$5x \equiv 1 \pmod{11},$$

then we can multiply both sides of our congruence by x to effectively eliminate the 5 on the left-hand side. In this example, there *is* an integer: $x = 9$. Using this integer, we have

$$5x = 9 \cdot 5 = 45 \equiv 1 \pmod{11}.$$

Therefore, multiplying both sides of our congruence by 9 gives us

$$z = 1 \cdot z \equiv (5 \cdot 9)z = 9 \cdot (5z) \equiv 9 \cdot 7 \pmod{11}.$$

Thus,

$$z \equiv 9 \cdot 7 = 63 \equiv 8 \pmod{11},$$

and we've solved our congruence: $z \equiv 8 \pmod{11}$. While we solved this congruence, note that we basically guessed what the solution is. However, there's a way to get such x.

Definition 2.4

Fix a positive integer n. An integer a is *invertible* mod n (or a *unit* mod n) if there exists another integer x such that $ax \equiv 1 \pmod{n}$. The number x is then called an *inverse of* a mod n and, in symbols, one writes $x \equiv a^{-1} \pmod{n}$.

So, in the above example, we found that $9 \equiv 5^{-1} \pmod{11}$ because $5 \cdot 9 \equiv 1 \pmod{11}$.

(Exercise.) Explain why 2 is not invertible mod 4.

Essentially, we need to show why there does not exist an integer x such that

$$2x \equiv 1 \pmod{4}.$$

However, notice that both 2 and 4 are even. Therefore, multiplying 2 by any integer gives us an even number. Because 4 is even as well, it follows that we'll never be able to find an x such that $2x \equiv 1 \pmod{4}$.

Theorem 2.3: Modular Inversion Theorem

Fix a positive integer n and another integer a. Then, a is invertible mod n if and only if $\gcd(a, n) = 1$. Moreover, if $\gcd(a, n) = 1$ and x and y are Bezout coefficients for a and n, then x is an inverse of a mod n.

(Example.) Suppose we want to find the inverse of 7 (mod 23). Using the Euclidean Algorithm to compute $\gcd(23, 7)$, we get

$$23 = 3 \cdot 7 + 2$$

$$7 = 3 \cdot 2 + 1$$

$$2 = 2 \cdot 1 + 0.$$

So, $\gcd(23, 7) = 1$ and thus 7 is in fact invertible mod 23. Working backwards, we find that

$$\begin{aligned} 1 &= 7 - 3 \cdot 2 \\ &= 7 - 3 \cdot (23 - 3 \cdot 7) \\ &= 10 \cdot 7 - 3 \cdot 23. \end{aligned}$$

Therefore, the Modular Inversion Theorem tells us that 10 is the inverse of 7 mod 23.

(Exercise.) Which of the following integers is invertible mod 210?

(a) 3

(b) 4

(c) 5

(d) None of the above

The answer is **D**. Note that

(a) $\gcd(3, 210) \neq 1$.

(b) $\gcd(4, 210) \neq 1$.

(c) $\gcd(5, 210) \neq 1$.

So, by theorem (2.3), the answer must be D.

(Exercise.) For each of the following, determine whether a is invertible mod n. If it is, find an inverse of a mod n.

- $a = 14, n = 21$.

  First, let's calculate $\gcd(14, 21)$.

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
14	21	$21 = 14q + r$	1	7
7	14	$14 = 7q + r$	2	0

  Therefore, $\gcd(14, 21) = 7$. By Theorem (2.3), it follows that 14 is not invertible mod 21.

- $a = 3, n = 7$.

First, we calculate $\gcd(3, 7)$.

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
3	7	$7 = 3q + r$	2	1
1	3	$3 = 1q + r$	3	0

Therefore, $\gcd(3, 7) = 1$. By Theorem (2.3), it follows that 3 is invertible mod 7.

With this in mind, let's find the Bezout coefficients. We note that the equations we used to find the GCD were

- (Eq. 1) $7 = 3(2) + 1 \implies 1 = 7 - 3(2)$
- (Eq. 2) $3 = 1(3) + 0$

Starting with the last equation with a nonzero remainder, which is Eq. 1, we have

$$7 = 3(2) + 1 \implies 1 = 7 - 3(2).$$

Because we are able to write an equation in terms of 3 and 7, we find that

$$\gcd(3, 7) = 1 = 3(-2) + 7(1).$$

From this, it follows that $x = -2$ and $y = 1$. So, by Theorem (2.3), it follows that -2 is an inverse of 3 (mod 7).

We should note that Bezout coefficients are not unique. If we wanted a positive answer, we note that

$$-2 \equiv 5 \pmod 7$$

so that another possible answer is $\boxed{5}$.

- $a = 41, n = 50$.

First, we calculate $\gcd(41, 50)$.

$\mathbf{a}$	$\mathbf{b}$	$\mathbf{b = aq + r}$	$\mathbf{q}$	$\mathbf{r}$
41	50	$50 = 41q + r$	1	9
9	41	$41 = 9q + r$	4	5
5	9	$9 = 5q + r$	1	4
4	5	$5 = 4q + r$	1	1
1	4	$4 = 1q + r$	4	0

Therefore, $\gcd(41, 50) = 1$. By Theorem (2.3), it follows that 41 is invertible mod 50.

Next, we need to find the Bezout coefficients. We note that the equations we used to find the GCD were

- (Eq. 1) $50 = 41(1) + 9 \implies 9 = 50 - 41(1)$
- (Eq. 2) $41 = 9(4) + 5 \implies 5 = 41 - 9(4)$
- (Eq. 3) $9 = 5(1) + 4 \implies 4 = 9 - 5(1)$
- (Eq. 4) $5 = 4(1) + 1 \implies 1 = 5 - 4(1)$
- (Eq. 5) $4 = 1(4) + 0$

Now, working backwards from the last equation with a nonzero remainder (i.e., Eq. 4):

$$\begin{aligned}
1 &= 5 - 4(1) \\
&= 5 - (\underbrace{9 - 5(1)}_{\text{Eq. 3}})(1) \\
&= 5 - 9 + 5 \\
&= 5(2) - 9 \\
&= (\underbrace{41 - 9(4)}_{\text{Eq. 2}})(2) - 9 \\
&= 41(2) - 9(4)(2) - 9 \\
&= 41(2) - 9(8) - 9 \\
&= 41(2) - 9(9) \\
&= 41(2) - (\underbrace{50 - 41(1)}_{\text{Eq. 1}})(9) \\
&= 41(2) - 50(9) + 41(9) \\
&= 41(11) - 50(9)
\end{aligned}$$

Therefore, we have

$$\gcd(41, 50) = 1 = 41(11) + 50(-9)$$

and so $x = 11$ and $y = -9$. From this, by Theorem (2.3) it follows that $\boxed{11}$ is an inverse of 41 (mod 50).

(Exercise.) Find an inverse of 54 (mod 131), if possible.

Begin by finding the GCD.

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
54	131	$131 = 54q + r$	2	23
23	54	$54 = 23q + r$	2	8
8	23	$23 = 8q + r$	2	7
7	8	$8 = 7q + r$	1	1
1	7	$7 = 1q + r$	7	1

Because $\gcd(54, 131) = 1$, there exists Bezout coefficients and hence an inverse. Note that the equations used to find the GCD were

- (Eq. 1) $131 = 54(2) + 23 \implies 23 = 131 - 54(2)$
- (Eq. 2) $54 = 23(2) + 8 \implies 8 = 54 - 23(2)$
- (Eq. 3) $23 = 8(2) + 7 \implies 7 = 23 - 8(2)$
- (Eq. 4) $8 = 7(1) + 1 \implies 1 = 8 - 7(1)$
- (Eq. 5) $7 = 1(7) + 0$

Starting from Eq. 4 (last operation with a nonzero remainder), we have

$$\begin{aligned}
1 &= 8 - 7(1) \\
&= 8 - (\underbrace{23 - 8(2)})_{\text{Eq. 3}})(1) \\
&= 8 - 23 + 8(2) \\
&= 8(3) - 23 \\
&= (\underbrace{54 - 23(2)}_{\text{Eq. 2}})(3) - 23 \\
&= 54(3) - 23(6) - 23 \\
&= 54(3) - 23(7) \\
&= 54(3) - (\underbrace{131 - 54(2)}_{\text{Eq. 1}})(7) \\
&= 54(3) - 131(7) + 54(14) \\
&= 54(17) - 131(7)
\end{aligned}$$

Therefore, we have

$$\gcd(54, 131) = 54(17) + 131(-7),$$

So, the answer must be 17.

(Exercise.) Solve the following congruences for z.

- $2z \equiv 3 \pmod{11}$

Trivially, $\gcd(2, 11) = 1$. However, let's find the GCD using the Euclidean Algorithm regardless.

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
2	11	$11 = 2q + r$	5	1
1	2	$2 = 1q + r$	2	0

Therefore, the GCD is 1. We can now find the Bezout coefficients. Note that the equations used to find the GCD were

- (Eq. 1) $11 = 2(5) + 1$
- (Eq. 2) $2 = 1(2) + 0$

Starting with the last equation with a nonzero remainder, which is Eq. 1, we have

$$1 = 11 - 2(5).$$

Immediately, it follows that

$$\gcd(2, 11) = 1 = 11(1) + 2(-5).$$

Hence, by Theorem (2.3), $x = -5 \equiv 6 \pmod{11}$ is the inverse of $2 \pmod{11}$.

With this in mind, we now know that

$$\begin{aligned} 2z &\equiv 3 \pmod{11} \\ &\implies 6(2z) \equiv 6(3) \pmod{11} \\ &\implies 12z \equiv 18 \pmod{11} \\ &\implies z \equiv 7 \pmod{11}. \end{aligned}$$

Therefore, the answer is $z \equiv \boxed{7} \pmod{11}$.

- $3z \equiv 2 \pmod 7$

Using the strategy of trial-and-error, we find that $z \equiv 3 \pmod 7$.

- $5z \equiv 3 \pmod{15}$

We note that $\gcd(5, 15) = 5$. Therefore, by Theorem (2.3), there is no solution that satisfies this congruence.

- $5z \equiv 17 \pmod{101}$

First, we want to find $\gcd(5, 101)$. Using the Euclidean Algorithm gives us:

$\mathbf{a}$	$\mathbf{b}$	$\mathbf{b = aq + r}$	$\mathbf{q}$	$\mathbf{r}$
5	101	$101 = 5q + r$	20	1
1	5	$5 = 1q + r$	5	0

Therefore, the GCD is 1. We can now find the Bezout coefficients. Note that the equations used to find the GCD were

- (Eq. 1) $101 = 5(20) + 1 \implies 1 = 101 - 5(20)$
- (Eq. 2) $5 = 1(5) + 0$

Starting with the last equation with a nonzero remainder, which is Eq. 1, we have

$$1 = 101 - 5(20).$$

Immediately, it follows that

$$\gcd(5, 101) = 1 = 101(1) + 5(-20).$$

Hence, by Theorem (2.3), $x = -20 \equiv 81 \pmod{11}$ is the inverse of $5 \pmod{101}$.

With this in mind, we now know that

$$\begin{aligned} 5z &\equiv 17 \pmod{101} \\ &\implies 81(5z) \equiv 81(17) \pmod{101} \\ &\implies 405z \equiv 1377 \pmod{101} \\ &\implies z \equiv 64 \pmod{101}. \end{aligned}$$

Therefore, the answer is $z \equiv \boxed{64} \pmod{101}$.

If we use $x = -20$ instead, we have

$$\begin{aligned} 5z &\equiv 17 \pmod{101} \\ &\implies -20(5z) \equiv -20(17) \pmod{101} \\ &\implies -100z \equiv -340 \pmod{101} \\ &\implies z \equiv -340 \pmod{101} \\ &\implies z \equiv 64 \pmod{101}. \end{aligned}$$

So, in summary, given the congruence $az \equiv b \pmod{n}$, the steps for solving for z are as follows:

1. Find $\gcd(a, n)$. If $\gcd(a, n) \neq 1$, then there are no possible solutions.
2. Find the Bezout coefficients for $\gcd(a, n)$. Specifically, for

$$\gcd(a, n) = ax + ny,$$

   find x (the Bezout coefficients for a). This represents your inverse of $a \pmod{n}$.
3. Multiply both sides of the congruence by x; that is,

$$x(az) \equiv x(b) \pmod{n},$$

   and then simplify.

As you can tell, Bezout coefficients are not unique, and inverses aren't strictly unique either. Notice, for example, that $3(2) \equiv 1 \pmod{5}$ and $8(2) \equiv 1 \pmod{5}$ so that 8 and 3 are both inverses of 2 (mod 5). However, notice that $8 \equiv 3 \pmod{5}$. In other words, inverses are *kind of* unique when they exist: they are unique mod n.

Lemma 2.4

Fix a positive integer n and suppose a is invertible mod n. If x and x' are both inverses of a mod n, then

$$x \equiv x' \pmod{n}.$$

2.6 Affine Cipher

Recall that the encryption function for the Caesar cipher is given by

$$E(x) = (x + b) \pmod{26},$$

where $b = 0, 1, 2, \ldots, 25$ is the shift. Here, x represents the number associated with the letter (e.g., A is 0, B = 1, C = 2, and so on). We can generalize this to the *affine cipher*. Specifically, an **affine cipher** is one whose encryption function is of the form

$$E(x) = (ax + b) \pmod{26},$$

where a and b are integers which form the key.

(Example.) Suppose that $a = 3$ and $b = 5$. The encryption function is defined by

$$E(x) = (3x + 5) \pmod{26}.$$

Suppose we wanted to encrypt the letter `Y`.

Note that the letter `Y` corresponds to the number `24`. So,

$$E(24) = (3 \cdot 24 + 5) \pmod{26} = (72 + 5) \pmod{26} = 77 \pmod{26} = 25.$$

Therefore, the encryption of `Y` is `Z`, which corresponds to 25.

(Exercise.) Use the same encryption function as above with $a = 3$ and $b = 5$.

(a) What is the encryption of `A`?

Note that `A` corresponds to the number `0`. So,

$$E(0) = (3 \cdot 0 + 5) \pmod{26} = 5 \pmod{26}.$$

Here, the number 5 corresponds to the letter `F`.

(b) What is the encryption of `D`?

`D` corresponds to the number `3`, so

$$E(3) = (3 \cdot 3 + 5) \pmod{26} = 14 \pmod{26}.$$

Here, the number 14 corresponds to the letter `O`.

Lemma 2.5: Affine Cipher

Suppose

$$E : \{0, \ldots, 25\} \mapsto \{0, \ldots, 25\}$$

is a function of the form

$$E(x) = (ax + b) \pmod{26}$$

for some integers a and b. Then, there exists a function

$$D : \{0, \ldots, 25\} \mapsto \{0, \ldots, 25\}$$

such that $D(E(x)) = x$ if and only if a is invertible mod 26. Moreover, if $c \equiv a^{-1} \pmod{26}$, then

$$D(y) = c(y - b) \pmod{26}.$$

(Example.) Suppose again $a = 3$ and $b = 5$. Using the process for finding the inverse of a mod 26, we find that this must be 9. So, the Affine Cipher Lemma tells us that the decryption function must be given by

$$D(y) = 9(y - 5) \pmod{26}.$$

Suppose we wanted to decrypt the letter `Z`, which corresponds to the number 25. Then,

$$D(25) = 9(25 - 5) \pmod{26} = 9 \cdot 20 \pmod{26} = 180 \pmod{26} = 24,$$

which corresponds to `Y` as expected.

(Exercise.) Alice and Bob are using the same affine encryption function as above with $a = 3$ and $b = 5$. Bob has just received the message `LNKRLFKH`. Decrypt it.

The letters correspond to the numbers:

$$L \mapsto 11 \qquad N \mapsto 13 \qquad K \mapsto 10 \qquad R \mapsto 17 \qquad F \mapsto 5 \qquad H \mapsto 7.$$

Decrypting each letter results in

- L: $D(11) = 9(11 - 5) \pmod{26} = 9(6) \pmod{26} = 2 \mapsto C$
- N: $D(13) = 9(13 - 5) \pmod{26} = 9(8) \pmod{26} = 20 \mapsto U$
- K: $D(10) = 9(10 - 5) \pmod{26} = 9(5) \pmod{26} = 19 \mapsto T$
- R: $D(17) = 9(17 - 5) \pmod{26} = 9(12) \pmod{26} = 4 \mapsto E$
- F: $D(5) = 9(5 - 5) \pmod{26} = 9(0) \pmod{26} = 0 \mapsto A$
- H: $D(7) = 9(7 - 5) \pmod{26} = 9(2) \pmod{26} = 18 \mapsto S$

Therefore, we have `CUTECATS`, or `cute cats`.

(Exercise.) Suppose the encryption function for an affine cipher is $E(x) = (5x + 17) \pmod{26}$. What is the corresponding decryption function D?

We need to find the inverse of $a = 5 \bmod 26$. So, first, let's find $\gcd(5, 26)$.

a	**b**	$\mathbf{b = aq + r}$	**q**	**r**
5	26	$26 = 5q + r$	5	1
1	5	$5 = 1q + r$	5	1

Since the GCD is 1, there exists an inverse. Moreover, because we only have one equation with a nonzero remainder, it follows that

$$\gcd(5, 26) = 1 = 26(1) + 5(-5).$$

Therefore, the inverse is $-5 \equiv 21 \pmod{26}$. From here, it follows that the decryption function is

$$D(y) = 21(y - 17) \pmod{26}.$$

Remark: Of the numbers between 0 and 25, there are 12 that are invertible mod 26:

$$\{1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, 25\}.$$

So, the number of pairs (a, b) such that $E(x) = ax + b \pmod{26}$ is a legitmate encryption function for an affine cipher is $12 \cdot 26 = 312$.

(Exercise.) The *Atbash cipher* is a simple substitution cipher in which encryption and decryption both simply reverse the order of the alphabet. In other words, `A` and `Z` are interchanged, `B` and `Y` are interchanged, and so forth. For example, the plaintext `APPLE` corresponds to the ciphertext `ZKKOV`. Show that the Atbash cipher is a special case of the affine cipher. What are the corresponding values of a and b?

To see why this is a special case of the affine cipher, we need to understand how the affine cipher works. Consider the encryption function

$$E(x) = (ax + b) \pmod{26}.$$

First, let's set $b = 0$. This way, we just need to try all valid values of a. Notice that, when $a = 25$, we have

- $(25 \cdot 0) \pmod{26} = 0$.
- $(25 \cdot 1) \pmod{26} = 25$.
- $(25 \cdot 2) \pmod{26} = 24$.
- $(25 \cdot 3) \pmod{26} = 23$.
- $(25 \cdot 4) \pmod{26} = 22$.
- $\ldots$
- $(25 \cdot 24) \pmod{26} = 2$.
- $(25 \cdot 25) \pmod{26} = 1$.

This looked very similar to what the Atbash cipher does, albeit with one of the numbers being off (remember that A is supposed to map to Z, but with $a = 25$ and $b = 0$, A maps to A still). However, at that point, it became kind of obvious that if you set $b = -1 \equiv 25$, you'll end up with the correct values of a and b.

(Exercise.)

(a) Make sense of and justify the following statement: "Two affine ciphers in succession result in just another affine cipher."

Consider
$$E_1(x) = (a_1x + b_1) \pmod{26}$$
and
$$E_2(x) = (a_2x + b_2) \pmod{26}.$$
We note that
$$\begin{aligned} E_1(E_2)(x) &= (a_1(a_2x + b_2) + b_1) \pmod{26} \\ &= a_1a_2x + a_1b_2 + b_1 \pmod{26} \\ &= (a_1a_2x) + (a_1b_2 + b_1) \pmod{26}. \end{aligned}$$

(b) Is it possible for "two affine ciphers in succession" to result in a Caesar cipher? Explain.

Consider $a_1 = a_2 = 1$. Then, from the previous part, we'll end up with
$$E_1(E_2)(x) = x + (b_2 + b_1) \pmod{26}.$$
So, it's possible.

2.7 Simple Substitution

We can use a general **simple substitution cipher**, also known as a *simple monoalphabetic substitution cipher* or *monoalphabetic substitution cipher*, by using a full conversion table as a key. For example, we might use a table like the following:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
P	V	J	W	D	C	H	T	S	K	Z	F	N	Q	E	Y	O	R	I	G	A	U	M	L	X	B

This tells us that

- to *encrypt*, we just need to convert every instance of the top letter to the corresponding bottom letter. For example, encrypting A becomes P, encrypting B becomes V, and so on.
- to *decrypt*, we just need to convert every instance of the bottom letter to the corresponding top letter. For example, decrypting P becomes A, decrypting V becomes B, and so on.

(Example.) Suppose Alice wants to encrypt the message `You must destroy all of the horcruxes!` She starts by encoding the message[a]:

`YOUMUSTDESTRYALLOFTHEHORCRUXES`

Then, she converts each letter using the table:

`XEANAIGWDIGREXPFFECGTDTERJRALDI`

This is the ciphertext she sends to Bob. To decrypt the message, Bob uses the same table backwards.

[a]Removing all spaces, punctuations, and then capitalizing everything.

Notice that, if the entire table is our key, the number of possible keys is 26!, a *huge* number. Despite this, simple substitution can still be broken relatively easily using some ideas from probability theory.

(Exercise.) Using the same table given above, do the following by hand.

(a) Encrypt the message `The moon is pitted with holes!`

Encoding the message gives `THEMOONISPITTEDWITHHOLES`. Then, we just need to map each letter appropriately.

```
plaintext  T H E M O O N I S P I T T E D W I T H H O L E S
ciphertext G T D N E E Q S I Y S G G D W M S G T T E F D I
```

The answer is `GTDNEEQSIYSGGDWMSGTTEFDI`.

(b) Decrypt the message `TEMPRDXEAWESQHGEWPX`.

Mapping each letter appropriately gives us

```
ciphertext T E M P R D X E A W E S Q H G E W P X
plaintext  H O W A R E Y O U D O I N G T O D A Y
```

Which, decoded, gives us `How are you doing today?`

2.8 Polybius Square

The **Polybius Square** is another simple substitution cipher which replaces each letter of the plaintext with *two* letters of ciphertext. The idea behind a Polybius square is that it's a table with labeled rows and columns; the alphabet for the messages we're encrypting lives inside the table. For example, if the alphabet we're encrypting includes the capital letters `A` through `Z` and the digits `0` through `9`, then we have 36 letters – perfectly enough to fit in a 6×6 grid. Consider the following arrangement, using the rows and columns `ADFGVX`:

	A	D	F	G	V	X
A	N	A	1	C	3	H
D	8	T	B	2	O	M
F	E	5	W	R	P	D
G	4	F	6	G	7	I
V	9	J	0	K	L	Q
X	S	U	V	X	Y	Z

This table represents our key. To encrypt a message, we convert each letter in the plaintext to a pair of letters indicating the *row* and *column* of that letter in the table above. For example, `K` would be replaced with `VG`. Similarly, `S` would be replaced with `XA`.

(Example.) Suppose Alice wants to encrypt the message

`Storm the gates at 14:37.`

She begins by encoding the message:

`STORMTHEGATESAT1437`

Then, she goes through and replaces each letter by the corresponding pairs as described above:

`XADDDVFGDXDDAXFAGGADDDFAXAADDDAFGAAVGV`

This is the ciphertext. Bob, who knows the table, can undo this process to decrypt the message.

(Exercise.) Use the square given above.

(a) Encrypt the message `Hide tide at 7:01am`.

Encoding the message gives us `HIDETIDEAT701AM`. Then, we can map each individual character in the plaintext to its ciphertext representation:

Plain	**Cipher**
H	AX
I	GX
G	GG
H	AX
T	DD
I	GX
D	FX
E	FA
A	AD
T	DD
7	GV
0	VF
1	AF
A	NN
M	DX

Combining all of this gives us

`AXGXGGAXDDGXFXFAADDDGVVFAFNNDX`

(b) Decrypt the message `XAAAADVGFAFVADDDAXADDDDGFVDX`.

To decrypt, we can map each pair of characters in the ciphertext to its plaintext representation:

Cipher	**Plain**
XA	S
AA	N
AD	A
VG	K
FA	E
FV	P
AD	A
DD	T
AX	H
AD	A
DD	T
DG	2
FV	P
DX	M

Combining and decoding gives us

```
Snake path at 2pm
```