CCSA London Winter Session 2006/2007
Chris Christensen

Simple Substitution Ciphers

The art of writing secret messages – intelligible to those who are in possession of the key and unintelligible to all others – has been studied for centuries. The usefulness of such messages, especially in time of war, is obvious; on the other hand, their solution may be a matter of great importance to those from whom the key is concealed. But the romance connected with the subject, the not uncommon desire to discover a secret, and the implied challenge to the ingenuity of all from who it is hidden have attracted to the subject the attention of many to whom its utility is a matter of indifference.

Abraham Sinkov
In *Mathematical Recreations & Essays*
By W.W. Rouse Ball and H.S.M. Coxeter, c. 1938

We begin our study of cryptology from the romantic point of view – the point of view of someone who has the "not uncommon desire to discover a secret" and someone who takes up the "implied challenged to the ingenuity" that is tossed down by secret writing. The material in this section will help you do the *Quiptoquip* in the morning newspaper and it is excellent preparation for an appearance on the gameshow *Wheel of Fortune*. (And, it will prepare you for our future work.)

A simple substitution cipher is a method of concealment that replaces each letter of a plaintext message with another letter. Here is the key to a simple substitution cipher:

```
Plaintext letters:  abcdefghijklmnopqrstuvwxyz
Ciphertext letters: EKMFLGDQVZNTOWYHXUSPAIBRCJ
```

The key gives the correspondence between a plaintext letter and its replacement ciphertext letter. (It is traditional to use small letters for plaintext and capital letters, or small capital letters, for ciphertext. We will not use small capital letters for ciphertext so that plaintext and ciphertext letters will line up vertically.) Using this key, every plaintext letter a would be replaced by ciphertext E, every plaintext letter e by L, etc. The plaintext message simple substitution cipher would become SVOHTL SAKSPVPAPVYW MVHQLU.

The key above was generated by randomly drawing slips of paper with letters of the alphabet written on them from a bag that had been thoroughly shaken to mix up the slips. The first letter drawn E became the substitution for a, the second letter drawn K became the substitution for b, etc.

Encryption (or enciphering) is the process of using the key to produce ciphertext from plaintext. **Decryption** (or deciphering) is the process of using the key to produce plaintext from ciphertext.

To encrypt a message requires knowing two things: the method of encryption (in our case, simple substitution) and the key (in our case, the letter substitutions). Notice that if we believed that our messages were no longer secure, we could leave the method unchanged (simple substitution) but change the key (use different letter substitutions).

Here is a message to decrypt. It has been encrypted with a simple substitution cipher with key:

Plaintext letters: abcdefghijklmnopqrstuvwxyz
Ciphertext letters: HUFRCOGMTZXLKPNWYVABQSIEDJ

```
BMC  XTP  MHBM  PNBC  NO  HLL  BMHB  BMCD  TPBCPR,
UD  TPBCVFCBTNP  IMTFM  BMCD  RVCHK  PNB  NO.
```

Decrypt the message. Knowing the key, this should not be a problem. Although it might be useful to have the ciphertext letters in alphabetical order for decryption, the key is the same for encryption and decryption.

Plaintext letters: steyxdgawzmlhofnudvibrpkqj
Ciphertext letters: ABCDEFGHIJKLMNOPQRSTUVWXYZ

But, how would a person solve the message not knowing the key? Solving the message not knowing the key is called **cryptanalysis**. Cryptanalysts take up the “implied challenged to the ingenuity” that is tossed down by secret writing, and they find, when successful, satisfaction of their “not uncommon desire to discover a secret.”

Brute Force

If the cryptanalyst knew that the method of encryption were simple substitution cipher, then the cryptanalyst could try all possible keys to solve the message. Or, maybe not! How many keys are possible? How long would it take to try them all?

When constructing a key for a simple substitution cipher, there are 26 choices of letters to substitute for `a`, then 25 remaining letters that can be substituted for `b`, then 24 remaining letters that can be substituted for `c`, etc. This results in

$$26\times 25\times 24\times 23\times ...\times 3\times 2\times 1 = 403{,}291{,}461{,}126{,}605{,}635{,}584{,}000{,}000$$

possible keys. That's a lot of keys.

Now, not all of these would make good choices for a key. One of the choices is plaintext, and others keep many plaintext letters unchanged. If many common plaintext letters remained unchanged, it would not be much of a challenge to cryptanalyze the ciphertext message.

The security of cryptosystems often depends on forcing the cryptanalyst into doing a brute force attack – forcing the crypanalyst to try all possible keys – and “having a large keyspace” – having too many possible keys to making trying them all practical.

> *Cardano [an Italian mathematician, 1501 – 1576] heads a long line of cryptographers in erroneously placing cryptographic faith in large numbers – a line that stretches right down to today. … Cryptanalysts do not solve [simple substitution ciphers] – or any cipher for that matter – by testing one key after another. … If the cryptanalyst tried one of these [403,291,461,126,605,635,584,000,000 possibilities] every second, he [or she] would need*
>
> $$\frac{403{,}291{,}461{,}126{,}605{,}635{,}584{,}000{,}000}{60\times 60\times 24\times 365} \approx 1.2788\times 10^{19} \textit{ years]} \ldots$$

> *to run through them all. Yet most[simple substitution ciphers] are solved in a matter of minutes.* David Kahn, *The Codebreakers: The comprehensive history of secret communication from ancient times to the internet*, Scribner, 1996.

Ok, so it is not a good idea to try to solve one of these by brute force. Would a computer do better? Yes, a computer would do better. Computers now provide an alternative to hand checking of possible keys, but even checking 1000 or 10,000 keys per second wouldn't make a significant dent in the time required to check all possibilities. Brute force attack is just not a good attack. It is certainly not an elegant way of cryptanalysis.

Discovering Patterns

How are simple substitution ciphers attacked? By finding patterns. Every language has rules so that the language "makes sense." There are rules for punctuation, there are rules for combining letters, there is word length, … . These rules create patterns in messages that can be exploited by cryptanalysts. Usually cryptograms that appear in newspapers preserve word length and punctuation, and they preserve letter frequencies. For example, e is the most frequent letter in plaintext English. If we used the key

```
abcdefghijklmnopqrstuvwxyz
EKMFLGDQVZNTOWYHXUSPAIBRCJ
```

we would expect that the most frequent ciphertext letter would be L. Now, it might not be, but it is likely that the most frequent ciphertext letter corresponds to one of the most frequent letters e, t, a, o, i, n, or s. An attack on ciphertext that uses letter frequencies is called **frequency analysis**. Using letter frequencies and other patterns, cryptanalysts are usually able to quickly solve simple substitution ciphers.

Cryptanalysis

Here is a cryptogram that was taken from a local newspaper.

```
D  RNXHT  VHRVCK  VKKXOW  FYVF  V  OVFY

GENBWKKNE'K  PWEC  BVPNEDFW  TWKKWEF  DK  GD.
```

This puzzle is called a *Cryptoquip*. The method used for encrypting it was simple substitution. It obeys the traditional rule for such puzzles that no letter is encrypted as itself. This is very useful information. For example, in this message we know that `PWEC` cannot be the ciphertext for `when`.

If you did this puzzle daily, you would become familiar with the puzzler's writing style. You would know that the plaintext message is a humorous statement. Information about the writing style of the sender or the nature of the plaintext message is often available to cryptanalysts. Use it.

Often cryptogram puzzles give a clue – typically one plaintext/ciphertext correspondence is given. We will attack this message without a clue.

Even though this puzzle might not require all the effort that we will spend on it, we will try to establish a pattern by collecting a great deal of information prior to starting the cryptanalysis.

Here is the information that was gathered about the ciphertext

D RNXHT VHRVCK VKKXOW FYVF V OVFY

GENBWKKNE'K PWEC BVPNEDFW TWKKWEF DK GD.

Most frequent English letters: etaoins

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

The five most frequent letters appear above in **bold**.

1-letter English words: a i
One-letter words: D, V

Most frequently doubled letters in English: setflmo
Doubled letters: K, K, K

Most frequent 2-letter words in English: an, at, as, he, be, in, is, it, on, or, to, of, do, go, no, so, my
Two-letter words: DK, GD

Most frequent 3-letter words in English: the, and, for, was, his, not, but, you, are, her
Three-letter words:

Most frequent initial letters in English: tasoi
V
Initial letters: R V F O P B T D G

Most frequent final letters in English: esdnt
K W F
Final letters: T K W F Y C D

Here’s a cryptanalysis of the message.

We begin with the one-letter words `D` and `V.` `V` is more frequent than `D;` so, it is likely that `V` is `a` and `D` is `i.` Put those in place above the letters of the ciphertext.

Usually we would hunt for a three-letter word that could be `the`, but there are no three-letter words in this *Cryptoquip*.

Notice the `’K`. This suggests that `K` could be `s`. Because `K` is doubled and `K` appears often as a final letter, there is additional information suggesting that `K` is `s`. Put that in place. Additional confirmation that our choice is correct comes from noting that `DK` becomes `is`.

Notice `ass_ _ _` with the final letter being high frequency. This suggests that `X` is `u` and `O` is `m` and `W` is `e`. Put those in place.

Notice `FYaF`. `F` is a high frequency initial and final letter. This is likely to be `that`. Put those letters in place.

We have now identified all the high frequency ciphertext letters other than `E`.

Notice `math _ _ _ _ e s s _ _ ’s`. Doesn’t `math professor’s` just leap out? Put those letters in place.

We still do not seem to have any contradictions.

Everything comes together quickly now:

> `f a _ o r i t e` suggests that `P = v`.
>
> `v e r _` suggests that `C = y`.
>
> `_ e s s e r t` suggests that `T = d`.
>
> `_ o u _ d` suggests that `H = l`.
>
> `a l _ a y s` suggests that `R = w`.

Done! Funny?

Exercises:

1. Cryptanalyze the following cryptoquip:

```
TS F LDRBJ QBFGX AFVJUBJ YFTU, PDMUJ LDM BKZBNX
ATY XD QB RVDPV FG F ADGX-DSSTNB DK?
```

2. Decrypt the following message that was encrypted with a simple substitution cipher and the following key:

Plaintext	`abcdefghijklmnopqrstuvwxyz`
Ciphertext	`YNFROTMKPHELQWBDJXZAUSVCGI`

Ciphertext message:

```
PWMBR   VOAXU   ZAYLL   BAKOX   ZVOQB   WPABX
```

Transposition Ciphers

Up to this point, the ciphers that we have used have been substitution ciphers – plaintext letters were replaced by other letters or numbers or symbols. Another type of cipher is the transposition cipher. Transposition ciphers use the letters of the plaintext message, but they permute the order of the letters.

It should be easy to spot a transposition cipher because the letter frequencies should mimic the usual frequencies for English – high frequencies for `a, e, i, n, o r, s, t`.

But, cryptanalysis of a transposition cipher might be difficult. The essential technique is anagramming – rearranging the ciphertext letters to “make sense.”

The key to the cipher is the pattern of rearrangement.

Jumble

Another word game that appears in newspapers is Jumble. Each weekday Jumble consists of four words with scrambled letters – two five-letter words and two six-letter words – and a picture which has a clever caption that is determined by unscrambling a subset of the letters of the four words. The game is to unscramble the letters and determine the words and the caption. Jumble is solved by anagramming. Here are the four words from the May 31, 2001, *Cincinnati Enquirer* Jumble, unscramble them.

1a. LUGAH
1b. YIXTS
1c. SLIZZE
1d. HIMSUL

The first word LUGAH has five distinct letters. There are $5 \times 4 \times 3 \times 2 \times 1 = 120$ ways to arrange five distinct letters, and exactly one of them should result in a word. A brute force attack would involve trying possible arrangements of the letters until the word were determined; it would take at most 120 trials. A better scheme is to use patterns in the language to put together pieces of the word and arrange the pieces to form the word. For example, a is a common initial letter; so, we might think of a _ _ _ _. It is unlikely that u would be the final letter; so, we might have u surrounded by consonants. That does not seem to work. It is unlikely that either a or u are the final letters; so, they might be surrounded by the consonants. Consonant-vowel-consonant-vowel-consonant seems unlikely for these letters. If consonants form a digraph; it seems most likely that those would be gl (probably at the beginning of the word) or gh (probably at the end of the word). In English, gh is much more common than gl. _ _ _ g h. l _ _ g h. If the vowels form a digraph, it seems likely that it would be au. laugh is the word.

The third word has repeated letters SLIZZE. There are $6 \times 5 \times 4 \times 3 \times 2 \times 1 = 720$ ways to arrange 6 letters. But, it is not possible to distinguish between the two zs. There are 2 ways to arrange 2 letters. If we could tell the two zs apart, there would be 720 ways to arrange the letters, but because we cannot distinguish between them and there are 2 ways to arrange them; the numbers of ways to arrange the 6 letters is 720/2 = 360. e is likely as a final letter: `_ _ _ _ _ e`. Rarely used letters are often easier to place than commonly used ones. `z` combines most frequently with vowels – either vowel –`z` or `z`-vowel. `z` rarely combines with other consonants; if it combines with a consonant, it is likely to combine with another `z`. `nz` or `zl` are next most likely after `zz`. So, maybe `zzl` ending with `e`. `_ zzl _ e` or `_ _ zzle`. `sizzle` works.

Columnar transposition

Columnar transposition is probably the most commonly studied transposition cipher. We will use that method to encrypt the following "pilot's saying:"

```
The nose is pointing down and the houses are getting bigger.
```

There are 49 letters in the message. We want to place the letters of the message in a rectangular array. In this case, because we would like the rectangular array to have 49 cells, a 7×7 array may be used. We also need a keyword having its length the same as the number of columns – we will use *analyst*.

A	N	A	L	Y	S	T
1	4	2	3	7	5	6
t	h	e	n	o	s	e
i	s	p	o	i	n	t
i	n	g	d	o	w	n
a	n	d	t	h	e	h
o	u	s	e	s	a	r
e	g	e	t	t	i	n
g	b	i	g	g	e	r

The ciphertext is obtained by reading down the columns in the order of the numbered columns (which are alphabetically ordered).

```
TIIAOEGEPGDSEINODTETGHSNNUGBSNWEAIEETNHRNROIOHSTG
```

Our message exactly fit the rectangular array. If the message does not completely fill the array, nulls (meaningless letters) may be added to fill it (this is the easier cipher to break) or not (this is harder to break because the columns do not all have the same length). In the latter case, the length of the keyword determines the number of columns, and the number of letters in the message determines the number of complete and partial rows.

The transposition should be applied several times if the plaintext message were longer than 49 letters.

Remember, for encrypting, “in by rows and out by columns.”

Decrypting the columnar transposition

Here is a message that was encrypted using a rectangular array with keyword *analyst*.

```
TRLEELIGCIGEHALANTNCTECYENEN
```

Because the keyword has 7 letters, we know that the rectangular array has 7 columns. The message has 28 letters; therefore, the array must be 4×7. Each column must have 4 entries.

First, we place the letters of the keyword in alphabetical order: *aalnsty*. Then place the ciphertext letters in columns.

A	A	L	N	S	T	Y
t	e	c	h	n	t	e
r	l	i	a	t	e	n
l	i	g	l	n	c	e
e	g	e	a	c	y	n

Now rearrange the letters of the keyword to form *analyst*.

A	N	A	L	Y	S	T
t	h	e	c	e	n	t
r	a	l	i	n	t	e
l	l	i	g	e	n	c
e	a	g	e	n	c	y

The plaintext message is `the central intelligence agency`. (Notice that there could be some ambiguity about which "A" column comes first. We have used the convention that the first "A" column will correspond to the first *a* in *analyst*.)

Remember, decryption reverses the encryption process; so, “in by columns and out by rows” when decrypting.

Cryptanalysis of the columnar transposition

We will do only "the easy case;" i.e., we will assume that the columnar transposition uses a rectangular array that was completely filled.

Here is the ciphertext:

```
     ASAIR  ITFNM  IMTKL  SOIEE  M
```

The “key” to cryptanalyzing the ciphertext is to determine the number of columns; i.e., the length of the keyword. There are 21 letters in the ciphertext. Because we know that the message completely fills the rectangle, this suggests either a 3×7 or a 7×3 array.

We arrange the ciphertext in columns.

Either

A	I	T	M	T	S	E
S	R	F	I	K	O	E
A	I	N	M	L	I	M

or

A	F	L
S	N	S
A	M	O
I	I	I
R	M	E
I	T	E
T	K	M

.

The solution is by anagramming (making a word or portion(s) of word(s) by rearranging letters) a row.

The 7×3 arrangement seems unlikely because it has a string TKM with no vowels that is unlikely. Also, the III is unlikely. So, let us try the 3×7 arrangement. Notice that there are $7\times6\times5\times4\times3\times2\times1=5040$ arrangements of the columns. We would like to not have to try all of them!

A	I	T	M	T	S	E
S	R	F	I	K	O	E
A	I	N	M	L	I	M

In the first row, MATE seems to leap out. This leaves ITS. Perhaps, a slightly wrong guess – ESTIMAT- seems to be a possibility.

Let us rearrange the columns.

E	S	T	I	M	A	T
E	O	K	R	I	S	F
M	I	L	I	M	A	N

Not quite, but there are two Ts in the first row. Let us swap those columns.

E	S	T	I	M	A	T
E	O	F	R	I	S	K
M	I	N	I	M	A	L

This works. Notice that because we have multiple rows that are permuted the same way, we can use multiple anagramming for cryptanalysis.

It is often worthwhile to write the ciphertext in columns, cut out the columns, and rearrange the columns to do the anagramming.

Exercises:

3. Use a columnar transposition cipher with a rectangular array and keyword *mathematician* to encrypt the following message:

```
Sample the electronic environment of the east
coast of North Korea.  Emphasis is intercepting
coastal radars.
```

4. The following message was encrypted with a columnar transposition cipher using a full rectangular array and keyword `mathematics`. Decrypt it.

```
RIUGS   IPNCT   MSPAL   AUNCY   SOOCH   UEYSA
RTE
```

5. Cryptanalyze the following message. It was encrypted with a columnar transposition cipher using a full rectangular array.

```
NTDVC   ILRDT   LFNIT   AUEEE   UEOUA   OVSEN
CIOTN   CCSLS   ATIPN   RNVA
```

Caesar Ciphers

Suetonius, the gossip columnist of ancient Rome, says that [Julius] Caesar [100? – 44 B.C.] wrote to Cicero and other friends in a cipher in which the plaintext letters were replaced by letters standing three place further down the alphabet …

David Kahn, *The Codebreakers*

So, cryptology has existed for more than 2000 years. But, what is cryptology? The word ***cryptology*** is derived from two Greek words: *kryptos*, which means "hidden or secret," and *logos*, which means, "description." Cryptology means secret speech or communication.

Cryptology encompasses two competing skills – concealment and solution.

The concealment portion of cryptology is called ***cryptography***. The aim of cryptography is to render a message incomprehensible to the unauthorized reader. Cryptography is often called “code making.”

The solution portion of cryptology is called ***cryptanalysis***. Cryptanalysis is often called “code breaking.” The word cryptanalysis was coined (c. 1920) by the American cryptologist William Friedman.

William Friedman
Center for Cryptologic History photo

Friedman (1891 – 1969) is often called the dean of modern American cryptologists. He was a pioneer in the application of scientific principles to cryptology. During World War II, Friedman was the director of communications research for the Signal Intelligence Service (SIS). SIS later became the Army Security Agency (ASA). After World War II, Friedman served first as a consultant for ASA and then for the National Security Agency (NSA) after its birth in 1952. Friedman and his wife Elizebeth, who was also a cryptologist, jointly authored the book *The Shakespearean Ciphers Examined*.

Cryptography of Caesar Ciphers

Here is the key for a simple substitution cipher:

Plaintext letters:	`abcdefghijklmnopqrstuvwxyz`
Ciphertext letters:	`YNROTKMCPBDVXZALEWUSFQJHGI`

Could you remember the plaintext/ciphertext correspondences? Probably not; you would probably need a written copy of the key. But, having a written copy of the key could lead to problems with key security – the key might be lost or stolen. It is desirable to have a key that need not be written down. (Of course a person who has memorized the key might be coerced to give it up, but that it a different story.)

Caesar's cipher, to which reference was made in the David Kahn quote at the beginning of this section, was a simple substitution cipher, but it had a memorable key. For Caesar's cipher, "letters were replaced by letters standing three place further down the alphabet … ." Here is the key to Caesar's cipher:

Plaintext letters	`abcdefghijklmnopqrstuvwxyz`
Ciphertext letters	`DEFGHIJKLMNOPQRSTUVWXYZABC`

The key can be memorized because there is a pattern to it -- the ciphertext alphabet is just the plaintext alphabet shifted to the right three places. Sender and receiver just need to remember the shift.

Mathematics of the Caesar Cipher

The mathematical transformation that shifts the alphabet is called a translation. The shift to the right of three spaces can be symbolized as $\mathrm{C} = \mathrm{p} + 3$ where p represents a plaintext letter and C represents the corresponding ciphertext letter. More generally, a shift of *b* spaces to the right can be symbolized by $\mathrm{C} = \mathrm{p} + b$. The Caesar cipher can be described as C = p + key.

Of course, to make sense of this transformation, first, we must number the letters of the alphabet. Computer scientists would probably prefer a = 00, …, z = 25. There are also mathematical reasons to prefer this numbering, but we will use the more naive

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
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 that we must make provision for "falling off the end of the alphabet"; e.g. with a shift of 3, what happens to plaintext x when we shift 3 places to the right? We do "the obvious" – we wrap back to the beginning of the alphabet.

Plaintext letters	abcdefghijklmnopqrstuvwxyz
Ciphertext letters	DEFGHIJKLMNOPQRSTUVWXYZABC

a, which is represented by 01, is mapped to 01 + 3 = 04, which represents D. a → D, b → E, c → F, etc. When we come to the end of the plaintext alphabet, the ciphertext alphabet returns to the beginning: w , which is represented by 23, is mapped to 23 + 3 = 26, which represents Z: w → Z; but, x, which is represented by 24, is mapped to 24 + 3 = 27, which wraps back to A: x → A. y → B, and z → C. This is called addition modulo 26.

Modular addition operates on the integers: …, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, … . The symbol $\mathbb{Z}$ is often used to represent the integers, the "counting numbers" (*zählen* is German for *to count*.) When we add modulo *n*, (or mod *n*) where *n* is a positive integer, we add "in the usual way," and then we divide by *n* and take the remainder. If we divide by *n*, the remainder after division can be represented by one of 0, 1, 2, …, *n*-1. *n* is called the modulus.

> If we divide by 5, the possible remainders are 0, 1, 2, 3, 4.
>
> If we divide by 8, the possible remainders are 0, 1, 2, 3, 4, 5, 6, 7.
>
> If we divide by 12, the possible remainders are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11.
>
> If we divide we 26, the possible remainders are 0, 1, 2, … , 23, 24, 25.
>
> If we divide by 2, the possible remainders are just 0 and 1.

Selecting the remainders is where the problems occur with our numbering of the letters of the alphabet beginning with 1. We would like each of the letters of the alphabet to correspond to a remainder modulo 26: 0, 1, 2, … , 23, 24, 25. But, we have chosen to number the letters 1, 2, 3, … , 24, 25, 26. There is a way to make this work. Mathematicians say that two integers are *congruent modulo* 26 if they have the same remainder when we divide them by 26. 0 and 26 are congruent modulo 26, 1 and 27 are congruent modulo 26, 2 and 28 are congruent modulo 26, 3 and 29 are congruent modulo 26, etc. We treat congruence as the "modular equivalent of equals." Because 0 and 26 are congruent modulo 26 we say that they are equal mod 26: 0 = 26 mod 26. 0, 1, 2, … , 23, 24, 25 are the usual representatives of the integers modulo 26, but we have chosen instead to take the equivalent set of representatives: 1, 2, 3, … , 24, 25, 26. In either case, each of the 26 possible remainders is represented once. Other sets of representatives of the integers modulo 26 are also possible (e.g., 0, 1, 28, -23, 4, 57, -46, 7, 8, 9, 10, -15, -2757, 13, 14, -11, 68, 17, 18, 19, 20, -5, 22, -3, 24, 25), but these representatives would not have an obvious meaning for ciphers.

So, when we take plaintext w (23) and shift it by 3 (23 + 3) we get 26 = 0 mod 26; we get 26 (Z). When we take plaintext x (24) and shift it by 3 (24 + 3) we get 27 = 1 mod 26; we get 1 (A). When we take plaintext y (25) and shift it by 3 (25 + 3) we get 28 = 2 mod 26; we get 2 (B). When we take plaintext z (26) and shift it by 3 (26 + 3) we get 29 = 3 mod 26; we get 3 (C). This reflects the "wrapping back" to the beginning of the alphabet that occurs in the plaintext/ciphertext correspondence – the turning around “to bite its tail.”

Addition modulo 12 is just “clock arithmetic.” The remainders are taken to be 1, 2, 3, … , 12 – the hours of the day. When we divide by 12 and take remainders, we get that 13 = 1 mod 12 (13 o’clock is 1 pm), 14 = 2 mod 12 (2 pm), … , 23 = 11 mod 12 (11 pm), and 24 = 12 mod 12 (midnight).

Addition modulo 2 has remainders of 0 or 1. Every even integer has remainder 0, and every odd integer has remainder 1. Every even integer is represented by 0, and every odd integer is represented by 1.

1, 2, 3, … , 24, 25, 26 (or 0, 1, 2, 3, … , 24, 25 or any other set of representatives of the remainders) under addition modulo 26 is written as $\mathbb{Z}_{26}$. 0, 1, 2, … , 10, 11 under addition modulo 12 is written as $\mathbb{Z}_{12}$. 0 and 1 under addition modulo 2 is written as $\mathbb{Z}_2$. In general, 0, 1, 2, … , $n - 1$ under addition modulo n is written as $\mathbb{Z}_n$.

Encryption of a Message with a Caesar Cipher

Let us use the Caesar cipher with additive key 5 to encrypt the plaintext message:

> The book *Gadsby* by Ernest Vincent Wright does not contain the letter *e*.

Giving word length and punctuation gives the cryptanalyst too much information. It is usually easy to solve simple substitution ciphers when word length and punctuation are given, it can be very difficult to solve simple substitution ciphers when word length and punctuation are not give.

Word length and punctuation provide patterns that permit us to quickly make sense of plaintext. Without word length and punctuation, even plaintext can be difficult to read. Here is an example of plaintext without word length and punctuation:

> CARDANOALSOACHIEVEDTHEDUBIOUSRENOWNOFBEING
> THEFIRSTCRYPTOLOGISTTOCITETHEENORMOUSNUMBERO
> FVARIATIONSINHERENTINACRYPTOGRAPHICSYSTEMASPR
> OOFOFTHEIMPOSSIBILITYOFACRYPTANALYSTSEVERREAC
> HINGASOLUTIONDURINGHISLIFETIME.

Usually cryptographers do not give word length and punctuation.

After the invention of the telegraph in the Nineteenth Century, nearly instantaneous communication over long distances became possible, but communication by telegraph involved handing messages to operators who transmitted them in Morse Code. Both the sending and receiving telegraph operators (and probably other telegraph employees) would have access to messages. Business communications and even personal communications were often encrypted. For the convenience of telegraph operators, messages were usually sent in blocks which allowed momentary pauses for the operators’ hands. Traditionally the blocks consisted of four or five letters. That practice became a tradition in cryptology. Often ciphertext messages are blocked in blocks of four or five letters. (We will use five-letter blocks.)

Decryption of a Message Encrypted with a Caesar Cipher

Certainly it would be necessary to be able to decrypt any message that has been encrypted. Mathematically, what we are requiring is that every encryption method should have an inverse method. We are requiring that for each possible key there must an inverse.

What undoes addition mod 26? Well, subtraction mod 26, but subtraction is just “adding the additive inverse.” What undoes addition of 3 mod 26 is addition of 23 mod 26 because 3 + 23 = 26 = 0 mod 26. If we shift to the right by 3 and then by 23, we have shifted to the right be 26 and returned to plaintext.

$$\text{plaintext} \xrightarrow{+3 \text{ mod } 26} \text{CIPHERTEXT} \xrightarrow{+23 \text{ mod } 26} \text{plaintext}$$

So, the pattern should be clear

Key	Additive inverse
1	25
2	24
3	23
…	…
23	3
24	2
25	1
26	0

If a message were encrypted with a Caesar cipher with additive key 5

Plaintext letters	abcdefghijklmnopqrstuvwxyz
Ciphertext letters	FGHIJKLMNOPQRSTUVWXYZABCDE

then shifting ciphertext 21 letters further would yield plaintext.

Ciphertext letters	ABCDEFGHIJKLMONPQRSTUVWXYZ
Plaintext letters	vwxyzabcdefghijklmnopqrstu

Cryptanalysis Using Brute Force

Unfortunately, Caesar ciphers have a small key space, and messages encrypted with Caesar ciphers can be easily broken by brute force if it is recognized that the message has been encrypted with a Caesar cipher.

How many distinct Caesar ciphers are possible? Well, a shift of 0 would not make any sense; we would still have plaintext. Shifts of 1, 2, 3, … 25 make sense. But, a shift of 26 would (because the alphabet returns to the beginning) be the same as a shift of 0. Similarly, a shift of 27 is the same as a shift of 1, a shift of 28 is the same as a shift of 2, etc. So, there are only 26 possible Caesar ciphers, and one of those is a shift of 0 which would provide no encryption at all.

Notice that with the exception of the Caesar cipher with additive key 26, when using a Caesar cipher, no letter substitutes for itself. Also, if we know one plaintext/ciphertext correspondence we know them all because the shift is the same for each letter.

Because of the small number of possible keys, a brute force attack is possible – we could try all possible keys and see which one yields plaintext.

Here is a brute force ciphertext attack on a Caesar cipher.

The following message is known to have been encrypted with a Caesar cipher:

```
VRRQS  HRSOH  EHJDQ  VOLGL  QJWKH  DOSKD  EHWEB
DPRXQ  WVGLI  IHUHQ  WWKDQ  WKUHH  WRGHW  HUPLQ
HFLSK  HUHTX  LYDOH  QWV
```

Begin with VRRQS, the first five-letter block of the ciphertext. Now beneath it write the five letters that would result by shifting each of the cipehrtext letters to the right by one. On the next line, write the result by shifting each of the ciphertext letters to the right by two. Do this for each of the 26 possible shifts. This attack on a Caesar cipher is sometimes called “running the alphabet.”

```
VRRQS
WSSRT
XTTSU
YUUTV
ZVVUW
AWWVX
BXXWY
CYYXZ
DZZYA
EAAZB
FBBAC
GCCBD
HDDCE
IEEDF
JFFEG
KGGFH
LHHGI
MIIHJ
NJJIK
OKKJL
PLLKM
QMMLN
RNNMO
SOONP
TPPOQ
UQQPR
```

Now scan the column for something that makes sense. Notice near the bottom SOONP. This line corresponds to shifting the ciphertext alphabet to the right 23 places. The key inverse is 23. The additive key is 3.

Cryptanalysis Using a Known Plaintext Attack

Another possibility is to do a **known plaintext attack**. The name is a bit deceiving because sometimes we only “suspect” rather than “know” a piece of the plaintext message. Consider that in a message of reasonable length we should expect to find the word `the`. If it occurs in a message encrypted with a Caesar cipher, it was encrypted one of the following ways:

Trigraph	Shift
THE	0
UIF	1
VJG	2
WKH	3
XLI	4
YMJ	5
ZNK	6
AOL	7
BPM	8
CQN	9
DRO	10
ESP	11
FTQ	12
GUR	13
HVS	14
IWT	15
JXU	16
KYV	17
LZW	18
MAX	19
NBY	20
OCZ	21
PDA	22
QEB	23
RFC	24
SGD	25

Here is a message that is known to have been encrypted with a Caesar cipher:

```
FGWFM  FRXNS  PTAKN  WXYBT  WPJIF  XFHWD  UYTQT
LNXYB  NYMYM  JBFWI  JUFWY  RJSY
```

To determine the key, search through the ciphertext for a Caesar cipher ciphertext of `the`. Because the beginning and ending of words is hidden by the five-letter blocks, when searching for an encrypted `the`, we must check every three consecutive letters – every trigraph:

FGW GWF WFM FMF MFR FRX RXN XNS NSP SPT PTA TAK AKN
KNW NWX WXY XYB YBT BTW TWP WPJ PJI JIF IFX FXF XFH
FHW HWD WDU DUY UYT YTQ TQT QTL TLN LNX NXY XYB YBN
BNY NYM MYM **YMJ** MJB JBF BFW FWI WIJ IJU JUF UFW FWY
WYR YRJ RJS JSY.

The trigraph in bold is `the` encrypted with an additive key of 5. If we assume the message was encrypted with an additive key of 5, the message decrypts.

This technique of searching for an encrypted version of a word or phrase was used during World War II by the British codebreakers at Bletchley Park who broke the German Enigma messages. The Enigma machine had letters but no numbers on its keyboard; so, numbers were written out in plaintext messages. It was common that the word *Eins* (one) would appear in a message. With a lot of patience and having a catalog of the encrypted versions of *Eins*, the Enigma key might be determined.

The word `the` when used as we have in this process is called a *crib*. Gordon Welchman, one of the cryptologists at Bletchely Park writes:

> *Cryptologically speaking, however, one has a "crib" to a cipher text if one can guess the clear text from which some specific portion of the cipher text was obtained. As my analysis of the Enigma traffic began to reveal certain routine characteristics in the preambles of individual messages, I realized that, if we could somehow determine to whom they were addressed, or by whom they were sent, we might be able to guess a portion of the clear text either at the beginning or the end of each of the messages, and so have cribs.* Gordon Welchamn, *The Hut Six Story*

Stereotyped writing provides cribs. In cryptography, variety breeds security.

Recognition of a Caesar Cipher and Its Key by Frequency Analysis

A Caesar cipher is easy to break, but how do we recognize that a Caesar cipher was used? It is easy to spot a Caesar cipher from frequency analysis of the ciphertext.

Patterns occur in the letter frequencies of any language. Here are the patterns for English:

Frequencies for English

a	1111111
b	1
c	111
d	1111
e	1111111111111
f	111
g	11
h	1111
i	1111111
j	
k	
l	1111
m	111
n	11111111
o	1111111
p	111
q	
r	11111111
s	111111
t	111111111
u	111
v	1
w	11
x	
y	11
z	

Abraham Sinkov (who was one of William Friedman's cryptanalysts suring World War II) in his text *Elementary Cryptanalysis: A Mathematical Approach* points out the following patterns which are useful for elementary cryptanalysis:

1. `a`, `e`, and `i` are all high frequency letters (at the beginning of the plaintext alphabet), and they are equally spaced (four letters apart) with `e` the most frequent.
2. `n` and `o` form a high frequency pair (near the middle of the plaintext alphabet).
3. `r`, `s`, and `t` form a high frequency triple (about 2/3 of the way through the plaintext alphabet).
4. `j` and `k` form a low frequency pair (just before the middle of the plaintext alphabet).
5. `u`, `v`, `w`, `x`, `y`, and `z` form a low frequency six-letter string (at the end of the plaintext alphabet).

Because a Caesar cipher just translates the letters of the plaintext alphabet to the right, it translates the frequency patterns we expect with plaintext.

Here is a ciphertext message:

```
VRRQS  HRSOH  EHJDQ  VOLGL  QJWKH  DOSKD  EHWEB
DPRXQ  WVGLI  IHUHQ  WWKDQ  WKUHH  WRGHW  HUPLQ
HFLSK  HUHTX  LYDOH  QWV
```

Here is a frequency analysis of the ciphertext:

A	
B	1
C	
D	111111
E	111
F	1
G	111
H	111111111111111
I	11
J	11
K	11111
L	111111
M	
N	
O	1111
P	11
Q	11111111
R	11111
S	1111
T	1
U	1111
V	1111
W	111111111
X	11
Y	1
Z	

Notice that the pattern of frequencies suggests that `H = e`. It is only necessary to determine one correspondence between a plaintext and ciphertext letter to determine the key. The frequency patterns suggest that `H = e`; so, 8 = 5 + additive key. The additive key is 3.

Exercises

6. Encrypt the following message using a Caesar cipher with additive key 9. Use five-letter blocks.

```
The telegraph made cryptography what it is today.
```

7. The following message was encrypted using a Caesar cipher with additive key 9. Decrypt the message.

```
KUNCL QUNHY JATRB OXACH VRUNB WXACQ XOUXW MXW
```

8. Use frequency analysis to cryptanalyze the following ciphertext:

```
MAXGX   QMWTR   VKXPF   XFUXK   LHGMA   XFTWW
HQLBZ   AMXWY   BOXGH   KMAOB   XMGTF   XLXGT
ORTMM   TVDUH   TML
```

9. The following was enciphered with a Caesar cipher. By running the alphabet on the first 5-letter block determine the shift and decipher the message.

```
dvysk dhyad vthyr zhjoh unlpu jyfwa vsvnf hsaov
bnoao lylhy lleht wslzv mthao lthap jphuz zabkf
punjv klzhu kjpwo lyzao yvbno vbaop zavyf dvysk
dhyad vthyr zaolw vpuah adopj ojpwo lyjby lhbzi
lnhua vyljy bpath aolth apjph uzmvy aolpy wyvis
ltzvs cpunh ipspa plz
```

10. Search through the following ciphertext that is known to have been encrypted with a Caesar cipher and find an encrypted version of the word `the`. Determine the key and recover the plaintext.

```
IBYBC   KBSJS   BHCRO   MWGHV   SGDMK   OFHVO
HFOUS   ROHHV   SHCDC   THVSK   CFZR
```

Cryptography of the Vigenère Cipher

Cryptanalysis is based upon finding the ghosts of patterns of the plaintext. We have seen that an important technique of doing this is frequency analysis. So, cryptographers try to develop ciphers that are not easily attacked by frequency analysis. There are two basic ways to do this: use more than one ciphertext alphabet or encrypt more than one letter in a block. First, we will consider using more than one cipher text alphabet.

Simple substitution ciphers, Caesar ciphers, multiplicative ciphers, and affine ciphers are all examples of **monoalphabetic ciphers** – only one ciphertext alphabet is used.

> Even if the original word lengths are concealed and the substitution alphabet is random, it is possible to find a solution by using frequency data, repetition patterns and information about the way letters combine with one another. What makes the solution possible is the fact that a given plain language letter is always represented by the same cipher letter. As a consequence, all the properties of plain language such as frequencies and combinations are carried over into the cipher and may be utilized for solution. In effect we could say that all such properties are invariant except that the names of the letters have been changed.
>
> It would seem then that one way to obtain greater security would be to use more than one alphabet in enciphering a message. The general system could be one that uses a number of different alphabets for encipherment, with an understanding between correspondents of the order in which the alphabets are to be used. Sinkov, Abraham, *Elementary Cryptanalysis: A mathematical approach*, Mathematical Association of America, 1968.

A simple scheme would be to have two cipher alphabets and alternate between them during encryption. Such a scheme is an example of a **polyalphabetic cipher** a cipher in which there is more than one ciphertext alphabet and a rule that describes how to use them. For example, our ciphertext alphabets might be a Caesar cipher with additive key 3 and a Caesar cipher with additive key 5. Our enciphering rule is that we will use the Caesar cipher alphabet with additive key 3 to encrypt the first plaintext letter, the Caesar cipher alphabet with additive key 5 to encrypt the second plaintext letter, the Caesar cipher alphabet with additive key 3 to encrypt the third plaintext letter, the Caesar cipher alphabet with additive key 5 to encrypt the fourth plaintext letter, etc. Our rule is to alternate between the two alphabets beginning with the Caesar cipher with additive key 3.

For example, we will encrypt the plaintext message Northern Kentucky University:

The key

```
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

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 C

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

Plaintext and ciphertext

```
n o r t h e r n k e n t u c k y u n i v e r s i t y
Q T U Y K J U S N J Q Y X H N D X S L A H W V N W D
```

Notice that two of the ns are encrypted with Q and two with S. Two rs are encrypted with U and one with W. Two ts are encrypted with Y and one with W. Etc. But, for example, because of the spacing of the plaintext letters, both of the ys are encrypted as D.

For the inverse process – decryption – there are two Hs, but one has been substituted for plaintext c and the other for plaintext e.

Because two ciphertext letters correspond to each plaintext letter, this scheme will tend to balance frequencies, and it is memorable.

We might balance frequencies even better if we have several cipher alphabets and rotate among them according to some scheme to which the correspondents have agreed. We will examine a classic example of such a method – the Vigenère cipher.

The Cryptographer:
Blaise de Vigenère (1523 – 1596)

> Vigenère was not a nobleman. The "de" in his name simply indicates that his family came from the village of Vigenère or Viginaire. He himself was born in the village of Saint-Pourçain, about halfway between Paris and Marseilles, on April 15, 1523. At 17, he was taken from his studies and sent to court and, five years later, to the Diet of Worms as a very junior secretary. This gave him an initiation into diplomacy, and his subsequent travels through Europe broadened his experience. At 24, he entered the service of the Duke of Nevers, to whose house he remained attached the rest of his life, except for periods at court and as a diplomat. In 1549, at 26, he went to Rome on a two-year diplomatic mission.
>
> It was here that he was first thrown into contact with cryptology, and he seems to have steeped himself in it. He read the books of Trithemius [1462 – 1516], Belaso [? - ? but known to have published in 1553 a booklet in which he proposed a polyalphabetic cipher], Cardano [1501 – 1576], and Porta [1535 – 1615], and the unpublished manuscript of Alberti [1404 – 1472]. He evidently conversed with the experts of the papal curia … . … in 1566 he was sent again to Rome as secretary to King Charles IX. Here he renewed his acquaintance with the cryptographic experts, and this time seems to have been admitted to their chambers … . Finally in 1570, at 47, Vigenère quit the court for good, turned over his annuity of 1,000 livres a year to the poor of Paris, married the much younger Marie Varé, and devoted himself to writing.
>
> He turned out some 20-odd books before he died of throat cancer in 1596. … [The] book which is constantly cited by workers in its field is his *Traicté des Chiffres*, which was written in 1581 … .
>
> It is a curious work. In its more than 600 pages, it distilled not only much of the cryptographic lore of Vigenère's day … but a hodgepodge of other topics.

> … [The] *Traicté* is reliable in its cryptographic information. Vigenère was scrupulous in assigning credit for material from other authors and quoted them accurately and with comprehension.
>
> Among the numerous ciphers that Vigenère discussed … were polyalphabetics. Each of his used a Trithemius-like tableau [which he improved by adding mixed alphabets on the sides. He also improved upon the autokey system of Cardano. Vigenère's system] works well and affords fair guarantees of security … .
>
> In spite of Vigenère's clear exposition of his devices, both were entirely forgotten and only entered the stream of cryptology late in the 19th-Century after they were reinvented. Writers on cryptology then added insult to injury by degrading Vigenère's system into one more elementary.
>
> This system is … more susceptible to solution than Vigenère's original. Nevertheless, a legend grew up that this degenerate form of Vigenère's work was the indecipherable cipher par excellence, a legend so hardy that as late as 1917, more than a half century after it had been exploded, the Vigenère was touted as "impossible of translation" in a journal as respected as *Scientific American.* Kahn, David, *The Codebreakers: The comprehensive history of secret communication from ancient times to the internet*, Scribner, 1996.

The method we shall study below is the corrupted version of the cipher that now bears Vigenère 's name. His original cipher was more secure than this.

The Vigenère Square

The Vigenère cipher is based upon a square that consists of the 26 Caesar cipher alphabets; this is in fact the square used by Trithemius [1462 – 1516].

```
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
BCDEFGHIJKLMNOPQRSTUVWXYZA
CDEFGHIJKLMNOPQRSTUVWXYZAB
DEFGHIJKLMNOPQRSTUVWXYZABC
EFGHIJKLMNOPQRSTUVWXYZABCD
FGHIJKLMNOPQRSTUVWXYZABCDE
GHIJKLMNOPQRSTUVWXYZABCDEF
HIJKLMNOPQRSTUVWXYZABCDEFG
IJKLMNOPQRSTUVWXYZABCDEFGH
JKLMNOPQRSTUVWXYZABCDEFGHI
KLMNOPQRSTUVWXYZABCDEFGHIJ
LMNOPQRSTUVWXYZABCDEFGHIJK
MNOPQRSTUVWXYZABCDEFGHIJKL
NOPQRSTUVWXYZABCDEFGHIJKLM
OPQRSTUVWXYZABCDEFGHIJKLMN
PQRSTUVWXYZABCDEFGHIJKLMNO
QRSTUVWXYZABCDEFGHIJKLMNOP
RSTUVWXYZABCDEFGHIJKLMNOPQ
STUVWXYZABCDEFGHIJKLMNOPQR
TUVWXYZABCDEFGHIJKLMNOPQRS
UVWXYZABCDEFGHIJKLMNOPQRST
VWXYZABCDEFGHIJKLMNOPQRSTU
WXYZABCDEFGHIJKLMNOPQRSTUV
XYZABCDEFGHIJKLMNOPQRSTUVW
YZABCDEFGHIJKLMNOPQRSTUVWX
ZABCDEFGHIJKLMNOPQRSTUVWXY
```

The Cipher

The key to this method of encryption is a memorable word or phrase. Let us use the name of the French mathematician *Galois* (1811 – 1832) as our key to encipher `Northern Kentucky University`.

The letters of the keyword determine the alphabets used to encrypt:

> The first letter of the keyword is *g*; so, the first letter of the message is encrypted using row *g* of the table. Plaintext `n` corresponds to ciphertext `T`.
>
> The second letter of the keyword is *a*; so, the second letter of the message is encrypted using row *a* of the table. Row *a* corresponds to a shift of 0 – plaintext; so, plaintext `o` corresponds to ciphertext `O`.
>
> The third letter of the keyword is *l*; so, the third letter of the message is encrypted using row *l* of the table. Plaintext `r` corresponds to ciphertext `C`.
>
> The fourth letter of the keyword is *o*; so, the fourth letter of the message is encrypted using row *o* of the table. Plaintext `t` corresponds to ciphertext `H`.
>
> The fifth letter of the keyword is *i*; so, the fifth letter of the message is encrypted using row *i* of the table. Plaintext `h` corresponds to ciphertext `P`.
>
> The sixth and last letter of the keyword is *s*; so, the sixth letter of the message is encrypted using row *s* of the table. Plaintext `e` corresponds to ciphertext `W`.
>
> Now we returned to the beginning of the keyword. The first letter of the keyword is *g*; so, the seventh letter of the message is encrypted using row *g* of the table. Plaintext `r` corresponds to ciphertext `X`.
>
> Etc.

Here are the keyword, plaintext, and ciphertext messages:

```
g a l o i s g a l o i s g a l o i s g a l o i s g a
n o r t h e r n k e n t u c k y u n i v e r s i t y
T O C H P W X N V S V L A C V M C F O V P F A A Z Y
```

Notice that the four ns are encrypted as T, N, V, and F. The three rs are encrypted as C, X, and F. The three ts are encrypted as H, L, and Z.

But, notice that because of the spacing of the plaintext letters, the two ks are each encrypted with row *l* as V.

For the inverse process – decryption – A represents u, s, and i. V represents both k and v.

Etc.

Ideally, a different alphabet could be used to encrypt each letter of the plaintext message. (Of course, there are only 26 possible shifts.)

Both the sender and receiver of a message need a Vigenère square. So, it is possible that someone could discover the method of encryption. But, the keyword need not be written; so, the key can remain secure even if the method is known.

Rotating among the alphabets tends to equalize the frequencies of ciphertext letters and makes frequency analysis more challenging (but not impossible). The more alphabets that are used (i.e., the longer the keyword or phrase) the more the frequencies can be equalized.

Here is a plaintext message:

```
It is all but impossible to draw a distinction between Bletchley Park’s
work on wartime Germany and its growing work on the Soviet Union in the
nineteen forties.  Knowledge of wartime Germany required the tracking
of events on the eastern front and involved learning as much as
possible about the Soviet effort.  British intelligence began to value
the Germans for their knowledge of the Soviet Union as soon as Ultra
came onstream.  German messages used to send their own Sigint summaries
about the Soviet Union back to Berlin were, in turn, intercepted by the
British.  This “second-hand” Signit proved to be London’s best source
on the performance of Soviet forces.  As early as nineteen forty-three
the Joint Intelligence Committee – Britains’ highest intelligence
authority – was able to produce detailed and accurate reports on the
capabilities of the Soviet Air Force, based upon Luftwaffe Sigint
material.
```

After encrypting it with a Vigenère cipher using the keyphrase *Northern Kentucky Univeristy*, the ciphertext message is:

```
vhzlh pcoex vfjqc qcotz xfvzt unrzl amepd mbgvg duyrv wpvlk ajrmg tyojj
yvxhh ykpnv uzkvj utllo ewpxj tbsjb higml xwixy wahtv sknum fasrg aypsl
ygmzr wgzmg rgbgv acrnk rhzyk pnvuz kvjut llfvj bmirn xuxnt kaevv bswwd
xlggf galvr kwgxl pppia bvrua vomyj vwsir exmaz uuwsw uintf kabzy sruvy
kgrif hpkor ysnjv ktzbr vgybu xvyvm txheo zytii xfnie srhyx niizk rfyit
dfyvz frfot xbtsf yalvf yzvxn wxgia inwfg vtqhz kkhgr zosal ntoyg tmmqr
fuxqf oxxzy jrnxb lypnr brqms nfabe vbklb qdnbm rludy sngpz wfnqx rhbzh
ufrpu xbuyt vghjm mizfb npawe mlvtr zxrwv adfyo zdxzk pmfvg jxjse qreaw
mkqlc gxmsm wlmmo schuh facfr lnuys lpmjr kzmic etfkt eepos slixs cnswm
gvkil cnfcr hwevx igxyp pmlgg oliwm mfrxf buxza diyec iolwr kjqda bmcrp
ibaez aclvz bgcrc abzpc aoxlp srnal fesxl puukz frbjt iglna rrvmh mcrne
awuem slnbz vvhwk rfcem oitnz eobfk dgyfw axywa htvsk tpvwb bgruu uoboc
wiplx bpyst vlpkz adqnm ytsyf
```

(This message was encrypted with software that may be downloaded from http://faculty.goucher.edu/blewand/cryptomath/)

Here are the plaintext frequencies:

```
a     1111111111111111111111111111111111111111111111111
b     1111111111111111111
c     11111111111111111111
d     11111111111111111111
e     11111111111111111111111111111111111111111111111111111111111111111
      1111111111111111111111111111
f     11111111111111111
g     1111111111111111111
h     1111111111111111111111111
i     1111111111111111111111111111111111111111111111111111111111111111
j     1
k     1111111
l     11111111111111111111111111
m     11111111111111111
n     1111111111111111111111111111111111111111111111111111111111111111
o     11111111111111111111111111111111111111111111111111111111111
p     1111111111
q     1
r     111111111111111111111111111111111111111111111111
s     11111111111111111111111111111111111111111111111111
t     11111111111111111111111111111111111111111111111111111111111111111
      11111111111
u     1111111111111111111
v     11111111111
w     1111111111111
x
y     1111111
z
```

Here are the ciphertext frequencies:

```
A     11111111111111111111111111111111
B     11111111111111111111111111111111
C     11111111111111111111
D     111111111111
E     1111111111111111111111
F     11111111111111111111111111111111111
G     11111111111111111111111111111111
H     111111111111111111111
I     111111111111111111111111111
J     111111111111111111
K     1111111111111111111111111111
L     11111111111111111111111111111111111
M     111111111111111111111111111111111111
N     111111111111111111111111111111
O     111111111111111111111
P     1111111111111111111111111111
Q     111111111111
R     11111111111111111111111111111111111111111111
S     1111111111111111111111111111
T     1111111111111111111111111111
U     1111111111111111111111111111
V     1111111111111111111111111111111111111111111
W     1111111111111111111111111111
X     1111111111111111111111111111111111111
Y     1111111111111111111111111111111111
Z     1111111111111111111111111111111111
```

A balanced frequency analysis is a clue that a Vigenère cipher might have been used.

Exercises

11. Here is a message encrypted with a Vigenère cipher with keyword *ultra*.

```
      NSXLS   HLOPT   OCGVD   NZWRY   NZGJN   WCMFC   LPTKE
      UYXCE   WEKFN   CNFRC   BTGVT   BLMTO   UWWHU   CNDCY
      LPTUE   HTZDA
```

11a. Do a frequency analysis of the ciphertext.
11b. Decrypt the message.

12. Encrypt the following message using a Vigenère cipher with keyword *packers*. Then do a frequency analysis of the ciphertext.

```
While the autokey was a brilliant idea, Cardano formulated
it defectively.
```

Cryptanalysis of the Vigenère Cipher

The keyword of a Vigenère cipher describes the rotation among the Caesar cipher alphabets that are used. That rotation leads to patterns that can be exploited by a cryptanalyst. If we know the length of the keyword, we can often determine the keyword and, hence, decrypt all messages encrypted with that keyword.

Here is a ciphertext message that has been encrypted with a Vigenère cipher.

```
nifon aicum niswt luvet vxshk nissx wsstb husle chsnv ytsro
cdsoy nisgx lnona chvch gnonw yndlh sfrnh npblr yowgf unoca
cossu ouoll iuvef issoe xgosa cpbew uormh lftaf cmwak bbbdv
cqvek muvil qbgnh ntiri ljgig atwnv yuvev iorim cpbsb hxviv
buvet vxshk uorim mjbdb pjrut fbueg ntgof yuwmx miodm ipdek
uuswx lfjek sewfy yssnm zscmm bpgeb huvez ysaag usaew mffvb
wfgim qpilw bbjeu yfbef vbfrt mtwnz uorig wpbvx hjsnm zpfag
uhsnm npglb jbqrh mttrh huwek mpfak ljjen hbbnh ooqew vzdak
udvum yucbx yoquf vffew vzonx hjumt lfgef vmwnz uxsiz bumag
xbbtb kvotx xumpx qswtx l
```

Assume that, somehow, we have discovered that the keyword has length five (which is conveniently the same as the size of the blocks). Then the first letter of each block is encrypted with the same row of the Vigenère square – they are encrypted with the same Caesar cipher. Similarly, the second letter of each block is encrypted with the same row – the same Caesar cipher. The third letters with the same Caesar cipher. The fourth letters with the same Caesar cipher. And, the fifth letters with the same Caesar cipher.

Because Caesar ciphers are easily broken by frequency analysis, we can discover the letters of the keyword. Here is how we can proceed.

Strip off the first letters of each block and do a frequency analysis on the result. They should have all been encrypted with the same Caesar cipher.

Alphabet number one – first letters of each block

A	11
B	11111
C	11111111
D	
E	
F	1
G	1
H	1111111
I	1111
J	1
K	1
L	11111111
M	1111111
N	11111111
O	11
P	1
Q	111
R	
S	11
T	
U	111111111
V	1111111
W	111
X	111
Y	1111111111
Z	11

It appears that ciphertext `Y` corresponds to plaintext `e`. (Not just because it is the most frequent letter but because all the high frequency letter patterns fit – `U` would correspond to `a`; `C` would correspond to `i`; `H` and `I` would correspond to `n` and `o`; and `L`, `M`, and `N` would correspond to `r`, `s`, and `t`.)

Now recall that when we are encrypting using a Vigenère square plaintext `a` corresponds to the first letter of the row being used – the letter of the keyword being used. So, it appears that (because `U` corresponds to `a`) the first letter of the keyword is `u`.

The keyword is `u` _ _ _ _.

Alphabet number two – second letters of each block

A	
B	11111111
C	
D	11
E	1
F	11111111
G	1
H	111
I	1111111
J	11111
K	
L	
M	11
N	1111
O	11111111
P	1111111111
Q	1
R	
S	1111111
T	111111
U	11111111111111
V	1
W	
X	1111
Y	
Z	11

It appears that ciphertext F corresponds to plaintext e.

So, it appears that (because B corresponds to a) the second letter of the keyword is b.

The keyword is u b _ _ _.

Alphabet number three – third letters of each block

A	11
B	111111111
C	111
D	111
E	
F	111111
G	1111111
H	
I	11
J	111
K	
L	
M	11
N	
O	11111111
P	
Q	111
R	111111
S	11111111111111111
T	11
U	11
V	1111111111
W	111111111
X	
Y	
Z	

It appears that ciphertext S corresponds to plaintext e.

So, it appears that the third letter of the keyword is o.

The keyword is u b o _ _. (Perhaps, you can already guess the keyword.)

Alphabet number four – fourth letters of each block

A	1111111
B	1
C	11
D	111
E	1111111111111111111
F	1
G	11
H	11
I	11111111
J	
K	
L	111111
M	1111
N	1111111111111
O	1111
P	1
Q	
R	11111
S	1111
T	1111
U	1111
V	11
W	11
X	
Y	
Z	

It appears that ciphertext `E` corresponds to plaintext `e`.

So, it appears that the third letter of the keyword is `a`.

The keyword is `u b o a _`.

Alphabet number five – fifth letters of each block

A	11
B	1111111
C	
D	
E	11
F	1111111
G	111111
H	11111111
I	1
J	
K	111111111
L	11
M	1111111111
N	11
O	1
P	
Q	
R	1
S	
T	111111
U	11
V	11111
W	111111
X	1111111111
Y	11
Z	1111

It appears that ciphertext X corresponds to plaintext e.

So, it appears that the last letter of the keyword is t.

The keyword is uboat.

So, knowing just the length of the keyword, we were able to determine the keyword.

Two methods give us information about the length of the keyword of a Vigenère cipher – the Kasiski test and the Friedman test. We will discuss the Kasiski test.

First, some history.

The Cryptanalysts

The Vigenère cipher might first have been broken by the English mathematician Charles Babbage (1792 – 1871); Kahn quotes Babbage as saying “an indistinct glimpse of defeating it presented itself vaguely to my imagination.” But, if Babbage had a solution, he never published it. Babbage apparently had the tendency to never be satisfied with a work and to continue to refine things; so, he might never have been satisfied enough with his solution to publish it.

Friedrich Kasiski (1805 – 1881) is credited with breaking the Vigenère cipher in 1863. From the Sixteenth Century until the Nineteenth Century the cipher was generally considered to be secure. We will use Kasiski’s technique to determine the length of the keyword.

In the Twentieth Century, William Frederick Friedman (1891 – 1969), the dean of American cryptologists, developed a statistical method to estimate the length of the keyword.

Friedrich Kasiski

“Friedrich Kasiski was born in November 1805 in a western Prussian town and enlisted in an East Prussian infantry regiment at the age of 17.

He moved up through the ranks to become a company commander and retired in 1852 as a major. Although he had become interested in cryptology during his military career, it was not until the 1860s that he put his ideas on paper. In 1863 his 95-page text *Die Geheimschriften und die Dechiffrirkunst* (*Secret Writing and the Art of Deciphering*) was published. A large part of its contents addressed the solution of polyalphabetic ciphers with repeating keywords, a problem that had tormented cryptanalysts for centuries.

Disappointed by the lack of interest in his findings, Kasiski turned his attention to other activities including anthropology. He took part in artifacts searches and excavations and wrote numerous archeological articles for scholarly journals. He died in May 1881 not realizing the significance of his cryptanalytic findings.” Wrixon, Fred B., *Codes, Ciphers & other Cryptic & Clandestine Communication: Making and breaking secret messages from hieroglyphs to the internet*, Black Dog & Leventhal Publishers.

The Kasiski Test (or the Kasiski Attack)

Here is a message enciphered with a Vigenère cipher. (It is taken from: Beutelspacher, Albrecht, *Cryptology: An introduction to the Art and Science of Enciphering, Encrypting, Concealing, Hiding and Safeguarding Described Without any Arcane Skullduggery but not Without Cunning Waggery for the Delectation and Instruction of the General Public,* Mathematical Association of America, 1996.)

```
DBZMG   AOIYS   OPVFH   OWKBW   XZPJL   VVRFG   NBKIX
DVUIM   OPFQL   VVPUD   KPRVW   OARLW   DVLMW   AWINZ
DAKBW   MMRLW   QIICG   PAKYU   CVZKM   ZARPS   DTRVD
ZWEYG   ABYYE   YMGYF   YAFHL   CMWLW   LCVHL   MMGYL
DBZIF   JNCYL   OMIAJ   JCGMA   IBVRL   OPVFW   OBVLK
OPVUJ   ZDVLQ   XWDGG   IQEYF   BTZMZ   DVRMM   ANZWA
ZVKFQ   GWEAL   ZFKNZ   ZZVCK   VDVLQ   BWFXU   CIEWW
OPRMU   JZIYK   KWEXA   IOIYH   ZIKYV   GMKNW   MOIIM
KADUQ   WMWIM   ILZHL   CMTCH   CMINW   SBRHV   OPVSO
DTCMG   HMKCE   ZASYD   JKRNW   YIKCF   OMIPS   GAFZK
JUVGM   GBZJD   ZWWNZ   ZVLGT   ZZFZS   GXYUT   ZBJCF
PAVNZ   ZAVWS   IJVZG   PVUVQ   NKRHF   DVXNZ   ZKZJZ
ZZKYP   OIEXX   MWDNZ   ZQIMH   VKZHY   DVKYD   GQXYF
OOLYK   NMJGS   YMRML   JBYYF   PUSYJ   JNRFH   CISYL
N
```

We begin the attack by frequency analysis.

A	1111111111111111111		19
B	11111111111111		14
C	1111111111111111		16
D	11111111111111111111	20	
E	11111111		8
F	1111111111111111111		19
G	11111111111111111111	20	
H	111111111111		12
I	11111111111111111111111111		26
J	1111111111111111		16
K	1111111111111111111111111	25	
L	1111111111111111111111		22
M	11111111111111111111111111111111		32
N	1111111111111111		16
O	111111111111111111		18
P	1111111111111111		16
Q	1111111111	10	
R	11111111111111		14
S	11111111111		11
T	111111		6
U	11111111111		11
V	1111111111111111111111111111111111		34
W	1111111111111111111111111111		28
X	1111111111	10	
Y	111111111111111111111111111		27
Z	11111111111111111111111111111111111111111		41
			491

The "relatively equal" frequencies suggest multiple alphabets – a polyalphabetic cipher, which, for us, would suggest a Vigenère cipher.

Here's the idea behind the Kasiski test. Consider a Vigenère cipher with keyword *Galois*. (My favorite mathematician.)

```
abcdefghijklmnopqrstuvwxyz
GHIJKLMNOPQRSTUVWXYZABCDEF
ABCDEFGHIJKLMNOPQRSTUVWXYZ
LMNOPQRSTUVWXYZABCDEFGHIJK
OPQRSTUVWXYZABCDEFGHIJKLMN
IJKLMNOPQRSTUVWXYZABCDEFGH
STUVWXYZABCDEFGHIJKLMNOPQR
```

Think of a common trigraph – say, `the` – and assume that it appears twice in the plaintext message.

If `the` is not encrypted by the same three alphabets at both locations, the two ciphertexts of `the` would be different.

```
GALOISGALOISGALOISGALOISGALOIS...GALOISGALOISGALOIS
the                                     the
ZHP                                     TSS
```

But, if we are lucky and `the` is encrypted by the same three alphabets, we would see a duplicate trigraph.

```
GALOISGALOISGALOISGALOISGALOIS...GALOISGALOISGALOIS
the                                    the
ZHP                                    ZHP
```

What is important to notice is that the distance between the beginnings of the `ZHP` trigraphs is a multiple of the length of the keyword. This provides information about the length of the keyword.

So, we search through the ciphertext for trigraphs (or strings of other lengths), and we look for repetitions. Sometimes, of course, the repetitions are just accidental – two different strings of three letters are encrypted into the same three-letter string by different alphabets, but sometimes the repetitions correspond to the same three-letter string being encrypted by the same three alphabets. These are the occurrences that we would like to discover.

Here are the trigraphs of the ciphertext with the number of repetitions shown in ( ):

```
AOI ARL AWI AKB AKY ARP ABY AFH AJJ AIB ANZ AZV ALZ AIO ADU
ASY AFZ AVN AVW

BZM BWX BKI BWM BYY(2) BZI BVR BVL BTZ BWF BRH BZJ BJC

CGP CVZ CMW CVH CYL CGM CKV CIE CMT CHC CMI CMG CEZ CFO CFP
CIS

DBZ(2) DVU DKP DVL(3) DAK DTR DZW DGG DVR DUG DTC DJK DZW
DVX DNZ DVK DGQ

EYG EYM EYF EAL EWW EXA EZA EXX

GAO GNB GPA GAB GYF GYL GMA GGI GIQ GWE GMK GHM GAF GMG GBZ
GTZ GXY GPV GQX GSY

HOW HLC(2) HLM HZI HCM HVO HMK HFD HVK HYD HCI

IYS IXD IMO INZ IIC ICG IFJ IAJ IBV IQE IEW IYK IOI IYH IKY
IIM IMK IMI ILZ INW IKC IPS IJV IEX IMH ISY

JLV JNC JJC JCG JZD JZI JKR JUV JDZ JCF JVZ JZZ JGS JBY JJN
JNR

KBW(2) KIX KPR KYU KMZ KOP KFQ KNZ KVD KKW KWE KYV KNW KAD
KCE KRN KCF KJU KRH KZJ KYP KZH KYD KNM

LVV(2) LWD LMW LWQ LCM(2) LWL LCV LMM LDB LOM LOP LKO LQX
LZF LQB LZH LGT LYK LJB

MGA MOP MWA MMR MRL MZA MGY(2) MWL MMG MIA MAI MZD MMA MAN
MUJ MKN MOI MKA MWI MIL MTC MIN MGH MKC MIF MGB MWD MHV MJG
MRM MLJ

NBK NZD NCY NZW NWM NWS NWY NZZ(5) NKR NMJ NRF

OIY OPV(4) OWK OPF OAR OMI(2) OBV OPR OIY OII ODT OIE OCL
OLY

PVF(2) PJL PFQ PUD PRV PAK PSD PVU(2) PRM PVS PSG PAV POI
PUS

QLV QII QXW QEY QGW QBW QWM QNK QIM QXY

RFG RVW RLW(2)RPS RVD RLO RMM RMU RHV RNW RHF RML RFH

SOP SDT SBR SOD SYD SGA SGX SIJ SYM SYJ SYL
```

TRV TZM TCH TCM TZZ TZB

UIM UDK UCV UJZ(2) UCI UQW UVG UTZ UVQ USY

VFH VVR VRF VUI VVP VPU VWO VLM VZK VDZ VHL VRL VFW VLK VUJ VLQ(2) VRM VKF VCK VDV VGM VOF VSO VGM VLG VNZ VWS VZG VUV VQN VXN VKZ VKY

WKB WXZ WOA WDV WAW WIN WMM WQI WEY WLW WLC WOB WDG WAZ WEA WFX WWO WOP WEX WMO WMW WIM WSB WYI WWN WNZ WSI WDN

XZP XDV XWD XUC XAI XYU XNZ XXM XMW XYF

YSO YUC YGA YYE YEY YMG YFY YAF YLD YLO YFB YKK YMZ YVG YDJ YIK YUT YPO YDV YDG YFO YKN YMR YYF YFP YJJ YLN

ZMG ZPJ ZDA ZKM ZAR ZWE ZIF ZDV(2) ZMZ ZWA ZVK ZFK ZZZ(2) ZZV(2) ZVC ZIY ZIK ZHL ZAS ZKJ ZJD ZWW ZVL ZZF ZFZ ZSG ZBJ ZZA ZAV ZGP ZZK(2) ZKZ ZJZ ZKY ZZQ ZQI ZHY

Whew!

After identifying the repeated trigraphs, we look at their spacing.

		BYY	360	BYY				
DBZ	140	DBZ						
DVL	121	**DVL**	**50**	**DVL**				
HLC	170	HLC						
KBW	55	KBW						
LVV	20	LVV						
LCM	170	LCM						
MGY	20	MGY						
NZZ	**140**	**NZZ**	**25**	**NZZ**	**25**	**NZZ**	**20**	**NZZ**
OPV	**155**	**OPV**	**10**	**OPV**	**135**	**OPV**		
OMI	190	OMI						
PVF	155	PVF						
PVU	224	PVU						
RLW	20	RLW						
UJZ	71	UJZ						
VLQ	50	VLQ						
ZZV	139	ZZV						
ZZK	6	ZZK						
ZDV	19	ZDV						
ZZZ	195	ZZZ						

Remember that we are looking for a length that is a common divisor of "all" of these lengths – well, not “all” because some repetitions are accidental – but most. The bolded portions of the table seem to indicate that the length of the keyword might be five.

Now we return to the ciphertext and separate it into its five alphabets. We begin with the first letter and take every fifth letter after it. Then take the second letter and every fifth letter after it. Then take the third letter and every fifth after it. Etc.

If we determined the length of the keyword correctly, we should have partitioned the ciphertext into five sets of ciphertext letters each of which was encrypted with a Caesar cipher. Then we proceed as we did for the first example of this section.

Let us look at each alphabet separately.

Alphabet number one

A	1111
B	11
C	111111
D	1111111111
E	
F	
G	111111
H	1
I	11111
J	1111111
K	111
L	1
M	1111
N	1111
O	1111111111111
P	1111
Q	1
R	
S	1
T	
U	
V	1111
W	1
X	11
Y	1111
Z	1111111111111111

It appears that Z might correspond to e; that would make V correspond to a. The first letter of the keyword would be v. v _ _ _ _

Alphabet number two

A	1111111111
B	1111111111
C	11
D	11
E	
F	1
G	
H	
I	111111
J	1
K	1111
L	1
M	1111111111111
N	111
O	1111
P	1111111
Q	111
R	
S	
T	111
U	11
V	11111111111
W	111111111
X	1
Y	
Z	11111

It appears that `M` might correspond to `e`; that would make `I` correspond to `a`. The second letter of the keyword would be `i`. `v i _ _ _` (Guess?)

Alphabet number three

A	
B	
C	11
D	111
E	111111
F	11111
G	111
H	
I	1111111111
J	11
K	111111111111
L	111
M	
N	
O	
P	11
Q	
R	1111111111111
S	111
T	1
U	11
V	11111111111111
W	111
X	11
Y	111
Z	111111111

It appears that V might correspond to e; that would make R correspond to a. The third letter of the keyword would be r. v i r _ _ (Guess?)

Alphabet number four

A	11
B	11
C	111111
D	
E	
F	11111
G	1111
H	111111
I	11111
J	111
K	1
L	111111
M	111111111
N	111111111
O	
P	11
Q	1
R	1
S	1
T	
U	1111
V	111
W	111
X	111
Y	1111111111111111111
Z	111

It appears that Y might correspond to e; that would make U correspond to a. The fourth letter of the keyword would be u. v i r u _ (Not many possibilities.)

Alphabet number five

A	111
B	
C	
D	11111
E	11
F	11111111
G	1111111
H	11111
I	
J	111
K	11111
L	11111111111
M	111111
N	
O	1
P	1
Q	11111
R	
S	111111
T	11
U	111
V	11
W	111111111111
X	11
Y	1
Z	11111111

It appears that W might correspond to e; that would make S correspond to a. The fifth letter of the keyword would be s. The keyword would be virus. Notice that we know the keyword, but we have not yet deciphered the message.

Exercises

13. Here is a message that was encrypted with a Vigenère cipher with a keyword of length 6.

```
wgixf irtnx amwpz gfcln bztef roozn maour tlrno dsxjw
xxdan zhdix nqtta hogcm rwrvj numyb gxavt mgzdt ewlqs
wvwtm lgblk nrins ozgif bgnlm fpsqn xhvja ufgmj xyxum
hqsxv vztea bzrpt lrijy ivnto fywew uyfse beiaw vbimm
igwhq ceytk ppien udmiq nkmtw bnidy tgitm lcfqy fhegp
ghewv viqbi pwsql itmtp avlzk mdmao gxmsf bgxls sokdm
eagyz azntg zdhvx rameg qifre snood yeqts tlreg dirpt
apirt bnqfe zwaez mzukd meavz qlitz gytln bxqvi ntkpg
ieugz ywczu bowrl gxlmn viqwm gpsqx mpwgs amaaz fhihv
ofehf bgfew nvjih sfmju sgywy grium rbehc zubep nukdi
gnqtf oxumc mr
```

13a. Find the first letter of the keyword.
13b. Find the second letter of the keyword.
13c. … the third letter … .
13d. …the fourth letter … .
13e. … the fifth letter … .
13f. … the sixth letter … .
13g. Find the keyword.
13h. Decrypt the message.

14a. Use a Kasiski test to determine the length of the keyword.
14b. Find the keyword.
14c. Decrypt the message.

Good luck!

```
tfrvg akceo ekiii brjgy obqgr nfbuk zimme tfyeb puwyr vqibj
ymeyv bfwyc actpu gwvvm akout cnzxx zvnaz ojbgu tpzkg ukcrv
punhk zlsth jqbtu fvbpn eypxn xdvcm giafv gvzly cnjgl ujunr
enfea uyiil vttlp lhreh vafgu lzkxh nccbh xuuel vymlz kgogn
ymoxb wbnvk exjws hnwbq amaud auoky scgom frgnz mbvor ufykz
vivye brxhz zvgke fvkga tfvvt rthue ukkyk prstc eznoe qrgsx
hgcwa jhhkw gnxhf zgnuc jmpzx xkprh kueku rbhvn eufio nbxwn
fknsu lzltk cojbg umbvv bxmbn mfzhz kprxt ccene coekg ohhfv
nyecx pgnbf coegv yujlg guekv kgntp hxvbr vquoy itbud ceogn
xwcil vbnjw szaym iyrxs jbbuw vcmgi afvgc gke
```

Here is a list of trigraphs that occur more than once:

Trigraph	**Frequency**
kpr	3
afv	2
bgu	2
ceo	2
cmg	2
coe	2
fvg	2
gia	2
gke	2
gnx	2
iaf	2
ilv	2
jbg	2
jws	2
kgo	2
lzk	2
mbv	2
mgi	2
oek	2
ogn	2
ojb	2
rvq	2
uek	2
ulz	2
vcm	2
vkg	2
yeb	2
zkg	2
zxx	2