Shift and Vigenère Ciphers with Maplets

Richard E. Klima
klimare@appstate.edu
Department of Mathematical Sciences
Appalachian State University
Boone, North Carolina, USA

Neil P. Sigmon
npsigmon@radford.edu
Department of Mathematics and Statistics
Radford University
Radford, Virginia, USA

Abstract: *Cryptology, the science and art of communicating in secret, provides an excellent tool for illustrating practical uses of mathematics. Topics from number theory, linear and abstract algebra, probability and statistics, and other areas all appear prominently throughout cryptographic methods and their cryptanalysis. Although historical, or "classical," ciphers are no longer widely used in a standalone way in our modern digital society, some do form parts of layered modern ciphers. For example, a variation of a shift cipher is included as a layer in the Advanced Encryption Standard, which serves as a current U.S. federal standard for private key encryption. Classical ciphers can also be used to directly connect mathematics and cryptology though, and advanced technology can significantly enhance how classical ciphers can be implemented and broken. As examples of this, in this paper we describe and illustrate the implementation and cryptanalysis of shift and Vigenère ciphers, using Maple, a software system freely available to faculty and students at many colleges and universities. We have found that similar examples using other cipher and/or software systems make valuable projects for students in both secondary and collegiate classes, who can use them to develop expertise in cipher and/or software systems.*

1 Introduction

Cryptology is a subject with much historical and modern significance which provides many interesting applications of mathematics. It can be used to demonstrate applications of topics from number theory, linear and abstract algebra, probability and statistics, and other areas. The use of technology to quickly show realistic examples can play an integral role in the teaching of cryptographic methods, and can also be invaluable in *cryptanalysis*, the process of eavesdropping on encrypted conversations, through accelerating processes that would otherwise be prohibitively time-consuming.

In this paper, we will demonstrate how technology can be used to enhance the teaching of elementary cryptology. In particular, we will illustrate how Maplets can be used for both the implementation and cryptanalysis of shift and Vigenère ciphers, two types of ciphers which are commonly presented as a means for introducing students to the science and art of cryptology.

2 Shift Ciphers

With shift ciphers, users begin with some agreed-upon order for the letters in their alphabet, such as the natural order A, B, … , Z of letters in our alphabet, and then transform a plaintext (i.e., an undisguised message) into a ciphertext (i.e., a disguised message) by replacing each letter with the letter some designated number of positions to the right in the alphabet, wrapping from the end of the alphabet to the start whenever necessary. For example, for a shift cipher with our alphabet letters in the natural order and in which each plaintext letter is encrypted by being replaced with the letter three positions to the right, any plaintext letter A would be replaced with D, B with E, … , W with Z, X with A, Y with B, and Z with C. Such a cipher is called a *shift* cipher because the correspondences between plaintext and ciphertext letters, or *cipher alphabet*, can be formed by first listing the alphabet letters in order representing plaintext letters, and then listing the corresponding letters representing ciphertext letters by shifting the plaintext list to the left the designated number of positions, wrapping from the start of the alphabet to the end whenever necessary.

For example, consider a shift cipher with a shift of three positions for encryption. This yields the following cipher alphabet.

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:	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

Using this cipher alphabet, the plaintext I CAME, I SAW, I CONQUERED encrypts to the ciphertext L FDPH, L VDZ, L FRQTXHUHG, or, equivalently, using a common historical convention of discarding punctuation and expressing ciphertexts in blocks of five letters each until the letters run out in the last block, LFDPH LVDZL FRQTX HUHG.

The cipher in the previous example was first described by the ancient Roman Emperor Julius Caesar in his writings on the Gallic Wars. Shift ciphers are not just something from the ancient past though. They were used by the Russian military as recently as the twentieth century, and the modern *ROT13* cipher, whose name is an abbreviation for "rotate 13 positions," is just a shift cipher with a shift of 13 positions for encryption. A variation of a shift cipher is also used as a layer in the modern Advanced Encryption Standard, which serves as a current U.S. federal standard for private key encryption.

Although not necessary, it is sometimes useful to represent shift ciphers mathematically using modular arithmetic, as a way, for example, to introduce students to a simple application of modular arithmetic. For a plaintext written using the letters in our alphabet A, B, … , Z, if we convert these letters into numbers using the correspondences A $= 0$, B $= 1$, … , Z $= 25$, we can then apply a shift cipher with a shift of b positions for encryption by adding b to the plaintext numbers with modulo 26 arithmetic. That is, for each plaintext number x in the set $\mathbb{Z}_{26} = \{0, 1, \ldots, 25\}$, we can find the corresponding ciphertext number y in $\mathbb{Z}_{26}$ using the formula

$$y = (x + b) \bmod 26.$$

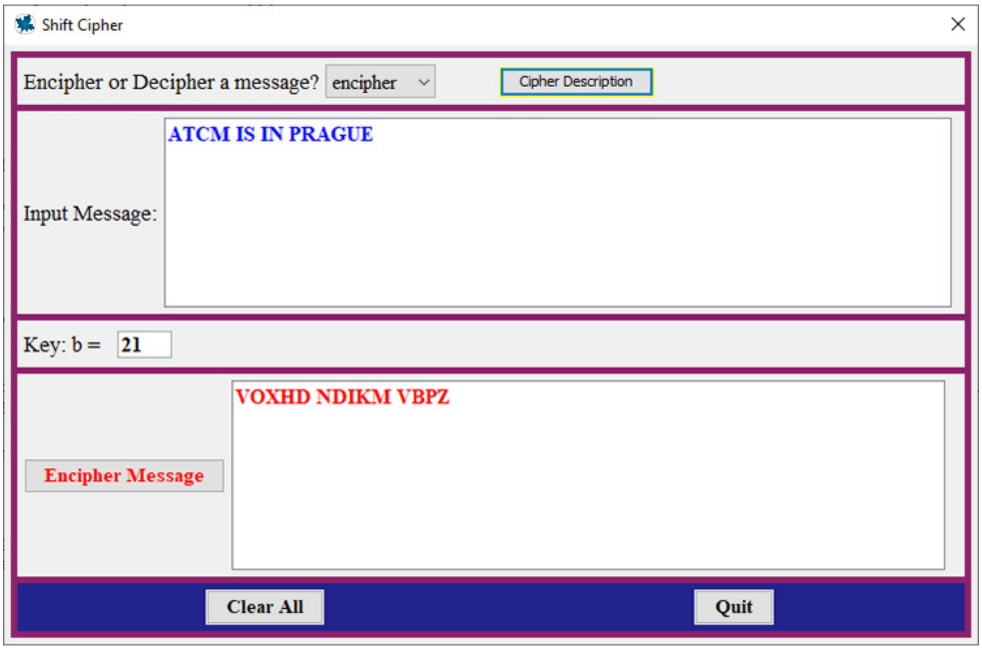
For example, for Caesar's cipher, encryption can be done using the formula $y = (x + 3) \bmod 26$. Resulting ciphertext numbers can then be converted into ciphertext letters using the same correspondences A $= 0$, B $= 1$, … , Z $= 25$.

To decrypt a ciphertext that was formed using a shift cipher, we must only undo what was done for encryption. That is, for a shift cipher with a shift of b positions for encryption, we would use a shift of b positions in the opposite direction for decryption. For a shift cipher represented using modular arithmetic, for each ciphertext number y in $\mathbb{Z}_{26}$, we can find the corresponding plaintext number x in $\mathbb{Z}_{26}$ using the formula

$$x = (y - b) \bmod 26.$$

For example, for ROT13 with the encryption formula $y = (x+13) \bmod 26$, the decryption formula is $x = (y-13) \bmod 26$.

We will now demonstrate a Maplet[1] written by the authors that can be used to encrypt or decrypt a message with a shift cipher. The source code for this Maplet and a directly usable version of it can be downloaded at [2]. Figure 1 shows how the Maplet can be used to encrypt the plaintext `ATCM IS IN PRAGUE` with the shift cipher $y = (x+21) \bmod 26$.



Encipher or Decipher a message? encipher

Input Message: ATCM IS IN PRAGUE

Key: b = 21

VOXHD NDIKM VBPZ

Figure 1: Shift cipher encryption example.

Note that for encryption, users must select the option to **encipher** from the drop-down menu at the top of the Maplet window. For decryption, users can select the option to **decipher** from this drop-down menu, with the ciphertext in the **Input Message** textbox.

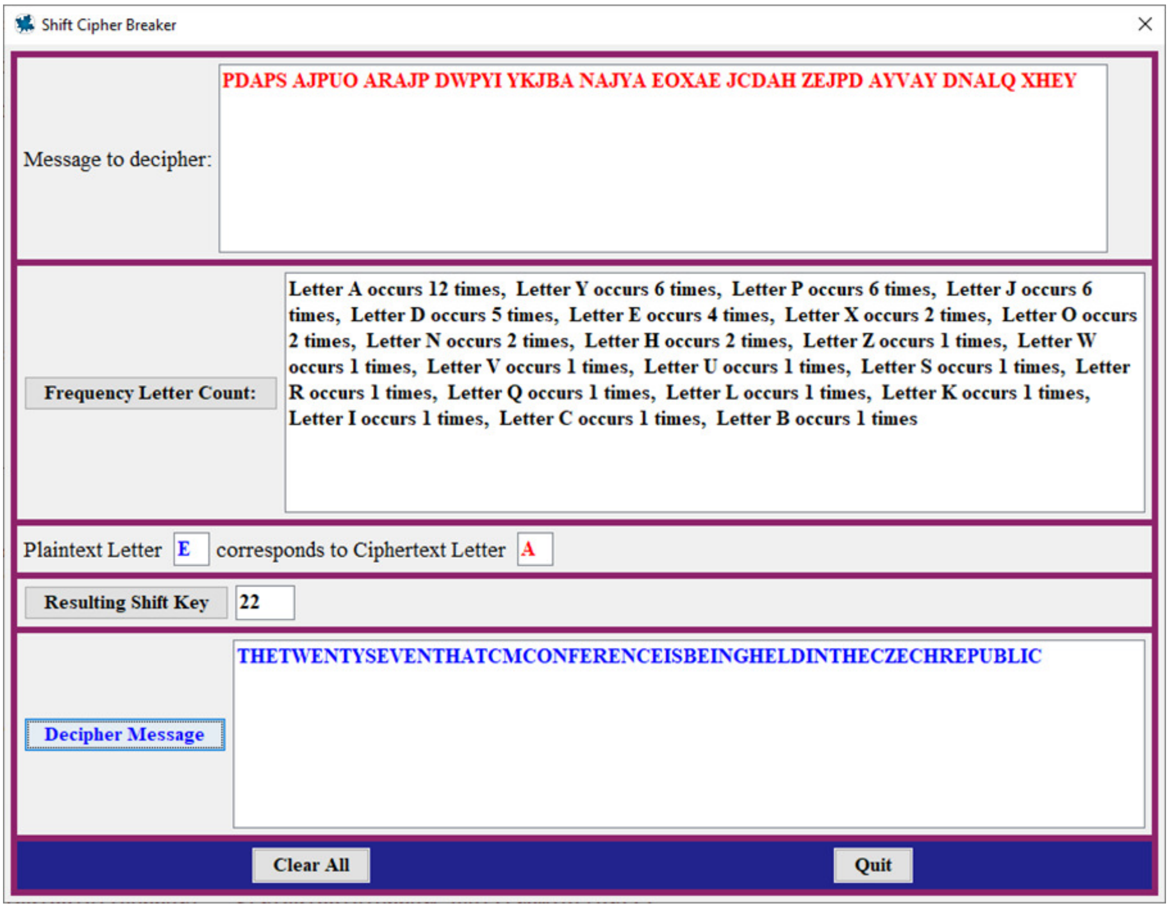
Ciphertexts formed using shift ciphers are not difficult to decrypt, even for those who do not know the specific shift, or *key*, for the cipher. The process of decrypting a ciphertext without knowledge of the key is called *cryptanalyzing*, or *breaking*, the cipher. For a plaintext written using our alphabet and encrypted with a shift cipher, the ciphertext could result from only 25 possible shifts (assuming a shift of 0 positions is not used). To break such a cipher, a *brute force* attack could be done by simply trying to decrypt the ciphertext assuming each of these 25 possible encryption shifts one at a time, stopping when the correct plaintext is revealed.

To reduce the time required to do this, *frequency analysis* could be used to identify some likely correspondences between plaintext and ciphertext letters. In particular, since the letters that naturally occur the most frequently in ordinary English are, in order, `E`, `T`, `A`, `O`, `I`, `N`, and `S`, for a plaintext written in ordinary English and encrypted using a shift cipher, it is reasonable to expect the letters

[1] A Maplet is like an applet, but uses (and requires) the engine of the computer algebra system Maple, and is written using Maple functions and syntax.

that occur in the ciphertext with the highest frequencies to correspond to letters such as these in the plaintext. Trying the decrypt the ciphertext assuming the encryption shifts that result from these correspondences first should limit the total number of shifts that must be checked.

We will now demonstrate a Maplet written by the authors that can be used to cryptanalyze a shift cipher. The source code for this Maplet and a directly usable version of it can be downloaded at [2]. Figure 2 shows how the Maplet can be used to cryptanalyze the ciphertext `PDAPS AJPUO ARAJP DWPYI YKJBA NAJYA EOXAE JCDAH ZEJPD AYVAY DNALQ XHEY`, which was formed using a shift cipher, using frequency analysis within the Maplet to recover the key. In particular, the Maplet performs a frequency count of the letters in the ciphertext, and shows that the most frequently occurring letter in the ciphertext is `A`. Assigning this ciphertext letter to the most frequently occurring letter in ordinary English, `E`, gives an encryption shift of 22, which in turn produces the plaintext.



Message to decipher: PDAPS AJPUO ARAJP DWPYI YKJBA NAJYA EOXAE JCDAH ZEJPD AYVAY DNALQ XHEY

Letter A occurs 12 times, Letter Y occurs 6 times, Letter P occurs 6 times, Letter J occurs 6 times, Letter D occurs 5 times, Letter E occurs 4 times, Letter X occurs 2 times, Letter O occurs 2 times, Letter N occurs 2 times, Letter H occurs 2 times, Letter Z occurs 1 times, Letter W occurs 1 times, Letter V occurs 1 times, Letter U occurs 1 times, Letter S occurs 1 times, Letter R occurs 1 times, Letter Q occurs 1 times, Letter L occurs 1 times, Letter K occurs 1 times, Letter I occurs 1 times, Letter C occurs 1 times, Letter B occurs 1 times

Plaintext Letter E corresponds to Ciphertext Letter A

Resulting Shift Key 22

THETWENTYSEVENTHATCMCONFERENCEISBEINGHELDINTHECZECHREPUBLIC

Figure 2: Shift cipher cryptanalysis example.

Additional information concerning the cryptanalysis of shift ciphers can be found in [1]. Understanding how shift ciphers work is essential in the cryptanalysis of Vigenère ciphers, which we will discuss next.

3 Vigenère Ciphers

Students are often introduced to the science and art of cryptology via *monoalphabetic* ciphers, which, like shift ciphers, use the same cipher alphabet throughout the entire encryption process. This makes monoalphabetic ciphers relatively easy to break though, since the distribution of letter frequencies in plaintexts is preserved into ciphertexts. One way to increase security is to change the cipher alphabet one or more times during encryption. Such ciphers are called *polyalphabetic*.

For a ciphertext formed using a polyalphabetic cipher, identical ciphertext letters will not necessarily correspond to identical plaintext letters. This makes polyalphabetic ciphers generally harder to break, possibly much harder, than monoalphabetic ciphers, since the changing cipher alphabets have the effect of evening out letter frequencies in ciphertexts. The World War II-era German Enigma machine was a polyalphabetic cipher device, as was a cipher wheel invented by Thomas Jefferson which was later produced and used as a U.S. Army field cipher during the twentieth century.

Vigenère ciphers[2] are often used to introduce students to polyalphabetic ciphers. This can be done without the need for modular arithmetic by implementing them as they were done so historically, using a rectangular array of letters called the *Vigenère square*, shown in Table 1.

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

Table 1: The Vigenère square.

The top row of the Vigenère square, which can be viewed as labeling the columns of the 26 × 26 inner part of the square, consists of the letters A through Z representing plaintext letters. The leftmost

[2] Vigenère ciphers are named for French diplomat Blaise de Vigenère (1523–1596), although they were first described by Italian cryptologist Giovan Battista Bellaso in 1553. Vigenère did describe an original variation in 1586 though.

column of the square, which can be viewed as labeling the rows of the inner part of the square, consists of the letters A through Z representing key letters. The letters in the 26×26 inner part of the square represent ciphertext letters that correspond to pairs of plaintext and key letters, with the ciphertext letter corresponding to a particular pair of plaintext and key letters being the letter where the column labeled with the plaintext letter intersects the row labeled with the key letter. The connection to shift ciphers is that the 26 rows in the inner part of the Vigenère square are the 26 possible shift cipher alphabets.

Vigenère ciphers require the originator and intended recipient of a message to agree upon one or more words to form a *keyword*. Encryption was done historically using the Vigenère square, with the key letters determined by repeating the letters in the keyword as many times as necessary until the total number of key letters matched the total number of plaintext letters. For example, for a Vigenère cipher with the keyword TIME used to encrypt the plaintext MEET AT, the first ciphertext letter is the letter in the inner part of the square where the column labeled with M intersects the row labeled with T. The entire encryption is as follows.

Plain:	M	E	E	T	A	T
Key:	T	I	M	E	T	I
Cipher:	F	M	Q	X	T	B

Note that this cipher is polyalphabetic, since it uses more than one cipher alphabet. In particular, note that the two plaintext letters E encrypt to different ciphertext letters, since they were formed using different rows of the Vigenère square, or, equivalently, different cipher alphabets.

For a Vigenère cipher with the keyword TIME used to decrypt the ciphertext YWGVI U, to find the first plaintext letter, we can go to the row of the square labeled with the first key letter T, and find the first ciphertext letter Y in this row. The label of the column in which this ciphertext letter appears is the first plaintext letter. The entire decryption is as follows.

Cipher:	Y	W	G	V	I	U
Key:	T	I	M	E	T	I
Plain:	F	O	U	R	P	M

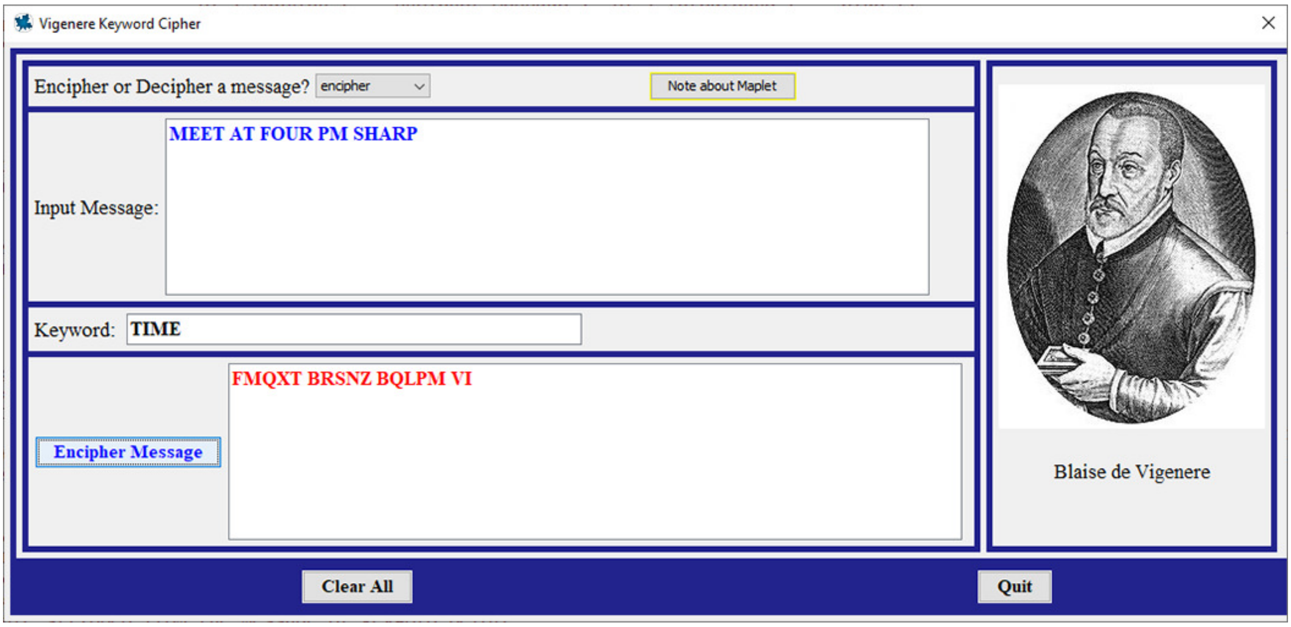
Since Vigenère ciphers are a combination of different shift ciphers, they can also be represented mathematically using modular arithmetic. If we convert letters into numbers using the correspondences $\mathtt{A} = 0$, $\mathtt{B} = 1, \ldots, \mathtt{Z} = 25$, then the Vigenère square can just be viewed as a modulo 26 addition table for $\mathbb{Z}_{26}$. For example, to encrypt the plaintext letter M using a Vigenère cipher with the key letter T, we can take the numerical representations 12 of M and 19 of T, and compute $(12+19) \bmod 26 = 5$, which is the numerical representation of the ciphertext letter F that results from using the Vigenère square to do the encryption directly. Similarly, to decrypt the ciphertext letter Y using a Vigenère cipher with the key letter T, we can take the numerical representations 24 of Y and 19 of T, and compute $(24-19) \bmod 26 = 5$, which is the numerical representation of the plaintext letter F that results from using the Vigenère square to do the decryption directly.

For student projects involving software coding, it is useful to consider a refined version of this modular arithmetic representation of Vigenère ciphers. In particular, for a sequence of plaintext numbers $(p_0, p_1, \ldots, p_{m-1})$ of length m, and a sequence of keyword numbers $(k_0, k_1, \ldots, k_{n-1})$ of length n with $n \leq m$, the entries in the resulting sequence of ciphertext numbers $(c_0, c_1, \ldots, c_{m-1})$ can be found for $i = 0, 1, \ldots, m-1$ using the formula

$$c_i = (p_i + k_{i \bmod n}) \bmod 26.$$

In the case when $n = m$, and with a keyword consisting of truly random letters, a Vigenère cipher becomes a *one-time pad*, which is provably the only unbreakable type of cipher system. So Vigenère ciphers, which derive from an easy process to understand and implement, leads very quickly to modern cryptographic ideas.

We will now demonstrate a Maplet written by the authors that can be used to encrypt or decrypt a message with a Vigenère cipher. The source code for this Maplet and a directly usable version of it can be downloaded at [2]. Figure 3 shows how the Maplet can be used to encrypt the plaintext `MEET AT FOUR PM SHARP` with a Vigenère cipher with the keyword `TIME`.



Vigenere Keyword Cipher

Encipher or Decipher a message? encipher

Input Message: MEET AT FOUR PM SHARP

Keyword: TIME

FMQXT BRSNZ BQLPM VI

Figure 3: Vigenère cipher encryption example.

Note that for encryption, users must again select the option to **encipher** from the drop-down menu at the top of the Maplet window. For decryption, users can again select the option to **decipher** from this drop-down menu, with the ciphertext in the **Input Message** textbox.

4 Cryptanalysis of Vigenère Ciphers

To break a Vigenère cipher, the keyword for the cipher must be determined. A first step in doing this is to find an estimate for the length of the keyword, for which the *index of coincidence* is a tool that can be used. The index of coincidence is a concept developed in the 1920s by William Friedman, one of history's greatest cryptologists, who is referred to as the "Dean of American Cryptology" on a bust at the U.S. National Cryptologic Museum. Although the index of coincidence is not Friedman's most technically advanced original idea, Friedman called it his greatest creation in his own writings on cryptology.

4.1 The Index of Coincidence

Consider the frequency percentages with which the 26 letters in our alphabet occur in ordinary English, which are shown in Table 2.

Letter	Frequency (%)	Letter	Frequency (%)
A	8.17	N	6.75
B	1.49	O	7.51
C	2.78	P	1.93
D	4.25	Q	0.10
E	12.70	R	5.99
F	2.23	S	6.33
G	2.02	T	9.06
H	6.09	U	2.76
I	6.97	V	0.98
J	0.15	W	2.36
K	0.77	X	0.15
L	4.03	Y	1.97
M	2.41	Z	0.07

Table 2: Letter frequency percentages in ordinary English.

These frequencies can also be viewed as probabilities. For example, the probability that a single letter chosen at random from ordinary English (equivalently, from a very large text written in ordinary English) will be an A is 0.0817.

The index of coincidence is a number that measures variation in character frequencies. More specifically, the index of coincidence for a language is the probability that two characters chosen at random from the language will be identical. Using the letter frequency percentages in Table 2, we can see that the probability of choosing the letter A at random twice from ordinary English is $(0.0817)^2 = 0.0067$. Similarly, the probability of choosing the letter B at random twice from ordinary English is $(0.0149)^2 = 0.0002$. Continuing in this manner, we find that the index of coincidence for ordinary English is

$$(0.0817)^2 + (0.0149)^2 + \cdots + (0.0007)^2 = 0.0655.$$

Consider now a mythical language that uses the same alphabet, but for which the frequencies with which the letters occur are distributed exactly evenly. In this language, for any first letter chosen at random, the probability that a second letter chosen at random would match the first would be $\frac{1}{26} = 0.0385$. Thus, the index of coincidence for this language would be 0.0385.

The index of coincidence is a concept that can also be applied to samples of text. Specifically, for a sample of text, the index of coincidence is the probability that two characters chosen at random from the text will be identical. Since monoalphabetic ciphers preserve letter frequencies, we would expect a ciphertext produced by a monoalphabetic cipher to have an index of coincidence closer to 0.0655 than 0.0385. Polyalphabetic ciphers, on the other hand, have letter frequencies that are distributed more evenly. Thus, we would expect a ciphertext produced by a polyalphabetic cipher to have an index of coincidence closer to 0.0385.

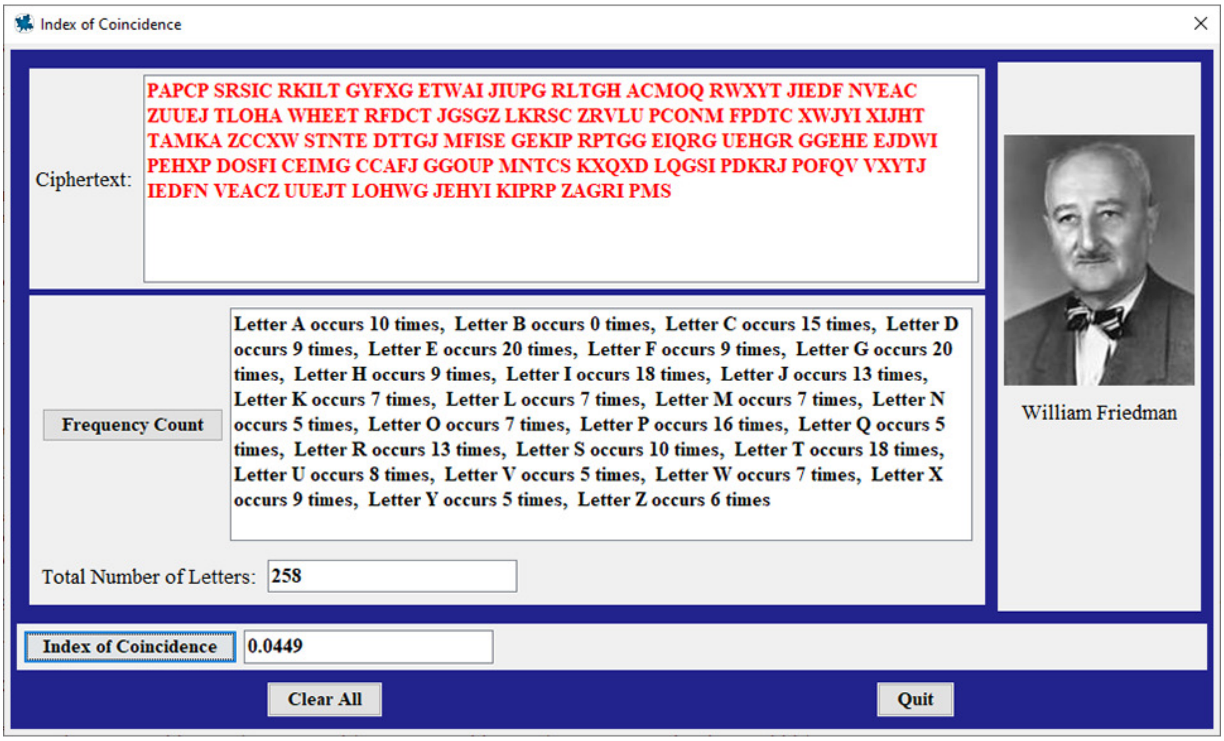
For a sample of text of length m, if the letter A appears m_0 times, then the probability of choosing A at random twice from the text (without replacement) is

$$\frac{1}{m(m-1)} m_0(m_0 - 1).$$

Similarly, if the letters A, B, ... , Z appeared in the text m_0, m_1, ... , m_{25} times, respectively, then the index of coincidence I for the text is given by the formula

$$I = \frac{1}{m(m-1)} \sum_{i=0}^{25} m_i(m_i - 1). \tag{1}$$

We will now demonstrate a Maplet written by the authors that uses (1) to find the index of coincidence for a provided sample of text. The source code for this Maplet and a directly usable version of it can be downloaded at [2]. Figure 4 shows how the Maplet can be used to find the index of coincidence for the ciphertext `PAPCP SRSIC RKILT GYFXG ETWAI JIUPG RLTGH ACMOQ RWXYT JIEDF NVEAC ZUUEJ TLOHA WHEET RFDCT JGSGZ LKRSC ZRVLU PCONM FPDTC XWJYI XIJHT TAMKA ZCCXW STNTE DTTGJ MFISE GEKIP RPTGG EIQRG UEHGR GGEHE EJDWI PEHXP DOSFI CEIMG CCAFJ GGOUP MNTCS KXQXD LQGSI PDKRJ POFQV VXYTJ IEDFN VEACZ UUEJT LOHWG JEHYI KIPRP ZAGRI PMS`, which was formed using a Vigenère cipher. The resulting index of coincidence of 0.0449 confirms that the cipher that produced the ciphertext is more likely to be polyalphabetic than monoalphabetic.



Ciphertext: PAPCP SRSIC RKILT GYFXG ETWAI JIUPG RLTGH ACMOQ RWXYT JIEDF NVEAC ZUUEJ TLOHA WHEET RFDCT JGSGZ LKRSC ZRVLU PCONM FPDTC XWJYI XIJHT TAMKA ZCCXW STNTE DTTGJ MFISE GEKIP RPTGG EIQRG UEHGR GGEHE EJDWI PEHXP DOSFI CEIMG CCAFJ GGOUP MNTCS KXQXD LQGSI PDKRJ POFQV VXYTJ IEDFN VEACZ UUEJT LOHWG JEHYI KIPRP ZAGRI PMS

Letter A occurs 10 times, Letter B occurs 0 times, Letter C occurs 15 times, Letter D occurs 9 times, Letter E occurs 20 times, Letter F occurs 9 times, Letter G occurs 20 times, Letter H occurs 9 times, Letter I occurs 18 times, Letter J occurs 13 times, Letter K occurs 7 times, Letter L occurs 7 times, Letter M occurs 7 times, Letter N occurs 5 times, Letter O occurs 7 times, Letter P occurs 16 times, Letter Q occurs 5 times, Letter R occurs 13 times, Letter S occurs 10 times, Letter T occurs 18 times, Letter U occurs 8 times, Letter V occurs 5 times, Letter W occurs 7 times, Letter X occurs 9 times, Letter Y occurs 5 times, Letter Z occurs 6 times

Total Number of Letters: 258

Index of Coincidence: 0.0449

Figure 4: Index of coincidence example.

4.2 Finding the Length of the Keyword

The index of coincidence is a tool that can also be used to find an estimate for the length of the keyword for a Vigenère cipher. To see how, consider the following example of encryption with a Vigenère cipher.

Plain:	H	A	V	I	N	G	A	P	E	T	C	A	N	M	A	K	E	Y	O	U	H	A	P	P	Y
Key:	T	R	I	X	I	E	T	R	I	X	I	E	T	R	I	X	I	E	T	R	I	X	I	E	T
Cipher:	A	R	D	F	V	K	T	G	M	Q	K	E	G	D	I	H	M	C	H	L	P	X	X	T	R

Note that in this encryption, every sixth plaintext letter starting with the first is encrypted using the same key letter T. As such, every sixth plaintext letter starting with the first is encrypted using a shift cipher with a key of 19. Similarly, every sixth plaintext letter starting with the second is encrypted using the same key letter R, or, equivalently, a shift cipher with a key of 17.

This reveals an important fact about Vigenère ciphers—each keyword letter yields its own shift cipher. In a ciphertext produced by a Vigenère cipher, all of the letters encrypted using the letter in the same keyword position form a *coset*. For example, in the example in the previous paragraph, the coset resulting from the keyword letter T consists of the ciphertext letters A, T, G, H, and R. Similarly, the coset resulting from the keyword letter I the first time it appears in the keyword consists of the ciphertext letters D, M, I, and P.

For a ciphertext produced by a Vigenère cipher, the number of cosets is the same as the length of the keyword. Also, and crucially, since all of the letters in a coset are formed using the same shift cipher, the index of coincidence for the coset should indicate that the cipher used to form the letters in the coset is more likely to be monoalphabetic than polyalphabetic.

To see how this idea can be used to find an estimate for the length of the keyword, suppose the number of cosets is n, and let I_j be the index of coincidence for the jth coset. As a measure of how likely a collection of cosets is to have been produced by monoalphabetic rather than polyalphabetic ciphers, we use the average of the indices I_j for the cosets, a number we will denote for a particular value of n by $\overline{I_{1:n}}$:

$$\overline{I_{1:n}} = \frac{1}{n} \sum_{j=1}^{n} I_j.$$

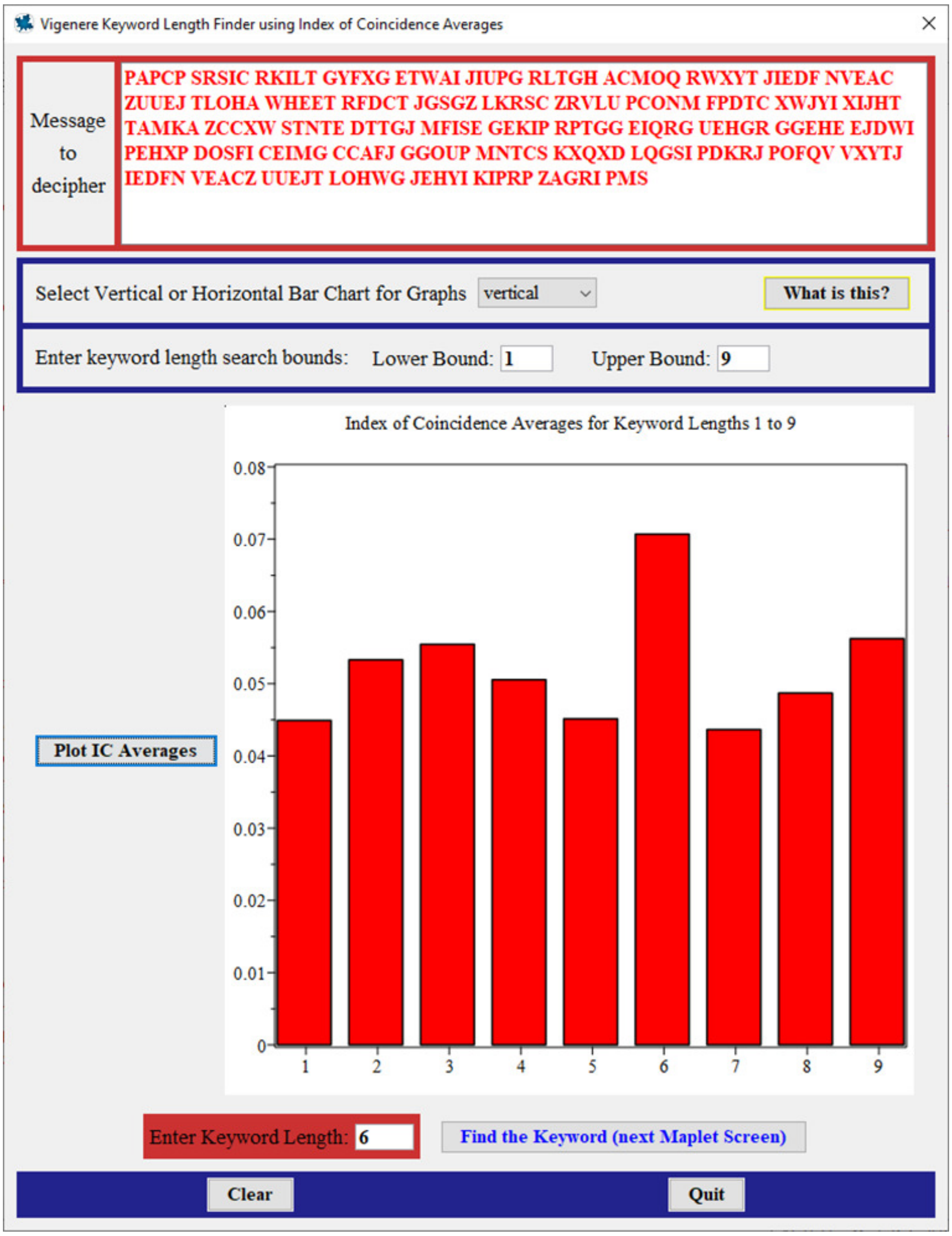
Of course, when starting the process of trying to break a Vigenère ciphertext, we would not know the value of n. The idea is to find $\overline{I_{1:n}}$ for several values of n, looking for the smallest n for which $\overline{I_{1:n}}$ is closer to 0.0655 than 0.0385, and, usually, noticeably larger than for the surrounding values of n. Such a value of n would be the likely length of the keyword for the cipher. For example, for the ciphertext shown in Figure 4, the (rounded) values of I_j for $j = 1, 2, \ldots, n$ and $\overline{I_{1:n}}$ for $n = 1, 2, \ldots, 9$ are shown in Table 3.

n	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	$\overline{I_{1:n}}$
1	0.045									0.045
2	0.060	0.046								0.053
3	0.060	0.058	0.048							0.055
4	0.059	0.042	0.059	0.043						0.051
5	0.044	0.043	0.037	0.038	0.064					0.045
6	0.080	0.058	0.069	0.052	0.084	0.082				**0.071**
7	0.050	0.039	0.045	0.032	0.051	0.048	0.041			0.044
8	0.042	0.045	0.054	0.042	0.079	0.036	0.050	0.040		0.049
9	0.076	0.052	0.049	0.057	0.052	0.062	0.061	0.058	0.040	0.056

Table 3: Average indices of coincidence.

In Table 3, since $\overline{I_{1:n}}$ is noticeably larger for $n = 6$, we would know, assuming the ciphertext was formed using a Vigenère cipher, that the length of the keyword for the cipher was most likely 6.

We will now demonstrate a Maplet written by the authors that finds the average of the indices of coincidence for a specified range of values of n, displaying each in a bar graph to make comparisons most easily evident. The source code for this Maplet and a directly usable version of it can be downloaded at [2]. Figure 5 shows how the Maplet can be used to find, assuming the ciphertext in Figure 4 was formed using a Vigenère cipher, that the length of the keyword for the cipher was most likely 6. After entering this number in the textbox at the bottom of the Maplet window, users can progress to a subsequent window in the Maplet to actually find the keyword, as we describe next.



Vigenere Keyword Length Finder using Index of Coincidence Averages

Message to decipher

PAPCP SRSIC RKILT GYFXG ETWAI JIUPG RLTGH ACMOQ RWXYT JIEDF NVEAC ZUUEJ TLOHA WHEET RFDCT JGSGZ LKRSC ZRVLU PCONM FPDTC XWJYI XIJHT TAMKA ZCCXW STNTE DTTGJ MFISE GEKIP RPTGG EIQRG UEHGR GGEHE EJDWI PEHXP DOSFI CEIMG CCAFJ GGOUP MNTCS KXQXD LQGSI PDKRJ POFQV VXYTJ IEDFN VEACZ UUEJT LOHWG JEHYI KIPRP ZAGRI PMS

Select Vertical or Horizontal Bar Chart for Graphs: vertical

Enter keyword length search bounds: Lower Bound: 1 Upper Bound: 9

Enter Keyword Length: 6

What is this?
Index of Coincidence Averages for Keyword Lengths 1 to 9
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
1
2
3
4
5
6
7
8
9
Plot IC Averages
Find the Keyword (next Maplet Screen)
Clear
Quit

Figure 5: Finding the likely length of the keyword for a Vigenère cipher.

4.3 Finding the Letters in the Keyword

After the length of the keyword for a Vigenère cipher has been found, the keyword itself can be determined one letter at a time. Recall that in a Vigenère cipher, each particular keyword letter produces ciphertext letters which are all part of the same coset, and which result from a corresponding monoalphabetic shift cipher. Let $\mathbf{x}_0$ be the following vector, containing the frequencies with which the 26 letters in our alphabet occur in ordinary English, which were shown previously as percentages in Table 2.

$$\begin{aligned}\mathbf{x}_0 \;=\; &(0.0817, 0.0149, 0.0278, 0.0425, 0.1270, 0.0223, 0.0202, 0.0609, 0.0697, 0.0015,\\ &0.0077, 0.0403, 0.0241, 0.0675, 0.0751, 0.0193, 0.0010, 0.0599, 0.0633, 0.0906,\\ &0.0276, 0.0098, 0.0236, 0.0015, 0.0197, 0.0007)\end{aligned}$$

Also, for each $i = 1, 2, \ldots, 25$, let $\mathbf{x}_i$ be the vector that results from shifting the entries in $\mathbf{x}_0$ to the right by i positions, with the entries from the end of the vector wrapping around to the start whenever necessary. For example, $\mathbf{x}_2$ would be the following vector.

$$\begin{aligned}\mathbf{x}_2 \;=\; &(0.0197, 0.0007, 0.0817, 0.0149, 0.0278, 0.0425, 0.1270, 0.0223, 0.0202, 0.0609,\\ &0.0697, 0.0015, 0.0077, 0.0403, 0.0241, 0.0675, 0.0751, 0.0193, 0.0010, 0.0599,\\ &0.0633, 0.0906, 0.0276, 0.0098, 0.0236, 0.0015)\end{aligned}$$

Finally, for a particular Vigenère ciphertext coset, let $\mathbf{y} = (y_0, y_1, \ldots, y_{25})$ be a vector containing the relative frequencies with which letters occur in the coset. Since all of the letters in the coset would have been formed using the same monoalphabetic shift cipher, we would expect the values in $\mathbf{y}$ to be distributed similarly to how the values in $\mathbf{x}_0$ are distributed, in the same order, except shifted to the right some number of positions, with the entries from the end of the vector wrapping around to the start whenever necessary.

To determine this number of positions, we can use the fact that the dot product $\mathbf{x}_i \cdot \mathbf{y}$ will be as large as possible whenever the vectors $\mathbf{x}_i$ and $\mathbf{y}$ are as close to parallel as possible. More specifically, given the vectors $\mathbf{x}_i$ for $i = 0, 1, \ldots, 25$ containing the frequencies with which the 26 letters in our alphabet occur in ordinary English shifted to the right by i positions, and a vector $\mathbf{y}$ containing the relative frequencies with which letters occur in a particular coset, the value of i for which $\mathbf{x}_i \cdot \mathbf{y}$ is the largest is likely to be the number of positions for which the vector that results from shifting the entries in $\mathbf{x}_0$ to the right this number of positions most closely resembles the relative frequencies of the letters in the coset. This will reveal the most likely shift cipher that produced the coset, and, consequently, the most likely letter in the keyword. Completing this procedure for each coset should reveal the entire keyword.

To demonstrate this, consider again the ciphertext in Figure 4, which was formed using a Vigenère cipher, for which we previously found the length of the keyword to likely be 6. We will now show how the first letter in the keyword can be determined. We start with the first coset in the ciphertext (i.e., every sixth ciphertext letter starting with the first), which is `PRIXI RCXDC THDGC PPJHA STIIG UGDXI CGTXI PXDCT JIG`. The number of times that each letter occurs in this coset is shown in the following list.

Letter:	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
Frequency:	1	0	5	4	0	0	5	2	7	2	0	0	0	0	0	4	0	2	1	4

Letter:	U	V	W	X	Y	Z
Frequency:	1	0	0	5	0	0

Dividing each of these frequencies by 43, which is the total number of letters in the coset, gives the values in the following vector $\mathbf{y}$, which contains the relative frequencies with which letters occur in the coset.

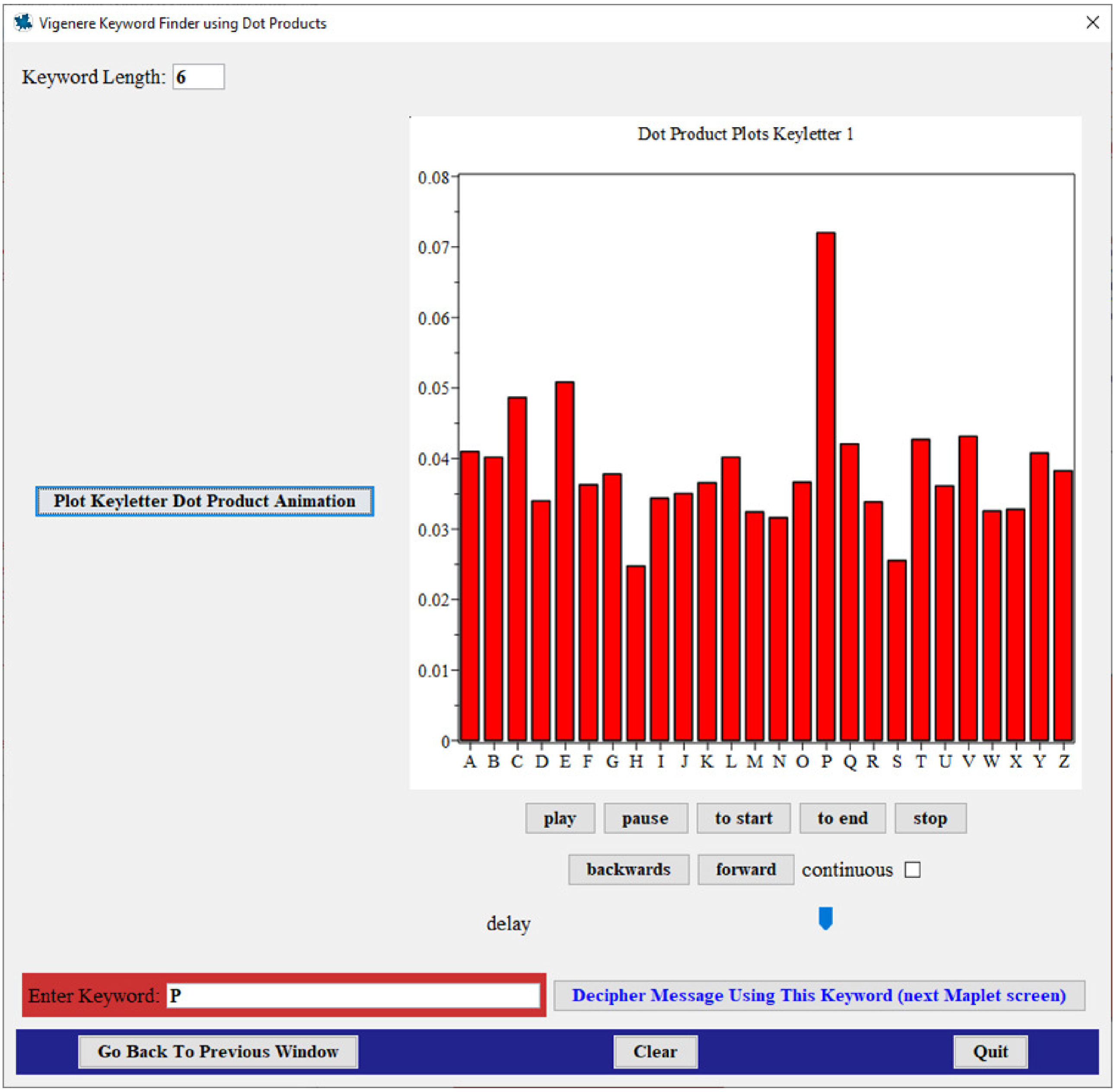
$$\begin{aligned}\mathbf{y} \;=\; &(0.0233, 0.0000, 0.1163, 0.0930, 0.0000, 0.0000, 0.1163, 0.0465, 0.1628, 0.0465,\\ &0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0930, 0.0000, 0.0465, 0.0233, 0.0930,\\ &0.0233, 0.0000, 0.0000, 0.1163, 0.0000, 0.0000)\end{aligned}$$

Then for each of the vectors $\mathbf{x}_i$ for $i = 0, 1, \ldots, 25$, we form the dot product $\mathbf{x}_i \cdot \mathbf{y}$. These calculations result in the following.

$$\begin{aligned}
\mathbf{x}_0 \cdot \mathbf{y} &= 0.0409\\
\mathbf{x}_1 \cdot \mathbf{y} &= 0.0402\\
\mathbf{x}_2 \cdot \mathbf{y} &= 0.0486\\
\mathbf{x}_3 \cdot \mathbf{y} &= 0.0340\\
\mathbf{x}_4 \cdot \mathbf{y} &= 0.0508\\
\mathbf{x}_5 \cdot \mathbf{y} &= 0.0363\\
\mathbf{x}_6 \cdot \mathbf{y} &= 0.0378\\
\mathbf{x}_7 \cdot \mathbf{y} &= 0.0247\\
\mathbf{x}_8 \cdot \mathbf{y} &= 0.0344\\
\mathbf{x}_9 \cdot \mathbf{y} &= 0.0350\\
\mathbf{x}_{10} \cdot \mathbf{y} &= 0.0365\\
\mathbf{x}_{11} \cdot \mathbf{y} &= 0.0402\\
\mathbf{x}_{12} \cdot \mathbf{y} &= 0.0324\\
\mathbf{x}_{13} \cdot \mathbf{y} &= 0.0316\\
\mathbf{x}_{14} \cdot \mathbf{y} &= 0.0366\\
\mathbf{x}_{15} \cdot \mathbf{y} &= \mathbf{0.0720}\\
\mathbf{x}_{16} \cdot \mathbf{y} &= 0.0421\\
\mathbf{x}_{17} \cdot \mathbf{y} &= 0.0338\\
\mathbf{x}_{18} \cdot \mathbf{y} &= 0.0255\\
\mathbf{x}_{19} \cdot \mathbf{y} &= 0.0427\\
\mathbf{x}_{20} \cdot \mathbf{y} &= 0.0361\\
\mathbf{x}_{21} \cdot \mathbf{y} &= 0.0431\\
\mathbf{x}_{22} \cdot \mathbf{y} &= 0.0326\\
\mathbf{x}_{23} \cdot \mathbf{y} &= 0.0328\\
\mathbf{x}_{24} \cdot \mathbf{y} &= 0.0408\\
\mathbf{x}_{25} \cdot \mathbf{y} &= 0.0383
\end{aligned}$$

The value of i for which $\mathbf{x}_i \cdot \mathbf{y}$ is the largest is $i = 15$, and so the most likely shift cipher that produced the first coset has key 15. Since the letter that corresponds to this shift in our list of correspondences is $\mathtt{P} = 15$, the most likely first keyword letter is `P`.

From the Maplet window in Figure 5, users can progress to a subsequent window in the Maplet to perform these calculations and comparisons. Figure 6 shows how the Maplet can be used to find the 26 dot products for the first coset, displaying each in a bar graph to make comparisons most easily evident, and thus revealing that the most likely first keyword letter is P.



Vigenere Keyword Finder using Dot Products

Keyword Length: 6

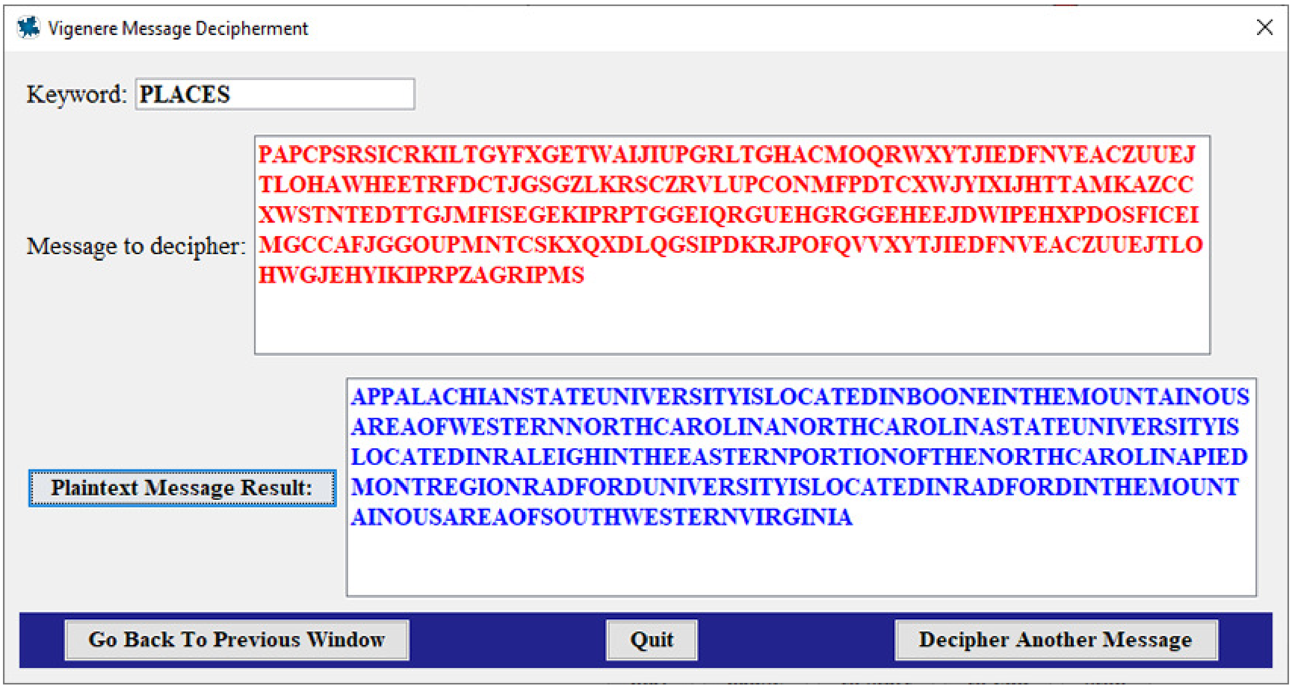
Enter Keyword: P

Plot Keyletter Dot Product Animation
Dot Product Plots Keyletter 1
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
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
play
pause
to start
to end
stop
backwards
forward
continuous □
delay
Decipher Message Using This Keyword (next Maplet screen)
Go Back To Previous Window
Clear
Quit

Figure 6: Determining the likely first letter in the keyword for a Vigenère cipher.

Using the animation tools within the Maplet, we would similarly be able to determine that the most likely second through sixth letters in the keyword are L, A, C, E, and S, respectively, thus revealing that the most likely entire keyword is PLACES.

After determining the keyword, users can progress to a final window in the Maplet to use the keyword to decrypt the entire ciphertext. Figure 7 shows how the Maplet can be used to recover the plaintext.



```
Keyword: PLACES

Message to decipher:
PAPCPSRSICRKILTGYFXGETWAIJIUPGRLTGHACMOQRWXYTJIEDFNVEACZUUEJ
TLOHAWHEETRFDCTJGSGZLKRSCZRVLUPCONMFPDTCXWJYIXIJHTTAMKAZCC
XWSTNTEDTTGJMFISEGEKIPRPTGGEIQRGUEHGRGGEHEEJDWIPEHXPDOSFICEI
MGCCAFJGGOUPMNTCSKXQXDLQGSIPDKRJPOFQVVXYTJIEDFNVEACZUUEJTLO
HWGJEHYIKIPRPZAGRIPMS

Plaintext Message Result:
APPALACHIANSTATEUNIVERSITYISLOCATEDINBOONEINTHEMOUNTAINOUS
AREAOFWESTERNNORTHCAROLINANORTHCAROLINASTATEUNIVERSITYIS
LOCATEDINRALEIGHINTHEEASTERNPORTIONOFTHENORTHCAROLINAPIED
MONTREGIONRADFORDUNIVERSITYISLOCATEDINRADFORDINTHEMOUNT
AINOUSAREAOFSOUTHWESTERNVIRGINIA
```

Figure 7: Vigenère cipher cryptanalysis example.

5 Conclusion

In this paper, we described and illustrated the implementation and cryptanalysis of shift and Vigenère ciphers, using Maplets which were written by the authors to help in demonstrating this. These Maplets are all available for download at [2]. Although these Maplets are only directly usable by readers who have access to Maple, which we recognize is notably not a free software system, many academic professionals like ourselves, and their students, have free access to Maple through site licenses purchased by their home institutions. Further, we have found that similar examples using other cipher systems make valuable projects for students in both secondary and collegiate classes, who can use them to develop expertise in other cipher systems. A myriad of additional examples using Maplets with both classical and modern ciphers can be found in [1].

Finally, even for professionals (and amateurs) without access to Maple, we still believe this paper has value, as we have also found that similar examples using other software systems, including open-source, make valuable projects for students in both secondary and collegiate classes, who can use them to develop expertise in other software systems. We hope that readers of this paper might be motivated to explore the use of their own favorite software system, open-source or not, to which they have access and of which they have their own expertise.

References

[1] Richard Klima and Neil Sigmon, *Cryptology, Classical and Modern, with Maplets*, Taylor & Francis, Boca Raton, FL, 2012.

[2] Neil Sigmon, 2022. Maplet Download Page for Shift and Vigenère Ciphers with Maplets. Available at: https://sites.radford.edu/~npsigmon/classicalcryptography/paper.html.