Science Olympiad Codebusters

Guide for Coaches, Event Supervisors and Students

John Toebes

codebusters@toebes.com

Updated for the 2023-2024 season.

Disclaimer:

This document is based on the Draft rules. The official rules released in September 2023 on the Science Olympiad website will be the official rules and used for all competitions.

1. Goals

Kids are fascinated by secret writing and more importantly excited by breaking those codes. The process of breaking the codes involves complex pattern matching and teaches kids skills which are critical to software development and science in general.

- Seeing patterns
- Frequency Analysis
- Quick guessing and trusting instinct
- Backtracking to correct errors

More importantly the process of solving ciphers in a fun style gets the kids addicted to ciphers in general.

2. Overview of Cipher Usages

Cipher map per year based on experiences from previous years.

Cipher	Type	2018-2019	2019-2020 Reg.	2019-2020 State Nat	2020-2021 Reg.	2020-2021 State Nat	2021-2022 Reg.	2021-2022 State Nat	2022-2023 Reg.	2022-2023 State Nat	2023-2024 Reg.	2023-2024 State Nat
Dancing Men	Monoalphabetic Symbol	D										
AtBash	Monoalphabetic		ED		ED		ED		ED		ED	
RSA	Numeric Math			IM								
Running Key	Polyalphabetic			ED								
Caesar	Monoalphabetic	D	ED		D		D		D		D	
Aristocrat	Monoalphabetic	DC	DC		DC		DC		DC		DC	
Aristocrat Misspelled	Monoalphabetic	DC	DC		DC		DC		DC		DC	
Patristocrat	Monoalphabetic	DC	DC		DC		DC		DC		DC	
Xenocrypt	Monoalphabetic Language	DC (0 or 1)	DC (0 or 1)	DC (1+)	DC (0 or 1)	DC (1+)	DC (0 or 1)	DC (2+)	DC (0 or 1)	DC (2 +)	DC (0 or 1)	DC (2 +)
Hill 2x2	Polyalphabetic Math	EDM		EDM		EDM	EDM		EDM		EDM	
Hill 3x3	Polyalphabetic Math	DM		EDM		EDM		EDM		EDM		EDM
Affine	Monoalphabetic Math	EDC	ED	EDC	ED	EDC	ED	EDC	ED	EDC	ED	EDC
Vigenère	Polyalphabetic	ED	ED	EDC	ED	EDC	ED	EDC				
Baconian	Steganography		C		C		C		C		C	
Morbit	Tomogrammic				D	DC	D	DC	D	DC		
Pollux	Tomogrammic				D	DC	D	DC	D	DC		
Fractionated Morse	Tomogrammic								D		D	
Porta	Polyalphabetic						D	D	D	D	D	D
Railfence	Transposition						DC		DC			
Nihilist	Polybius Square										ED	EDC
Cryptarithm	Math								DM		DM	
Complete Columnar	Transposition										C	

Key:

D – Decode (Cipher Text given with or without a hint)

E – Encode (Plain Text given with an encoding key)

C – Cryptanalysis (Cipher Text given with some corresponding Plain Text)

M – Mathematical computation

I – Identification of components

– Division B Only

– Not used

3. Cipher Descriptions

3.a. Dancing Men [Monoalphabetic Symbol] Div A

A symbol-based cipher associated with a Sherlock Holmes book – *The Adventure of the Dancing Men* written by Sir Arthur Conan Doyle. If a student memorizes the symbols, this can be easily sight-read. We include an unlabeled set of the symbols in the reference guide of the test. 

References:

- https://en.wikipedia.org/wiki/The_Adventure_of_the_Dancing_Men
- https://www.dcode.fr/dancing-men-cipher
- https://www.omniglot.com/conscripts/dancingmen.htm
- https://twitter.com/NCSO_cb/status/846766212407345152

3.b. PigPen/Masonic [Monoalphabetic Symbol] Div A

The PigPen cipher is believed to have originated with the Hebrew Rabbis, but its biggest claim to fame is that it was used by the Knights Templar during the Christian Crusades. It was also heavily used by the Freemasons for keeping their records. As a result, it is also known as the Masonic or Freemason's cipher. This is a trivial cipher for students to encode or decode and the table is easily constructed from a couple of tic-tac-toe grids and Xs.

References:

- https://en.wikipedia.org/wiki/Pigpen_cipher
- https://www.dcode.fr/pigpen-cipher
- https://derekbruff.org/blogs/fywscrypto/historical-crypto/prying-open-the-pigpen-cipher/
- http://www.civilwarsignals.org/cipher/pigpencipher.html

3.c. Tap Code [Monoalphabetic] Div A

Also known as a knock code, the Tap Code Cipher was commonly used by prisoners of war in order to communicate with one another using pairs of up to 5 knocks to select a character from a 5x5 alphabet block (both C and K share the same spot). It only requires memorizing the letters AFLQV and then counting the knocks in the first set to advance along them and then count the knocks in the second set to figure out how much to advance the letter.

References:

- https://en.wikipedia.org/wiki/Tap_code
- https://www.dcode.fr/tap-cipher
- https://www.boxentriq.com/code-breaking/tap-code
- https://www.braingle.com/brainteasers/codes/tapcode.php

3.d. AtBash [Monoalphabetic] Div A Div B

An alphabet-based cipher originally used to encrypt the Hebrew Alphabet. It is easily adapted to other alphabets as it is simply all the letters reversed. This is a trivial cipher for students to encode or decode. One interesting property of this cipher is that by encoding text twice produces the original text. The AtBash cipher is used for Geocaching.

References:

- https://en.wikipedia.org/wiki/Atbash
- https://www.dcode.fr/atbash-mirror-cipher
- http://practicalcryptography.com/ciphers/atbash-cipher-cipher/
- http://rumkin.com/tools/cipher/atbash.php

3.e. RSA [Numeric math] (NO LONGER USED IN CODEBUSTERS -- MOVED TO CYBER SECURITY)

Algorithm used by modern computers to encrypt and decrypt messages. Relies on a private key for security and is based on finding factors of large composite numbers. Because the real algorithm requires numbers which cannot be computed on a calculator, we must do a very simplified one using two to four-digit primes. As such we can only have students find a single number and not text but are considering a simple chunking of a few characters.

References:

- https://en.wikipedia.org/wiki/RSA_(cryptosystem)
- https://simple.wikipedia.org/wiki/RSA_algorithm
- https://www.dcode.fr/rsa-cipher

3.f. Caesar [Monoalphabetic] Div A Div B

One of the earliest known and simplest cipher. Originally attributed to Caesar for his private correspondence. The ROT13 version (with a shift of 13) is in common use for computer software and online forums as a means of hiding spoilers. ROT13 is a reversable cipher such that applying it twice results in the original text (like the AtBash cipher). The Caesar cipher is also the basis for the Vigenère and Running Key ciphers. Both ROT13 and the general Caesar cipher are used for Geocaching.

References:

- https://en.wikipedia.org/wiki/Caesar_cipher
- https://youngtyros.com/2018/06/04/caesar-cipher/
- https://en.wikipedia.org/wiki/ROT13
- http://practicalcryptography.com/ciphers/caesar-cipher/
- https://www.dcode.fr/caesar-cipher
- https://www.dcode.fr/rot-13-cipher
- https://learncryptography.com/classical-encryption/caesar-cipher

3.g. Aristocrat [Monoalphabetic] Div A Div B Div C

Most commonly seen in newspapers as Cryptoquotes, an Aristocrat is the standard substitution cipher with the restriction that no letter maps to itself. This mapping of the alphabet can be random or in order to simplify the solving, could use a K1 (keyword in the plaintext alphabet), K2 (keyword in the Ciphertext alphabet), K3 (Keyword in alphabet, but plaintext/ciphertext is shifted) or K4 (different keywords in the plaintext and ciphertext alphabets).

References:

- https://en.wikibooks.org/wiki/Cryptography/Substitution_cipher
- https://youngtyros.com/2018/06/04/aristocrat-substitution-cipher/

- https://entertainment.howstuffworks.com/puzzles/cryptoquote-puzzles.htm
- https://cryptograms.puzzlebaron.com/tutorial.php
- http://rossinglish.blogspot.com/p/aristocrat.html
- https://toebes.com/Ciphers/Solving%20a%20K1%20Alphabet.htm
- http://www.cryptogram.org/resources/samples/Solving%20Sample%20A-1.pdf

3.h. Aristocrat Misspelled [Monoalphabetic] Div B Div C

Using the same mechanism, the words can be misspelled, or homonyms substituted. This increases the difficulty.

References:

- (See the Aristocrat section)

3.i. Patristocrat [Monoalphabetic] Div B Div C

The same rules as for Aristocrats applies here except that all spaces and punctuation is removed and only the letters are kept, separating them into groups for convenience. The most common grouping is 5 and has been adopted by the ACA as a standard.

References:

- https://youngtyros.com/2023/02/12/patristocrat-cipher/
- https://sites.google.com/site/bionspot/aristocrat-patristocrat-page
- https://codepenguincom.wordpress.com/tag/patristocrat/

3.j. Xenocrypt [Monoalphabetic Language] Div B Div C

Identical to Aristocrats, except the language of the plaintext is other than English. Although it is one of the lower attempted items, it often ends up being a differentiator and serves to bring in additional people with different skills to the team.

References:

- https://toebes.com/Ciphers/Samples/Code Busters 2018 Sample 9 Xenocrypt Solution.pdf
- https://youngtyros.com/2023/02/14/xenocrypts/

3.k. Hill 2x2 [Polyalphabetic math] Div C

The Hill cipher was invented by Lester S. Hill in 1929 which is based on linear algebra. It requires basic knowledge of matrix math to encode or decode. Given the encoding matrix, it is possible to determine the decryption matrix for a 2x2 Hill cipher. The process of encoding and decoding uses identical math taking the numeric equivalent of letters two at a time. This has an appeal to the math-oriented students.

References:

- https://en.wikipedia.org/wiki/Hill_cipher
- https://youngtyros.com/2023/02/19/affine-hill-cipher-types/
- http://practicalcryptography.com/ciphers/hill-cipher/
- https://www.geeksforgeeks.org/hill-cipher/
- https://crypto.interactive-maths.com/hill-cipher.html

- https://www.dcode.fr/hill-cipher
- https://massey.limfinity.com/207/hillcipher.pdf

3.l. Hill 3x3 [Polyalphabetic math] Div C

The 3x3 version of the Hill cipher uses the numeric equivalent of letters as triplets. The process of encoding and decoded is also identical to the 2x2 version. However, for a 3x3 the math for determining the decryption matrix from the encoding matrix is significantly more complex and probably beyond what we would expect to use at the event, so we provide the decryption matrix for decoding 3x3.

References:

- (See the Hill 2x2 references)
- https://www.dcode.fr/hill-cipher

3.m. Affine [Monoalphabetic math] Div B

The Affine cipher is a monoalphabetic substitution cipher where the mapping of letters is controlled by a function $(\mathrm{ax} + \mathrm{b}) \bmod \mathrm{m}$ where a and b are the keys of the cipher and m is the size of the alphabet. Typically, we use m=26 in order to overlap with the tables used for the Hill ciphers, but a value of 27 (with space for 27) leads to more interesting math. If $\mathrm{a} = 1$ then the Affine becomes a Caesar cipher with b indicating the shift value. If $\mathrm{a} = 25$ and $\mathrm{b} = 25$ then the Affine produces the AtBash cipher. The Affine cipher is used for Geocaching.

References:

- https://en.wikipedia.org/wiki/Affine_cipher
- https://youngtyros.com/2023/02/19/affine-hill-cipher-types/
- http://practicalcryptography.com/ciphers/affine-cipher/
- https://crypto.interactive-maths.com/affine-cipher.html
- https://www.dcode.fr/affine-cipher
- https://www.geeksforgeeks.org/implementation-affine-cipher/

3.n. Vigenère [Polyalphabetic] (Also known as the autokey cipher) Div A

The Vigenère cipher is basically a collection of Caesar ciphers based on the letters of a repeated keyword. It was invented in 1553 and resisted all attempts to break it until 1863. It was also used during the American Civil War. The Vigenère cipher is used for Geocaching.

References:

- https://en.wikipedia.org/wiki/Vigenère_cipher
- https://youngtyros.com/2023/02/19/vigenere-cipher-type/
- https://www.geeksforgeeks.org/vigenere-cipher/
- http://crypto.interactive-maths.com/vigenegravere-cipher.html
- https://cryptii.com/pipes/vigenere-cipher
- https://www.dcode.fr/vigenere-cipher
- https://en.wikipedia.org/wiki/Autokey_cipher
- http://practicalcryptography.com/ciphers/autokey-cipher/

3.o. Porta [Polyalphabetic] Div B Div C

The Porta cipher is a predecessor to the Vigenère cipher with only thirteen possibilities. It was invented by Giovanni Battista della Porta in 1563 and has the distinction of being the first cipher ever devised using a variable literal key.

References:

- https://en.wikipedia.org/wiki/Giovan_Battista_Bellaso
- https://youngtyros.com/2023/02/19/porta-cipher/
- https://www.cryptogram.org/downloads/aca.info/ciphers/Porta.pdf
- http://practicalcryptography.com/ciphers/porta-cipher/
- https://toebes.com/Flynns/Flynns-19260220.htm
- https://www.historyofinforMion.com/detail.php?entryid=3137
- https://www.dcode.fr/porta-cipher

3.p. Running Key [Polyalphabetic] NOT CURRENTLY USED

The Running Key cipher can be simply described as a version of Vigenère cipher except that the key is longer than the cipher. Typically, the plain text is encrypted against a well-known book starting at an agreed upon passage. It is considerably more secure than the Vigenère cipher but can still be cracked. It doesn't appear to have any modern-day usage.

References:

- https://en.wikipedia.org/wiki/Running_key_cipher
- http://practicalcryptography.com/ciphers/running-key-cipher/
- https://www.aclweb.org/anthology/P12-2016
- https://crypto.interactive-maths.com/other-examples.html
- http://www.crypto-it.net/eng/simple/running-key.html?tab=0

3.q. Baconian [Steganography] Div B Div C

Initially devised by Francis Bacon in 1605, it is unlike the other ciphers in that it works to conceal the message in the text presentation rather than the content. There are many representations including alternate visual representations (i.e., bold characters), sets of symbols, and the word Baconian which look like headlines. The Baconian cipher is used for Geocaching.

References:

- https://en.wikipedia.org/wiki/Bacon's_cipher
- https://youngtyros.com/2023/02/13/baconian-cipher/
- http://rumkin.com/tools/cipher/baconian.php
- https://mothereff.in/bacon
- http://practicalcryptography.com/ciphers/baconian-cipher/
- https://www.geeksforgeeks.org/baconian-cipher/
- https://toebes.com/Flynns/Flynns-19250425.htm
- https://www.dcode.fr/bacon-cipher

3.r. Morbit [Tomogrammic] NOT CURRENTLY USED

The name nominally comes from **MOR**se **B**inary dig**IT** as a binary representation of Morse Code. Created by converting the plain text to Morse Code and then taking the Morse code pieces in pairs, encoding them at a single character. There are several variants, the most common being 9 digits which stand for all possible combinations of **– .** and space (typically represented by **X**). In general, because Morse code is longer than the equivalent characters, a Morbit encoded cipher text will be longer (approximately 50%) than the corresponding plain text. There are some theories that Kryptos (https://en.wikipedia.org/wiki/Kryptos) uses Morbit for the still undeciphered K4. The Morbit cipher is used for Geocaching.

References:

- http://www.cryptogram.org/downloads/aca.info/ciphers/Morbit.pdf
- https://youngtyros.com/2023/02/19/fractionated-ciphers/
- http://acaencodedecode.appspot.com/cipher_forms/morbit.html
- https://www.dcode.fr/morbit-cipher
- http://members.aon.at/cipherclerk/Doc/Morse.html
- https://web.itu.edu.tr/~orssi/dersler/cryptography/Chap2-1.pdf

3.s. Pollux [Tomogrammic] NOT CURRENTLY USED

The Pollux cipher is like the Morbit cipher, except that the Morse pieces are taken off a single digit at a time. Typically, more than one character is assigned to stand for **– .** and space (typically represented by **X**). Because of the Morse encoding, a Pollux encoded cipher text will be significantly longer than the corresponding plain text (approximately 2-3 times the size). The Pollux cipher is used for Geocaching.

References:

- http://www.cryptogram.org/downloads/aca.info/ciphers/Pollux.pdf
- https://youngtyros.com/2023/02/19/fractionated-ciphers/
- https://asecuritysite.com/coding/pollux
- https://sites.google.com/site/geocachinghelppuzzle/home/ciphers/appearance
- http://members.aon.at/cipherclerk/Doc/Morse.html
- https://www.dcode.fr/pollux-cipher

3.t. Fractionated Morse [Tomogrammic] Div B Div C

The Fractionated Morse is a combination of the Morbit/Pollux ciphers but using a K1/K2 type alphabet for the mapping where a single cipher letter stands for three Morse pieces. It was invented by ACA member FIDDLE in 1960. The letters in the keyword alphabet are mapped against a defined set of patterns with the first letter mapping to **. . .** and the last letter (very often the letter Z if it isn't in the keyword) mapping to **XX–**. Note that since there are only 26 letters in the alphabet, there is no mapping to **XXX**.

References:

- https://www.cryptogram.org/downloads/aca.info/ciphers/FractionatedMorse.pdf
- https://youngtyros.com/2023/02/19/fractionated-ciphers/
- http://practicalcryptography.com/ciphers/classical-era/fractionated-morse/
- https://www.dcode.fr/fractionated-morse

- https://sites.google.com/site/cryptocrackprogram/user-guide/cipher-types/substitution/fractionated-morse

3.u. Railfence [Transposition] (Also called a zigzag cipher) NOT CURRENTLY USED

The Railfence cipher works not by substituting letters, but by changing the order of the letters by putting them into a pattern and then reading them out in a different order.

References:

- https://www.cryptogram.org/downloads/aca.info/ciphers/Railfence.pdf
- https://youngtyros.com/2023/02/17/railfence-and-redefence-cipher/
- https://en.wikipedia.org/wiki/Rail_fence_cipher
- https://www.dcode.fr/rail-fence-cipher
- https://crypto.interactive-maths.com/rail-fence-cipher.html
- http://rumkin.com/tools/cipher/railfence.php

3.v. Cryptarithm [math] Div B Div C

Cryptarithms are a great cross between ciphers and mathematics. In them mathematical formulas with letters taking the place of their corresponding numbers are given to solve. The answer is driven from sorting the mappings of the letters. The actual origin of Cryptarithms is not quite known as they have been seen as far back as 1864. Although Cryptarithms can be done in any base, for Science Olympiad we will be sticking to Base 10 which means that any problem with have 10 unique letters.

References:

- https://en.wikipedia.org/wiki/Verbal_arithmetic
- https://youngtyros.com/2023/02/19/cryptarithms/
- https://www.futurelearn.com/info/courses/recreational-math/0/steps/43523
- https://nrich.maths.org/cryptarithms
- https://www.dcode.fr/cryptarithm-solver
- http://www.arml2.com/arml_2017/public_power_contest/contest_archive/Fall_2016/ARMLPower_Fall_2016_reading.pdf

3.w. Nihilist [Polybius Square] (Also called a Nihilist Substitution cipher) Div B Div C

The Nihilist cipher works by first building a Polybius Square based on a keyword with an alphabet that skips the letter J. Based on this square, each letter gets a unique 2 digit number based on the row/column that it appears. A separate key is then used to encode each character of the plain text by adding the value for that letter to the value associated with the corresponding key in order to generate an encoding number. As such, each cipher text value will be a number from 22 to 110.

References:

- https://www.cryptogram.org/downloads/aca.info/ciphers/NihilistSubstitution.pdf
- https://toebes.com/Flynns/Flynns-19250328.htm
- http://www.crypto-it.net/eng/simple/nihilist.html
- https://en.wikipedia.org/wiki/Nihilist_cipher
- https://www.dcode.fr/nihilist-cipher
- https://cryptii.com/pipes/nihilist-cipher

- https://asecuritysite.com/cipher/nihlist

3.x. Complete Columnar [Transposition] Div B Div C

The Complete Columnar cipher works by writing the cipher into a rectangular block by filling the rows, assigning a key at the top of the columns and then reading the cipher text out by columns in the order of the key.

References:

- https://www.cryptogram.org/downloads/aca.info/ciphers/CompleteColTransposition.pdf
- https://en.wikipedia.org/wiki/Transposition_cipher#Columnar_transposition
- https://www.dcode.fr/columnar-transposition-cipher
- https://www.geeksforgeeks.org/columnar-transposition-cipher/
- https://crypto.interactive-maths.com/columnar-transposition-cipher.html
- https://rumkin.com/tools/cipher/columnar-transposition/

4. Caesar

Div A Div B

With a Caesar cipher, there are three strategies depending on the Cipher Text. Fortunately, you can use the Vigenère table to do this lookup.

1. If you have a single letter word, it is likely to be either A or I, so determine the offset from the letter in the Cipher text and use that mapping to evaluate any other word in the cipher. If it reads correctly, then you can proceed to decode the remainder of the text.
2. If there is a double letter word, a simple trick is to test it quickly which requires looking up only eight characters: Six letters mapping the beginning (A B I M O U) and two letters at the end (O E). The letters are for the beginning and for the end. The starting letters Match against As/At/An/Am, Be/By, In/It/Is/If, Me/My, Of/Or/On, and Up/Us. The ending letters Match against dO/gO/nO/sO/tO and hE/wE. First look up the match for the starting letters against ABIMOU and see what the secondary letter would make the word be. Do the same for the two ending letters (OE) and see what the corresponding starting letter would be. With whatever offset produces the most logical words, test another word and make sure it makes sense. If it reads correctly then proceed to decode the remainder of the text.
3. In the case where there are no single or double letter words, it is necessary to brute force doing the lookup. Start with the second row of the table and go through the alphabet decoding characters one at a time until a word makes sense. An alternate way to do it is to write the cipher text on the page and then write the subsequent letters one below another one column at a time until you see a word make sense. For example, starting with RIK you could do:

```
RIK
SJL
TKM
ULN
VMO
WNP
XOQ
YPR
ZQS
ART
```

To see that the word is ART and the offset is 9.

5. Aristocrat

Div A Div B Div C

5.a. General Solving Rules

In general, the strategy for an Aristocrat is:
Fill in letters from any clues you are given
Look for single letter words which will generally be **A** or **I**
Check the frequency. The most common letters in English are **ETAOIN**.
Look for contractions (**DON'T**, **DOESN'T**)
Look for two and three letter words
Look for patterns "**IT IS**" and "**THAT**" are good ones
Look for double letters

A much more detailed guide can be found on Puzzle Baron's Cryptograms site at
https://cryptograms.puzzlebaron.com/tutorial.php

5.b. Solving with a K1 Alphabet

Sometimes an Aristocrat or Patristocrat will be encoded with a K1 alphabet instead of a random alphabet. This can make it much easier to solve once you have identified a few letters.
To understand what this means you must look at how the letters are chosen to replace the original text. This process goes as follows:

- When creating the encryption, pick a code word or phrase. For example, we choose a phrase of "**ALPHABET SOUP**" as our encryption code word.

Eliminate all duplicate letters in the phrase. In this case the letter **A** appears twice (once at the start and after the **H**) and the letter **P** appears twice (third letter and at the end). We also drop any spaces and punctuation to end up with **ALPHBETSOU**
Pick an offset in the alphabet to place the code word. In this example, we will start at offset 5 meaning that we shift the alphabet by 5 characters. This means that we will map the letter **A** to **F**, **L** to **G** etc. This means the word **THE** would be encoded as **LIK**

	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
Frequency																										
Replacement						A	L	P	H	B	E	T	S	O	U											

Next, we fill in the remainder of the alphabet starting at the end of the phrase with the start of the alphabet and wrapping back to the beginning to use up all the characters. However, in this case since we already used the letters A, B and E in our phrase, we would start with **C D F G** etc. This gives us a mapping of

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

Applying the knowledge

To see how this would be useful, let's take a simple Aristocrat which was encoded with a K1 alphabet. We know that because of the K1 in the replacement table.

MQKAI FXLA MVRUI DRBQ BQI DXAUN.

RB'M K MFVHXU XO OARIWNMQRY KWN YIKJI.

K1	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
Frequency	4	3		2		2		1	6	1	4	1	5	3	2		4	5			3	2	2	4	2	
Replacement																										

Taking a quick look, the apostrophe and single letter word at the start of the second sentence suggests **IT'S A**.

MQKAI FXLA MVRUI DRBQ BQI DXAUN.
S A   S I   IT T

RB'M K MFVHXU XO OARIWNMQRY KWN YIKJI.
IT'S A S   I   S I A   A .

K1	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
Frequency	4	3		2		2		1	6	1	4	1	5	3	2		4	5			3	2	2	4	2	
Replacement		T									A		S					I								

The three-letter word **BQI** starts with a **T** and the **I** is the most frequent letter, so we can assume that it is **THE**. Filling that in gives us

MQKAI FXLA MVRUI DRBQ BQI DXAUN.
SHA E   S I E ITH THE

RB'M K MFVHXU XO OARIWNMQRY KWN YIKJI.
IT'S A S   IE SHI A   EA E.

K1	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
Frequency	4	3		2		2		1	6	1	4	1	5	3	2		4	5			3	2	2	4	2	
Replacement		T							E		A		S				H	I								

This is where we get a huge break because we know that it is a K1 alphabet. If you look at the **E A** and **S** you notice that the **A** is after the **E** and the **S** is only one letter away from the **A**. This tells us that they must be part of the key phrase. Looking further we see the **HI** combination which we can guess is part of the remaining alphabet. Furthermore counting the letters after the **H I**, we try **J K L M N O P Q R** (skip the **S** because it was already used) **T** and see that it fits exactly in the space for **S** to **A** giving us a replacement alphabet of:

K1	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
Frequency	4	3		2		2		1	6	1	4	1	5	3	2		4	5			3	2	2	4	2	
Replacement	R	T							E		A		S				H	I	J	K	L	M	N	O	P	Q

MQKAI FXLA MVRUI DRBQ BQI DXAUN.
SHARE O R SMILE ITH THE ORL

RB'M K MFVHXU XO OARIWNMQRY KWN YIKJI.
IT'S A S M OL O   RIEN SHIP AN PEA E.

This fills in quite a bit for us and we can readily see that **D** must be **W**, **O** must be **F** and **N** must be **D** giving us:
MQKAI FXLA MVRUI DRBQ BQI DXAUN.
SHARE O R SMILE WITH THE WORLD

RB'M K MFVHXU XO OARIWNMQRY KWN YIKJI.
IT'S A S M OL OF FRIENDSHIP AND PEA E.

K1	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
Frequency	4	3		2		2		1	6	1	4	1	5	3	2		4	5			3	2	2	4	2	
Replacement	R	T		W					E		A		S	D	F		H	I	J	K	L	M	N	O	P	Q

We can look and know a couple of things. **P** must be **G** because of the single letter gap between the **F** and the **H**. The letters **B** and **C** must be in the key phrase because we started with the letter **D** after the phrase. But we also can see a couple of letters to substitute in the phase. **J** must be **C** to make the word **PEACE** and **FXLA** must be **YOUR**. This gives us:
MQKAI FXLA MVRUI DRBQ BQI DXAUN.
SHARE YOUR SMILE WITH THE WORLD

RB'M K MFVHXU XO OARIWNMQRY KWN YIKJI.
IT'S A SYM OL OF FRIENDSHIP AND PEACE.

K1	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
Frequency	4	3		2		2		1	6	1	4	1	5	3	2		4	5			3	2	2	4	2	
Replacement	R	T		W		Y			E	C	A	U	S	D	F	G	H	I	J	K	L	M	N	O	P	Q

Looking at this, we see that **V** must be between **T** and **W** (we already used the **U**) **X** must be between the **W** and the **Y** which is followed by **Z**. This leaves **H** to map to the letter **B**.
MQKAI FXLA MVRUI DRBQ BQI DXAUN.
SHARE YOUR SMILE WITH THE WORLD

RB'M K MFVHXU XO OARIWNMQRY KWN YIKJI.
IT'S A SYMBOL OF FRIENDSHIP AND PEACE.

K1	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
Frequency	4	3		2		2		1	6	1	4	1	5	3	2		4	5			3	2	2	4	2	
Replacement	R	T	V	W	X	Y	Z	B	E	C	A	U	S	D	F	G	H	I	J	K	L	M	N	O	P	Q

This gives us our solution and you can see that the keyword was "**BECAUSE**."
Note that just because it is a K1 alphabet doesn't mean that you must solve it that way, it just serves as a hint to make it easier.

5.c. Solving with a K2 Alphabet

5.d. Solving with a K3 Alphabet

With a K3 alphabet, both the source and replacement alphabets are the same. It is a bit harder to solve with a K3, but there are some hints that help you out.

To understand what this means you must look at how the letters are chosen to replace the original text. This process goes as follows:

- When creating the encryption, pick a code word or phrase. For example, we choose a keyword of "**MACHINERY**" as our encryption code word.

We then build up an alphabet starting with the keyword followed by all the other letters which weren't used:
MACHINERYBDFGJKLOPQSTUVWXZ
Pick an offset to shift the second alphabet by. If we pick an offset of 1 then we get a mapping like:
MACHINERYBDFGJKLOPQSTUVWXZ *Ciphertext*
Z**MACHINERY**BDFGJKLOPQSTUVWX *Plaintext*
When you build out the replacement table, you will notice that the keyword mostly disappears

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

However, as you can see where the highlighted letters end up, because of the offset of 1, only the mapping of the **M** and the **Y** don't correspond to another letter of the keyword. But all of the other letters map to a closely shifted letter in groups. For example, you see **STUVW** and **TUVWX** as a nice clean set and **KL**/**JK** as another nice pair mapping
You want offsets that produce some overlap between the letters. The toughest K3 would be a 13-character phrase with an offset of 13 so that there is no overlap. Small offsets mean that they can see the sequence of characters more readily. For this example, we will use an offset of 3 which would give us
MACHINERYBDFGJKLOPQSTUVWXZ *Ciphertext*
WXZ**MACHINERY**BDFGJKLOPQSTUV *Plaintext*
This gives us a replacement table that looks like:

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

Which you should notice is quite a bit different from the offset of 1 replacement table. Also, six of the characters map to other letters in the replacement set which is what you would expect with a shift of 3 for a 9-character keyword.

Applying the knowledge

To see how this would be useful, let's take a simple Aristocrat which was encoded with a K3 alphabet. We know that because of the K1 in the replacement table. We are asked to solve the K3 keyword and are giving a solution box of:

WESVB RDDRWIWRSYV IYJ XYBIVRYBVV FSX IDB KINBJ MRWE IDB BZRJBYNB WEIW FSX EIZB IY IVVRLYHBYW WS KXQKRQQ.

K3	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
Frequency		11		4	4	2		1	10	3	3	1	1	2			3	8	5			7	8	4	8	2
Replacement																										

Our immediate guess is that the high frequency of **B** suggests that it map to **E**.

```
WESVB RDDRWIWRSYV IYJ XYBIVRYBVV FSX IDB KINBJ MRWE
    E                  E    E          E    E

IDB BZRJBYNB WEIW FSX EIZB IY IVVRLYHBYW WS KXQKRQQ.
  E E   E  E             E           E             .
```

K3	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
Frequency		11		4	4	2		1	10	3	3	1	1	2			3	8	5			7	8	4	8	2
Replacement		E																								

Looking at the next most frequent letter **I**, we might guess that it is a **T**, but that the **WEIW** in the middle would come out as **WETW** and the only real word that would match that pattern is **HATH**, so we go to the **W** as the next most frequent which gives the word **HATH** and filling in a lot of other possibilities

```
WESVB RDDRWIWRSYV IYJ XYBIVRYBVV FSX IDB KINBJ MRWE
TH  E     TAT     A    EA    E       A E  A E    TH

IDB BZRJBYNB WEIW FSX EIZB IY IVVRLYHBYW WS KXQKRQQ.
A E E   E  E THAT     HA E A  A      E T T         .
```

K3	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
Frequency		11		4	4	2		1	10	3	3	1	1	2			3	8	5			7	8	4	8	2
Replacement		E			H				A														T			

With this in place, a couple of letters are just staring at us. The **EVV** almost certainly means that **V** must be **S** and that **TS** must be **TO** which gives us:

```
WESVB RDDRWIWRSYV IYJ XYBIVRYBVV FSX IDB KINBJ MRWE
THOSE     TAT O S A    EAS  ESS  O  A E  A E    TH

IDB BZRJBYNB WEIW FSX EIZB IY IVVRLYHBYW WS KXQKRQQ.
A E E   E  E THAT  O  HA E A  ASS    E T TO        .
```

K3	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
Frequency		11		4	4	2		1	10	3	3	1	1	2			3	8	5			7	8	4	8	2
Replacement		E			H				A										O			S	T			

A few more letters become obvious, but it is worth pointing out that we see the **STUVW** in the table mapping to **O??ST**. This is probably a good clue that TU are likely to map to wither **PQ** with **R** appearing in the keyword or **QR** with **P** appearing in the keyword. Because **TU** don't appear in the cipher text, making a guess won't help us at this point in time. However, looking at the **TATROYS** strongly hints that it should be **TATIONS** which gives us:

```
WESVB RDDRWIWRSYV IYJ XYBIVRYBVV FSX IDB KINBJ MRWE
THOSE I  ITATIONS AN   NEASINESS  O  A E  A E   ITH
```

IDB BZRJBYNB WEIW FSX EIZB IY IVVRLYHBYW WS KXQKRQQ.
A E E I EN E THAT O HA E A ASSI N ENT TO I .

K3	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
Frequency		11		4	4	2		1	10	3	3	1	1	2			3	8	5			7	8	4	8	2
Replacement		E			H				A									I	O			S	T		N	

We don't get much help with the K3 here, but several words are just begging to be filled in:
A few more letters become obvious, but it is worth pointing out that we see that **IDD**ITATIONS, **X**NEASINESS and ASSI**L**N**H**ENT must be IRRITATIONS, UNEASINESS and ASSIGNMENT respectively which gives us:

WESVB RDDRWIWRSYV IYJ XYBIVRYBVV FSX IDB KINBJ MRWE
THOSE IRRITATIONS AN UNEASINESS OU ARE A E ITH

IDB BZRJBYNB WEIW FSX EIZB IY IVVRLYHBYW WS KXQKRQQ.
ARE E I EN E THAT OU HA E A ASSIGNMENT TO U I .

K3	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
Frequency		11		4	4	2		1	10	3	3	1	1	2			3	8	5			7	8	4	8	2
Replacement		E		R	H			M	A			G						I	O			S	T	U	N	

This tells us a little more about the K3. The **HI** mapping to **MA** strongly hints that one or more of those characters are in the keyword. The **VWXY** mapping to **STUN** tells us that either **Y** or **N** (or both) are in the keyword. But we still have a few more obvious letters to fill in: AN**J** must be AND, **F**OU must be YOU, HA**Z**E must be HAVE and E**Z**I**J**EN**N**E must be EVIDENCE which gives us:

WESVB RDDRWIWRSYV IYJ XYBIVRYBVV FSX IDB KINBJ MRWE
THOSE IRRITATIONS AND UNEASINESS YOU ARE ACED ITH

IDB BZRJBYNB WEIW FSX EIZB IY IVVRLYHBYW WS KXQKRQQ.
ARE EVIDENCE THAT YOU HAVE AN ASSIGNMENT TO U I .

K3	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
Frequency		11		4	4	2		1	10	3	3	1	1	2			3	8	5			7	8	4	8	2
Replacement		E		R	H	Y		M	A	D		G		C				I	O			S	T	U	N	V

A couple of obvious things come out from looking at this. We know for certain that the letter **N** or **Y** (or both) is a part of the keyword by looking at the **VWXYZ** mapping to **STUNV**. We also know that **VWXZ** are not part of the keyword. We will solve the keyword in a minute, but the letters we filled in pretty much give away the remainder of the cipher:

WESVB RDDRWIWRSYV IYJ XYBIVRYBVV FSX IDB KINBJ MRWE
THOSE IRRITATIONS AND UNEASINESS YOU ARE FACED WITH

IDB BZRJBYNB WEIW FSX EIZB IY IVVRLYHBYW WS KXQKRQQ.
ARE EVIDENCE THAT YOU HAVE AN ASSIGNMENT TO FULFILL.

K3	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
Frequency		11		4	4	2		1	10	3	3	1	1	2			3	8	5			7	8	4	8	2
Replacement		E		R	H	Y		M	A	D	F	G	W	C			L	I	O			S	T	U	N	V

To figure out the K3 keyword, we need to take a quick look at the letters in the mapping to see what we can group up. We start by collecting the groups of letters that are obvious sequences and write them down.

A B C D E F G H I JKL M N O P Q R S T U VWX Y Z
  E   R H Y   M A DFG W C   L I O   STU N V

Since both sequences must match, we can build a few more clusters by lining up what we know under them. Using three lines of the sequence helps to align them quicker. What you are looking for is where letters must line up relative to others and figure out the shift of the letters. Once you know the shift for certain, you can use it to place obvious letters. With our basic set gathered, we write down the letters under them that we know. If you don't know something, a ? is pretty useful because it MUST be some letter, you just don't know what it is, but when you do find out, it fills it in for another space.

A B C D E F G H I JKL M N O P Q R S T U VWX Y Z
  E   R H Y   M A DFG W C   L I O   STU N V
  H   I M N   RY? T ?   G A ?   O?? C S

Next, we pull out the ones which we know for certain are in the keyword and push them together. The **RY** is certainly in the keyword because of the **JKL**/**DFG**. It also tells us the **E** is in the keyword. So, a little rewrite of the letters pushing a few of the ones we know are together.

A B C D E F G H I **JKL** M N O P **QSTUVWXZ** R Y
? E ? R H Y ? M A **DFG** W C   **LO STUV** I N
? H ? I M N ? W ? **RY?** T ?   **G? O??S** A C

It is clear at this point in time that the sequence **JKLOPQSTUVWXZ** is the end of the list so we can reorganize it as

A B C D E F G H I **JKLOPQSTUVWXZ** M N R Y
? E ? R H Y ? M A **DFG??LO??STUV** W C I N
? H ? I M N ? W ? **RY???G???O??S** T ? A C

We fill in the ones we now know - **JK** must correspond to **OP** because it is right before the **LO**

A B C D E F G H I JKLOPQSTUVWXZ M N R Y
? E ? R H Y ? M A DFG**JK**LO**PQ**STUV W C I N
? H ? I M N ? W ? RY?**DF**GJ**KL**OPQS T ? A C

Moving backwards we can pull in **DFG** in front of **JKL** filling from the others we know

A B C **DFG**JKLOPQSTUVWXZ E H I M N R Y
? E ? **RY?**DFGJKLOPQSTUV H M A W C I N
? H ? **IN?**RY?DFGJKLOPQS M W ? T ? A C

Now we can go forwards because we know **W** need to be after **TUV** on the second line and it tells us that the first letter of the keyword is **M**

A B C DFGJKLOPQSTUVWXZ**M** E H I N R Y
? E ? RY?DFGJKLOPQSTUV**W** H M A C I N
? H ? IN?RY?DFGJKLOPQS**T** M W ? ? A C

To figure what goes after **M**, we have only two letters which aren't identified, **A** and **C**. Since MC doesn't make a good word, we will go with **MA** which will have to map to **X** and subsequently **U**

B C DFGJKLOPQSTUVWXZM**A** E H I N R Y
E ? RY?DFGJKLOPQSTUVW**X** H M A C I N
H ? IN?RY?DFGJKLOPQST**U** M W ? ? A C

We also know that after **STU** on the bottom line, we need to have **VWXZ** but only the **W** is known (which is good because it also puts the **M** in the right place).

B C DFGJKLOPQSTUVWXZMA**-H** E I N R Y
E ? RY?DFGJKLOPQSTUVWX**-M** H A C I N
H ? IN?RY?DFGJKLOPQSTU**-W** M ? ? A C

From looking at this, we know the shift of the letters to be 3, so we can order the letters that we have remaining in place. The **H** in the second row must be three after the **H** in the first row. The **A** in the second row must be three after the **A** in the first row. Then the **I** in the second row should be three after the **I** in the first row giving us:

```
B C DFGJKLOPQSTUVWXZMA-HI-ER N Y
E ? RY?DFGJKLOPQSTUVWX-MA-HI C N
H ? IN?RY?DFGJKLOPQSTU-W?-MA ? C
```

With the three **C**'s staring at us and a gap of three after the **MA**, **HI** and **ER**, we know that they have to go there so we put them in place.

```
B DFGJKLOPQSTUVWXZMACHINERY
E RY?DFGJKLOPQSTUVWX?MACHIN
H IN?RY?DFGJKLOPQSTU?W??MAC
```

At this point, the answer comes out as being **MACHINERY**.

6. Patristocrat

Div B Div C

6.a. General solving approach

In general, there are three basic strategies for solving a Patristocrat. Because there are no word spacings, many of the Aristocrat rules don't apply

1. Frequency is your friend. Look for the high frequency letters to match them with **ETAOIN**.

6.b. Solving with a K1 Alphabet

Question #5 on [2018 Sample 7] is a Patristocrat with a [K1 alphabet] and a simple clue:

5) **[250 Points]** Solve this K1 key encoded Patristocrat which is a quote by Barbara Tuchman in "*Can History Be Served Up Hot?*" and has the word **THE** in it three times and ends with **HEARD**.

```
JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
OJFSV JKHOI NFHCD L
```

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement																										

Here's one approach to solving it that focuses more heavily on the K1 key as a major clue.

1. Since we are given that **FHCDL** corresponds to **HEARD**, we can go through and make that substitution globally as well as put it in our replacement below

```
JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
 HE   RE  R DED A         E   HER H A   R  D R
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
 E D  HE RE E   H E R   E A    RE    H  H  E
OJFSV JKHOI NFHCD L
  H     E    HEAR D
```

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement			A	R		H		E				D														

2. Next, we look for the locations of **THE** and see that **J** just correspond to **T** which is good because **J** also has a high frequency count. We fill that Information in to get:

```
JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
THE   RE  R DED A  T      E T HERTH A   R   D R
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
 E DT HETRE E  TH E R   E A    RE    H  H  E
OJFSV JKHOI NFHCD L
 TH   T E    HEAR D
```

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement			A	R		H		E		T		D														

3. Looking at the key, we have quite a few clues. Because we have the **H** and **E** between the **R** and the **T**, we know that all of them are part of the key word. Along with that since the **A** is right before the **R**, we also know that it is part of the K1 key word. Looking at the unused letter for **A**, we could make a good guess that it is the letter **Z** which means that our keyword goes from at least **B** to **J** as **?AR?H?E?T** which we mark giving us:

```
JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
THE   RE  R DED A  T      E T HERTH A   R   D R
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
 E DT HETRE E  TH E R   E A    RE T  H  H  E
OJFSV JKHOI NFHCD L
 TH   T E    HEAR D
```

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement	Z		A	R		H		E		T		D														

4. Looking at the unused letter at **T** and counting backwards from **Z** we could make a good guess that **T** stands for **Q** which would mean that **TUVWXYZ** maps to **QSUVWXY** because **R** and **T** were already used. Filling this in gives us:

```
JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
THEU  RE  R DED A ST S    E T HERTH A  UR   D R
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
 E DT HETRE E  TH E R   EVA    REST WH  H  E  W
OJFSV JKHOI NFHCD L
 TH U T E    HEAR D
```

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement	Z		A	R		H		E		T		D								Q	S	U	V	W	X	Y

5. Looking at the remaining letters between D and Q we see that there must be three gaps. Likewise, we can also see that the 4th letter much come from the K slot. This gives us K could be either B or C, M must be one of FGIJ (H was already used), N is GIJK, etc. We mark the Information down and then look to see if any of them make sense.

```
JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
THEU  RE  R DED A ST S    E T HERTH A  UR   D R
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
 E DT HETRE E  TH E R   EVA    REST WH  H  E  W
OJFSV JKHOI NFHCD L
 TH U T E    HEAR D
```

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement	Z		A	R		H		E		T	bc	D	fgij	gijk	ijkl	jklm	klmn	lmno	mnop	Q	S	U	V	W	X	Y

6. Some obvious ones stand out. Looking at the **N** right before **HEARD** at the end, we have choices of **GIJK**. Since we know of few words that end in either **I** or **J** we know it must be a **G** or a **K**. Looking back a bit more we see the **HRRX** which is **ElmnolmnoW** giving us choices of **ELLW EMMW ENNW** or **EOOW**. The only one of those which makes sense is **ELLW** which means **R** must be **L**. Filling these in gives us:

```
JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
THEU  RE  R DED A ST S    E T HERTH A  UR  LD R
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
 E DT HETRE E  TH E R   EVAL   REST WH  H ELLW
OJFSV JKHOI NFHCD L
 TH U T E   gkHEAR D
```

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement	Z		A	R		H		E		T	bc	D	fgij	gk	ijkl	J	K	L	mnop	Q	S	U	V	W	X	Y

7. Now that we filled in the **K** as a substitution for **Q**, we know that **N** must be a **G** and **O** must be an **I** and **M** must be an **F**. Filling these in gives us:

```
JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
THEU  RE  R DED A STIS    E T HERTH A  UR  LDFR
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
IE DT HETRE EI TH E RI  EVALF  REST WHI H FELLW
OJFSV JKHOI NFHCD L
ITH U T EI  GHEAR D
```

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement	Z		A	R		H		E		T	bc	D	F	G	I	J	K	L	mnop	Q	S	U	V	W	X	Y

8. Looking near the end of the second line we see **XFOEFMHRR** mapping to **WHI?HFELL** and could only be **WHICH FELL** meaning that **E** must be a **C**, which this also tells us that **K** must be **B** (because that was the only letter left). Filling that in gives us:

```
JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
THEU  REC R DED A STIS    E T HERTH A  UR  LDFR
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
IE DT HETRE EI TH E RI  EVALF  REST WHICH FELLW
OJFSV JKHOI NFHCD L
ITH U TBEI  GHEAR D
```

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement	Z		A	R	C	H		E		T	B	D	F	G	I	J	K	L	mnop	Q	S	U	V	W	X	Y

9. At this point, there are only four letters which haven't been mapped: **M N O P**. Looking at the end of the phrase we see **KHOIN** mapping to **BEI?G** leading us to the conclusion that **I** must map to **N**. Filling that in gives us:

```
JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
THEUN REC R DED A STISN  NE T HERTH AN UR  LDFR
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
IENDT HETRE EINTH E RI  EVALF  REST WHICH FELLW
```

OJFSV JKHOI NFHCD L
ITH U TBEIN GHEAR D

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement	Z		A	R	C	H		E	N	T	B	D	F	G	I	J	K	L	mop	Q	S	U	V	W	X	Y

---

10. With only **M O** and **P** left, we look at the start of the phrase and see **VIDHESDL** mapping to **UNREC?RD** with the only possible mapping for **S** being **O**. Filling that in gives us:

JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
THEUN RECOR DED A STISN ONEOT HERTH ANOUR OLDFR
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
IENDT HETRE EINTH E RI EVALF OREST WHICH FELLW
OJFSV JKHOI NFHCD L
ITHOU TBEIN GHEAR D

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement	Z		A	R	C	H		E	N	T	B	D	F	G	I	J	K	L	O	Q	S	U	V	W	X	Y

---

11. There are only three letters left to fill in and we could just leave it this way to get 50 points off on the test, but it doesn't take much of a guess with only **M** and **P** left, we read **THE UNRECORDED ?AST** and quickly come to the conclusion that **B** must stand for **P** to read as **THE UNRECORDED PAST** which also means that **G** must be **M**. This works well as we see the K1 key word is **PARCHMENT**. We complete it with filling them in as:

JFHVI DHESD LHLBC UJOUI SIHSJ FHDJF CISVD SRLMD
THEUN RECOR DEDPA STISN ONEOT HERTH ANOUR OLDFR
OHILJ FHJDH HOIJF HBDOG HWCRM SDHUJ XFOEF MHRRX
IENDT HETRE EINTH EPRIM EVALF OREST WHICH FELLW
OJFSV JKHOI NFHCD L
ITHOU TBEIN GHEAR D

K1	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
Frequency		2	4	9	2	9	1	15	7	10	1	5	3	1	7			4	7		3	3	1	2		
Replacement	Z	P	A	R	C	H	M	E	N	T	B	D	F	G	I	J	K	L	O	Q	S	U	V	W	X	Y

THE UNRECORDED PAST IS NONE OTHER THAN OUR OLD FRIEND THE TREE IN THE PRIMEVAL FOREST WHICH FELL WITHOUT BEING HEARD.

7. Xenocrypt

Div B Div C

Question #2 on [2018 Sample 9](#) is a Xenocrypt:

6) **[300 Points]** Solve this Xenocrypt which is a quote by Albert Einstein in Spanish.

MDJ FK FNFACKR IR FV KD CJQIUQCDK ADIFJD MF QIZKYQJ VRWJF KRV MFADV; FV KD YIQUD ADIFJD.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	4		3	10		10			6	6	7		3	1				5	4			2	5	1		2	1
Replacement																											

Here's one approach to solving it. Note that it is helpful to understand the most common words in Spanish. Wikipedia has a nice one at https://en.wikipedia.org/wiki/Most_common_words_in_Spanish with 100 words that are worth studying and recognizing.

1. Looking at the frequency, we see that both **D** and **F** are high frequency letters, so we will assume that they are ***E*** and ***A*** which are the most frequently used letters in Spanish. However, given that they are both the same, we must look at the usage. Looking at the two-letter words using them we see **FV KD**. Since there are almost no two letter words that start with ***A***, we can make a good guess that the **F** must be an ***E*** leaving **D** to stand for ***A***.

```
MDJ FK FNFACKR IR FV KD CJQIUQCDK ADIFJD MF QIZKYQJ
 A  E  E E          E   A        A   A  E A  E
VRWJF KRV MFADV; FV KD YIQUD ADIFJD.
    E      E A   E   A     A  A E A
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	4		3	10		10			6	6	7		3	1				5	4			2	5	1		2	1
Replacement				A		E																					

2. Taking another look at the **KD** the most obvious two-letter Spanish word is ***LA***, (the most common Spanish word), so we will guess that **K** stands for ***L***.

```
MDJ FK FNFACKR IR FV KD CJQIUQCDK ADIFJD MF QIZKYQJ
 A  EL E E  L      E  LA        AL  A E A  E    L   
VRWJF KRV MFADV; FV KD YIQUD ADIFJD.
    E L    E A   E  LA     A  A E A
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	4		3	10		10			6	6	7		3	1				5	4			2	5	1		2	1
Replacement				A		E					L																

3. The next two-letter word that is interesting is **FV**. The most common two-letter Spanish words are ***ES***, ***EL*** and ***EN***, but since we already have **K** standing for ***L***. We must choose between ***ES*** and ***EN***. Looking at the use of it right after the semi-colon, we can guess that it is **ES** since few sentences would start with ***EN***. We will assume that **V** stands for ***S***. Additional confirmation comes from looking at the **KRV** which would be ***L?S*** and must be ***LOS*** (since ***E*** and ***A*** are already known). This gives us that **R** stands for ***O***.

```
MDJ FK FNFACKR IR FV KD CJQIUQCDK ADIFJD MF QIZKYQJ
 A  EL E E  LO  O ES LA        AL  A E A  E    L   
VRWJF KRV MFADV; FV KD YIQUD ADIFJD.
SO  E LOS  E AS  ES LA     A  A E A
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	4		3	10		10			6	6	7		3	1				5	4			2	5	1		2	1
Replacement				A		E					L								O				S				

4. Our remaining high frequency letters are **I J** and **Q**. Given that the eight most common Spanish letters are ***EAOSNRIL*** and we have used five of them, we can check to see if ***N R*** and ***I*** make sense for filling in for any of them. Looking at the **IR** as ***?O***, we can only see ***NO*** as the two-letter word with **I** standing for ***N***. Looking at the first word **MDJ** as ***?A?***, there are no common words that end as ***?AI*** so we can guess that **J** stands for ***R***.

```
MDJ FK FNFACKR IR FV KD CJQIUQCDK ADIFJD MF QIZKYQJ
 AR EL E E  LO NO ES LA  R N   AL  ANERA  E  N L  R
VRWJF KRV MFADV; FV KD YIQUD ADIFJD.
SO RE LOS  E AS  ES LA  N  A  ANERA
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	4		3	10		10			6	6	7		3	1				5	4			2	5	1		2	1
Replacement				A		E			N	R	L								O				S				

5. Another two-letter word stands out – **MF** as ***?E***. Another very common Spanish word is **DE** and since ***D*** has not been mapped, we will assume that **M** maps to ***D***. This works out well as it makes the first word be ***DAR***. Filling this in gives us:

```
MDJ FK FNFACKR IR FV KD CJQIUQCDK ADIFJD MF QIZKYQJ
DAR EL E E  LO NO ES LA  R N   AL  ANERA DE  N L  R
VRWJF KRV MFADV; FV KD YIQUD ADIFJD.
SO RE LOS DE AS  ES LA  N  A  ANERA
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	4		3	10		10			6	6	7		3	1				5	4			2	5	1		2	1
Replacement				A		E			N	R	L		D						O				S				

6. Coming back to the **Q** which we didn't map and remembering that we wanted to try ***I***, we look at usage and it fits nice with the last word on the first line, so we put it in to give us:

```
MDJ FK FNFACKR IR FV KD CJQIUQCDK ADIFJD MF QIZKYQJ
DAR EL E E  LO NO ES LA  RIN I AL  ANERA DE IN L IR
VRWJF KRV MFADV; FV KD YIQUD ADIFJD.
SO RE LOS DE AS  ES LA  NI A  ANERA
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	4		3	10		10			6	6	7		3	1				5	4			2	5	1		2	1
Replacement				A		E			N	R	L		D					I	O				S				

7. Looking at the ***?NI?A*** and the remaining letters, the only one which makes sense to be in front of the **N** is ***U*** so we will map **Y** to ***U***:

```
MDJ FK FNFACKR IR FV KD CJQIUQCDK ADIFJD MF QIZKYQJ
DAR EL E E  LO NO ES LA  RIN I AL  ANERA DE IN LUIR
VRWJF KRV MFADV; FV KD YIQUD ADIFJD.
SO RE LOS DE AS  ES LA UNI A  ANERA
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	4		3	10		10			6	6	7		3	1				5	4			2	5	1		2	1
Replacement				A		E			N	R	L		D					I	O				S			U	

8. Looking at the ***?ANERA*** we look for letters which remain and could make sense as a word. We haven't used ***TCMPBHQYVGFJZÑXKW***. Going through the letters one at a time we have ***TANERA***, ***CANERA***, ***MANERA***, ***PANERA***, etc. But the only one which makes sense is ***MANERA***, so we map A to ***M*** to give us:

MDJ FK FNFACKR IR FV KD CJQIUQCDK ADIFJD MF QIZKYQJ
DAR EL E EM LO NO ES LA RIN I AL MANERA DE IN LUIR
VRWJF KRV MFADV; FV KD YIQUD ADIFJD.
SO RE LOS DEMAS ES LA UNI A MANERA

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	4		3	10		10			6	6	7		3	1				5	4			2	5	1		2	1
Replacement	M			A		E			N	R	L		D					I	O				S			U	

9. At this point, we have 8 letters that haven't been filled in. Since this is a 300-point question, you can get 7 letters wrong and still get 50 points, so every letter from here on out is worth 50 points. Looking at **CJQIUQCDK** as ***?RIN?I?AL*** we see that the **C** is used twice in that word. With our unused letter list now at ***TCPBHQYVGFJZÑXKW***, we quickly try them as ***TRIN?ITAL***, ***CRIN?ICAL***, ***PRIN?IPAL*** etc. and stop as we see that it looks suspiciously like ***PRINCIPAL*** and that it looks good paired in ***PRINCIPAL MANERA*** so we assume **C** maps to ***P*** and **U** maps to ***C*** which gives us 5 more letters solved:

MDJ FK FNFACKR IR FV KD CJQIUQCDK ADIFJD MF QIZKYQJ
DAR EL E EMPLO NO ES LA PRINCIPAL MANERA DE IN LUIR
VRWJF KRV MFADV; FV KD YIQUD ADIFJD.
SO RE LOS DEMAS ES LA UNICA MANERA

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	4		3	10		10			6	6	7		3	1				5	4			2	5	1		2	1
Replacement	M		P	A		E			N	R	L		D					I	O			C	S			U	

10. With only 3 letters unmatched, on this 300-point question, we would score 250 points, so we could stop. However, there is no penalty for guessing and any one of the three being right would get us 50 more points. Our remaining unused letters are ***TBHQYVGFJZÑXKW***; running letters through the three remaining words, possible guess are ***EJEMPLO*** and ***EXEMPLO***. Since ***J*** is more common than ***X***, we will guess **N** maps to ***J***. The only letter that makes sense for ***IN?LUIR*** is ***F*** (***C*** was already taken, otherwise we would guess ***INCLUIR***) so **Z** must map to ***F***. Lastly ***SO?RE*** is the very common Spanish word ***SOBRE*** so we map **W** to ***B*** with a solution of:

MDJ FK FNFACKR IR FV KD CJQIUQCDK ADIFJD MF QIZKYQJ
DAR EL EJEMPLO NO ES LA PRINCIPAL MANERA DE INFLUIR
VRWJF KRV MFADV; FV KD YIQUD ADIFJD.
SOBRE LOS DEMAS ES LA UNICA MANERA

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	4		3	10		10			6	6	7		3	1				5	4			2	5	1		2	1
Replacement	M		P	A		E			N	R	L		D	J				I	O			C	S	B		U	F

DAR EL EJEMPLO NO ES LA PRINCIPAL MANERA DE INFLUIR SOBRE LOS DEMÁS; ES LA ÚNICA MANERA.

Translation: *Setting the example is not the main way to influence others; it's the only way*

8. Hill cipher 2x2

Div C

This utilizes matrix math in order to encode/decode groups of letters corresponding to the size of the matrix. For competition, you only need to worry about a 2x2 and 3x3 matrix. Note that if the message to encrypt is not a multiple of the size of the matrix, you add as many **Z**s to the remaining letter(s) to match the matrix size. You can typically assume a normal mapping alphabet such as:

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
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

However, sometimes the alphabet is longer by adding punctuation and even digits. If that is the case, you just need to know the size of the alphabet and use that instead of the 26 for all the modulus operations.
To encrypt you start with a key matrix either 2x2 or 3x3. Typically, this is chosen by letters to make it easier to remember. However, you can't use any combination of letters, the determinant of the matrix must be coprime with the size of the alphabet. This means that if you are making up your own examples, you need to check that the matrix is invertible, or the message would not actually be decryptable.

8.a. 2x2 Encryption

For example, we will use a 2x2 matrix of the string **AXLE** which would be encoded as
$\begin{pmatrix} A & X \\ L & E \end{pmatrix} = \begin{pmatrix} 0 & 2 \\ 11 & 4 \end{pmatrix}$
If we wanted to encode **CIPHERS**, we need to break it into groups of 2 as **CI PH ER SZ** and do a matrix multiplication. Note the letter Z at the end to make it be a group of 2.
When you do the math in this case you get:

$$\begin{pmatrix} A & X \\ L & E \end{pmatrix}\begin{pmatrix} C \\ I \end{pmatrix} \equiv \begin{pmatrix} 0 & 23 \\ 11 & 4 \end{pmatrix}\begin{pmatrix} 2 \\ 8 \end{pmatrix} \equiv \begin{pmatrix} 0 \times 2 + 23 \times 8 \\ 11 \times 2 + 4 \times 8 \end{pmatrix} \equiv \begin{pmatrix} 184 \\ 54 \end{pmatrix} \equiv \begin{pmatrix} 2 \\ 2 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} C \\ C \end{pmatrix}$$
$$\begin{pmatrix} A & X \\ L & E \end{pmatrix}\begin{pmatrix} P \\ H \end{pmatrix} \equiv \begin{pmatrix} 0 & 23 \\ 11 & 4 \end{pmatrix}\begin{pmatrix} 15 \\ 7 \end{pmatrix} \equiv \begin{pmatrix} 0 \times 15 + 23 \times 7 \\ 11 \times 15 + 4 \times 7 \end{pmatrix} \equiv \begin{pmatrix} 161 \\ 193 \end{pmatrix} \equiv \begin{pmatrix} 5 \\ 11 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} F \\ L \end{pmatrix}$$
$$\begin{pmatrix} A & X \\ L & E \end{pmatrix}\begin{pmatrix} E \\ R \end{pmatrix} \equiv \begin{pmatrix} 0 & 23 \\ 11 & 4 \end{pmatrix}\begin{pmatrix} 4 \\ 17 \end{pmatrix} \equiv \begin{pmatrix} 0 \times 4 + 23 \times 17 \\ 11 \times 4 + 4 \times 17 \end{pmatrix} \equiv \begin{pmatrix} 391 \\ 112 \end{pmatrix} \equiv \begin{pmatrix} 1 \\ 8 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} B \\ I \end{pmatrix}$$
$$\begin{pmatrix} A & X \\ L & E \end{pmatrix}\begin{pmatrix} S \\ Z \end{pmatrix} \equiv \begin{pmatrix} 0 & 23 \\ 11 & 4 \end{pmatrix}\begin{pmatrix} 18 \\ 25 \end{pmatrix} \equiv \begin{pmatrix} 0 \times 18 + 23 \times 25 \\ 11 \times 18 + 4 \times 25 \end{pmatrix} \equiv \begin{pmatrix} 575 \\ 298 \end{pmatrix} \equiv \begin{pmatrix} 3 \\ 12 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} D \\ M \end{pmatrix}$$

Which gives us an encoded string of **CCFLBIDM.**
Now a common question is how to quickly do a mod `26` using a non-scientific calculator. The easiest way to do it is to take the number (for example `184`) and divide it by `26` to get `7.0769231`. You can subtract out the integer portion to get `0.0769231` and then multiply that by `26` to get `2.0000006` (remember that it may not have the same precision as a scientific calculator), so we know that the remainder is `2` which corresponds to the letter **C**.

8.b. 2x2 Decryption

To decrypt you will need to determine the inversion of the 2x2 matrix. For a 2x2 there is a well-known solution:

$$A^{-1} \equiv \begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} \equiv \frac{1}{\det(A)}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix} \equiv \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

To simplify matters, the det(A) is given as a table on the resources page. Since there are only 13 possible values for A you map them as follows.

1	3	5	7	9	11	15	17	19	21	23	25
1	9	21	15	3	19	7	23	11	5	17	25

9. Hill cipher 3x3

Div C

For a 3x3 matrix of the string **PRACTICED** which would be encoded as

$$\begin{pmatrix} P & R & A \\ C & T & I \\ C & E & D \end{pmatrix} \equiv \begin{pmatrix} 15 & 17 & 0 \\ 2 & 19 & 8 \\ 2 & 4 & 3 \end{pmatrix}$$

If we wanted to encode **SPECIALS**, we need to break it into groups of 3 as **SPE CIA LSZ** and do a matrix multiplication. In this case we get

$$\begin{pmatrix} P & R & A \\ C & T & I \\ C & E & D \end{pmatrix}\begin{pmatrix} S \\ P \\ E \end{pmatrix} \equiv \begin{pmatrix} 15 & 17 & 0 \\ 2 & 19 & 8 \\ 2 & 4 & 3 \end{pmatrix}\begin{pmatrix} 18 \\ 15 \\ 4 \end{pmatrix} \equiv \begin{pmatrix} 15 \times 18 + 17 \times 15 + 0 \times 4 \\ 2 \times 18 + 19 \times 15 + 8 \times 4 \\ 2 \times 18 + 4 \times 15 + 3 \times 4 \end{pmatrix} \equiv \begin{pmatrix} 525 \\ 353 \\ 108 \end{pmatrix} \equiv \begin{pmatrix} 5 \\ 15 \\ 4 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} F \\ P \\ E \end{pmatrix}$$

$$\begin{pmatrix} P & R & A \\ C & T & I \\ C & E & D \end{pmatrix}\begin{pmatrix} C \\ I \\ A \end{pmatrix} \equiv \begin{pmatrix} 15 & 17 & 0 \\ 2 & 19 & 8 \\ 2 & 4 & 3 \end{pmatrix}\begin{pmatrix} 2 \\ 8 \\ 0 \end{pmatrix} \equiv \begin{pmatrix} 15 \times 2 + 17 \times 8 + 0 \times 0 \\ 2 \times 2 + 19 \times 8 + 8 \times 0 \\ 2 \times 2 + 4 \times 8 + 3 \times 0 \end{pmatrix} \equiv \begin{pmatrix} 166 \\ 156 \\ 36 \end{pmatrix} \equiv \begin{pmatrix} 10 \\ 0 \\ 10 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} K \\ A \\ K \end{pmatrix}$$

$$\begin{pmatrix} P & R & A \\ C & T & I \\ C & E & D \end{pmatrix}\begin{pmatrix} L \\ S \\ Z \end{pmatrix} \equiv \begin{pmatrix} 15 & 17 & 0 \\ 2 & 19 & 8 \\ 2 & 4 & 3 \end{pmatrix}\begin{pmatrix} 11 \\ 18 \\ 25 \end{pmatrix} \equiv \begin{pmatrix} 15 \times 11 + 17 \times 18 + 0 \times 25 \\ 2 \times 11 + 19 \times 18 + 8 \times 25 \\ 2 \times 11 + 4 \times 18 + 3 \times 25 \end{pmatrix} \equiv \begin{pmatrix} 471 \\ 564 \\ 169 \end{pmatrix} \equiv \begin{pmatrix} 3 \\ 18 \\ 13 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} D \\ S \\ N \end{pmatrix}$$

Which gives us an encoded string of **FPEKAKDSN.**

Decoding is done in the same manner, but the 3x3 decoding matrix will be provided. In this case we can decode the string **FPEKAKDSN** which was encoded using the string **PRACTICED** for which we will get the inverse matrix:

$$\begin{pmatrix} P & R & A \\ C & T & I \\ C & E & D \end{pmatrix}^{-1} \equiv \begin{pmatrix} 15 & 17 & 0 \\ 2 & 19 & 8 \\ 2 & 4 & 3 \end{pmatrix}^{-1} \equiv \begin{pmatrix} 1 & 25 & 20 \\ 16 & 7 & 16 \\ 4 & 0 & 9 \end{pmatrix}$$

Using this matrix, we proceed the same way as encoding breaking up in groups of 3 and do the matrix multiplications

$$\begin{pmatrix} 1 & 25 & 20 \\ 16 & 7 & 16 \\ 4 & 0 & 9 \end{pmatrix}\begin{pmatrix} F \\ P \\ E \end{pmatrix} \equiv \begin{pmatrix} 1 & 25 & 20 \\ 16 & 7 & 16 \\ 4 & 0 & 9 \end{pmatrix}\begin{pmatrix} 5 \\ 15 \\ 4 \end{pmatrix} \equiv \begin{pmatrix} 1 \times 5 + 25 \times 15 + 20 \times 4 \\ 16 \times 5 + 7 \times 15 + 16 \times 4 \\ 4 \times 5 + 0 \times 15 + 9 \times 4 \end{pmatrix} \equiv \begin{pmatrix} 460 \\ 249 \\ 56 \end{pmatrix} \equiv \begin{pmatrix} 18 \\ 15 \\ 4 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} S \\ P \\ E \end{pmatrix}$$

$$\begin{pmatrix} 1 & 25 & 20 \\ 16 & 7 & 16 \\ 4 & 0 & 9 \end{pmatrix}\begin{pmatrix} K \\ A \\ K \end{pmatrix} \equiv \begin{pmatrix} 1 & 25 & 20 \\ 16 & 7 & 16 \\ 4 & 0 & 9 \end{pmatrix}\begin{pmatrix} 10 \\ 0 \\ 10 \end{pmatrix} \equiv \begin{pmatrix} 1 \times 10 + 25 \times 0 + 20 \times 10 \\ 16 \times 10 + 7 \times 0 + 16 \times 10 \\ 4 \times 10 + 0 \times 0 + 9 \times 10 \end{pmatrix} \equiv \begin{pmatrix} 210 \\ 320 \\ 130 \end{pmatrix} \equiv \begin{pmatrix} 2 \\ 8 \\ 0 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} C \\ I \\ A \end{pmatrix}$$

$$\begin{pmatrix} 1 & 25 & 20 \\ 16 & 7 & 16 \\ 4 & 0 & 9 \end{pmatrix}\begin{pmatrix} D \\ S \\ N \end{pmatrix} \equiv \begin{pmatrix} 1 & 25 & 20 \\ 16 & 7 & 16 \\ 4 & 0 & 9 \end{pmatrix}\begin{pmatrix} 3 \\ 18 \\ 13 \end{pmatrix} \equiv \begin{pmatrix} 1 \times 3 + 25 \times 18 + 20 \times 13 \\ 16 \times 3 + 7 \times 18 + 16 \times 13 \\ 4 \times 3 + 0 \times 18 + 9 \times 13 \end{pmatrix} \equiv \begin{pmatrix} 713 \\ 382 \\ 129 \end{pmatrix} \equiv \begin{pmatrix} 11 \\ 18 \\ 25 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} L \\ S \\ Z \end{pmatrix}$$

Which gives us an encoded string of **SPECIALSZ**. Since we know that the **Z** at the end is padding, our answer is **SPECIALS**.

10. Affine cipher

Div B

The Affine cipher is a simple substitution cipher where each letter maps to exactly one other letter. Given an alphabet of size *m*, you need to have two key values *a* and *b* such that *a* and *m* are coprime (i.e., there is no positive divisor for both other than 1). If *a=1,* then the Affine cipher is a trivial Caesar cipher. Assuming *m=26* as, you will find most commonly, then the possible values for a will be 1, 3, 5, 7, 9, 11 ,15, 17, 19, 21, 23 and 25.

10.a. Encryption

To encrypt a letter, the formula is

`E(x) = (ax + b) mod m`

Assuming a normal alphabet such as:

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
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

We can pick a value of *a=7* and *b=42*.

Plaintext	S	C	I	E	N	C	E	O	L	Y	M	P	I	A	D
x	18	2	8	4	13	2	4	14	11	24	12	15	8	0	3
(7x+42)	168	56	98	70	133	56	70	140	119	210	126	147	98	42	63
(7x+42) mod 26	12	4	20	18	3	4	18	10	15	2	22	17	20	16	11
ciphertext	M	E	U	S	D	E	S	K	P	C	W	R	U	Q	L

10.b. Decryption by Formula (hard)

If you had a message and were given the values of *a* and *b*, you must apply a formula to build the decryption for each letter. The formula is

`D(x) = a-1 (x - b) mod m`

where a^{-1} is the modular multiplicative inverse of *a* mod *m*.

`1 = aa-1 mod m`

If we want to decrypt it, we must figure out multiplicative inverse of *a* mod *m*. There are some approximation ways to do it, but since there are only 26 values, we can brute force it to look for the one value of *t* where the result of *t*a* mod *m* = 1.

t	1	2	3	4	5	6	7	8	9	10	11	12	13
*t**7 mod 26	7	14	21	2	9	16	23	4	11	18	25	6	13

T	14	15	16	17	18	19	20	21	22	23	24	25	26
*t**7 mod 26	20	1	8	15	22	3	10	17	24	5	12	19	0

Based on this, we know that a^{-1} = 15 and we can proceed to decrypt.

ciphertext	M	E	U	S	D	E	S	K	P	C	W	R	U	Q	L
y	12	4	20	18	3	4	18	10	15	2	22	17	20	16	11
15(y - 42)	-450	-570	-330	-360	-585	-570	-360	-480	-405	-600	-300	-375	-330	-390	-465
15(y - 42) mod 26	18	2	8	4	13	2	4	14	11	24	12	15	8	0	3
Plaintext	S	C	I	E	N	C	E	O	L	Y	M	P	I	A	D

While this is possible to do, it requires a bit of trial and error to figure out the multiplicative inverse. As such, there are easier ways you could approach decryption if you do know some characters:

10.c. Decryption when you know some characters (Easier)

Sometimes you will be given the ciphertext and a couple of plain text letters. For example.
Suppose you were given the ciphertext of

GLIID MGNF NF J XNKGLY

and are told that the first word is `HELLO`.
You can start out by figuring out what the values of *a* and *b* are as follows:
We know that the characters map like this:

```
H(7)    =>  G(7)
E(4)    =>  L(11)
L(11)   =>  I(8)
L(11)   =>  I(8)
O(14)   =>  D(3)
```

To determine the values of a and b from the formula:
`Output = ax+b (mod 26)`
You only need to have two letters mapped. For convenience, we just pick the first two, write them as the formula and then solve for b initially: So, we have:

$$a * 7 + b\ (Mod\ 26) = 6$$
$$a * 4 + b\ (Mod\ 26) = 11$$

You can cancel out the *a* in both by multiplying each by the other a value. I.e., since the first is *a**7, and then second is *a**4 we multiply the first by 4 and the second by 7

$$4 * (a * 7 + b\ (Mod\ 26)) = 4 * 6$$
$$7 * (a * 4 + b\ (Mod\ 26)) = 7 * 11$$

Simplify them to get:

$$a * 28 + 4 * b\ (Mod\ 26) = 24$$
$$a * 28 + 7 * b\ (Mod\ 26) = 77$$

Don't worry about the mod 26 portion for now, we will handle it in a bit. Next, we need to subtract to cancel out the *a*. For convenience, subtract the smaller from the larger:

$$a * 28 + 7 * b\ (Mod\ 26) = 77$$
$$\underline{- a * 28 + 4 * b\ (Mod\ 26) = 24}$$
$$3 * b\ (Mod\ 26) = 53$$

Since the modulus is a one-way transformation, we need to take the mod of the right-hand side which is 1. So, we know that:

$$3 * b\ (Mod\ 26) = 1\ (or\ some\ other\ mod\ 26\ value)$$

To discover which value of b there is, simply compute the other modulus values and see which is a perfect multiple. We know it can't be 1 since b must be an integer.
Add `26` to get `27` and we observe that `27/3 = 9`
So, we now know that b=`9`. Now we need to solve for *a*. All we need to do is substitute 9 in for b in either of the formulas and repeat the same process again. For convenience we use the second formula since it is easier to see if something is a power of 4 vs. a power of 7

$$a * 4 + 9\ (Mod\ 26) = 11$$
$$a * 4 + 9 - 9\ (Mod\ 26) = 11 - 9$$
$$a * 4\ (mod\ 26) = 2$$

Just like before we look for a modulus value which is a perfect multiple of `4`. We know that it isn't `2`, so we add `26` to `2` to get `28`. Since `28/4 = 7` we know that *a*=`7`.

Now that you know that $b=9$ and $a=7$, you need to decode the remainder of the text.

G	L	I	I	D		M	G	N	F		N	F		J		X	N	K	G	L	Y
H	E	L	L	O															H	E	

Starting with the most frequent characters, calculate the mappings for ETAOIN. However, take note that the letter A is 0 which means that all you need to do is look up the value of b in the table to know the output letter.

Unencrypted	Value	7*x+9	7*x+9 mod 26	Encrypted
E	4	We already knew this		**L**
T	19	142	12	**M**
A	0	Don't bother to calculate, just look *b* up		**J**
O	14	We already knew this		**D**
I	8	65	13	**N**
N	13	100	22	**W**

With the 4 new letters we can fill in the cipher as follows.

G	L	I	I	D		M	G	N	F		N	F		J		X	N	K	G	L	Y
H	E	L	L	O		T		I			I			A			I		H	E	

A quick look at what was decoded so far suggests that it says something like **HELLO THIS IS A** so you can confirm it by encoding the letters **H** and **S** to confirm.

Unencrypted	Value	7*x+9	7*x+9 mod 26	Encrypted
H	7	58	6	**G**
S	18	135	5	**F**

That confirms the guess, so we fill them in.

G	L	I	I	D		M	G	N	F		N	F		J		X	N	K	G	L	Y
H	E	L	L	O		T	H	I	S		I	S		A			I		H	E	

Looking at the next most frequent characters, we have **R L** and **D**, so we calculate them.

Unencrypted	Value	7*x+9	7*x+9 mod 26	Encrypted
R	17	128	24	**Y**
L	11	86	8	**I**
D	3	30	4	**E**

Only one of those letters are in the key giving us:

G	L	I	I	D		M	G	N	F		N	F		J		X	N	K	G	L	Y
H	E	L	L	O		T	H	I	S		I	S		A			I		H	E	R

As this point you have gotten all but two of the letters. By the current rules, this would count as a correct solution with two letters wrong and you could leave it and go on, or you could guess some more or continue down the list of the frequency table. For now, it looks like those last two letters might be a **C** and **P** respectively, so we can test that quickly.

Unencrypted	Value	7*x+9	7*x+9 mod 26	Encrypted
C	2	23	23	**X**
P	15	114	10	**K**

Which confirms our guess giving us a final solution of:

G	**L**	**I**	**I**	**D**		**M**	**G**	**N**	**F**		**N**	**F**		**J**		**X**	**N**	**K**	**G**	**L**	**Y**
H	**E**	**L**	**L**	**O**		**T**	**H**	**I**	**S**		**I**	**S**		**A**		**C**	**I**	**P**	**H**	**E**	**R**

11. Vigenère cipher

Div A

A Vigenère cipher uses a repeating key in order to apply a different Caesar cipher to each letter in the group. The typical mapping table looks like this.

	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

To encrypt any phrase, you need to first pick a code key.
Then you repeat the code phrase as many times as necessary to cover the entire plaintext that you wish to encode. Note that for any characters what aren't encoded (like spaces and punctuation marks) you pretend that they aren't there and just use the next code phrase character with the next character to encode.

Plaintext: **SCIENCE OLYMPIAD CODE BREAKERS**
Key: *CEASER*C **EASERCEA SERC EASERCEA**
Ciphertext: **UGIWRTG SLQQGKED USUG FRWEBGVS**

To encode, all you need to do is take the character from the plaintext and the corresponding character from the key and look them up in the column and row of the mapping table. In this example for the first character, you have a Plaintext of **S** and a Key of **C**. Look in the **S** row and the **C** column to find the letter **U**. Note you can use the **S** column and the **C** row and you will get the same result. You repeat this process for each of the letters in the Plaintext.

To decrypt, you need to do the reverse, BUT instead of using the letters as the row and column header, you use the corresponding key to find the row or column and then find the corresponding ciphertext character in that column (or row) and use the matching header as the decryption key. So, in this case with a Ciphertext of **U** and a key of **C** you go to the column labeled **C** and look down until you find the letter **U** and then find the corresponding row header to see that it is the letter **S**.

12. Porta cipher

Div B Div C

A Porta cipher works very much like the Vigenère cipher uses a repeating key in order to apply a different mapping to each letter in the group. The biggest difference is that it uses a different mapping and that there are only 13 different possibilities. Note that there are other options for the Porta table, but we are using the ACA convention for the table (https://www.cryptogram.org/downloads/aca.info/ciphers/Porta.pdf).

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

One significant attribute of the Porta cipher is that letters in the A-M range will map to a letter in the N-Z range and vice-versa. In many ways, this makes the cipher easier to break with only a few clues.

Plaintext: **SCIENCE OLYMPIAD CODE BREAKERS**
Key: *PORTA*PO RTAPORTA PORT APORTAPO
Ciphertext: **LWQNAWY GULITHRN XWGZ RVKZWXYKK**

Another interesting attribute of the Porta cipher is that it is 100% reversable. Encrypting the Ciphertext with the same key results in the Plaintext.

To decrypt, you take the letter from the key and use it to determine the row in the porta table. Then you look at the corresponding letter to encode/decode. If the letter is in the A-M range, you use the row at the top to determine the column and pull the corresponding letter out of the selected row. If the letter is in the N-Z range, you find the column in the selected row and then look at the top to find the corresponding character. For example, with the first two letters, we take the P as the key and L as the Cipher text character. We look at the next to the last column which starts with the L and then go down to the O,P row to see the letter S. For the second letter, we have O as the key and W as the Cipher text character. We look in the same O,P row and scan over until we see the letter W in the third data column. We look to the header cell at the top and see that the letter C as the decoded character.

12.a. Cryptanalysis of a Porta Cipher

The following quote has been encoded with the Porta Cipher using a very common four letter word for the key. The 30th through 33rd cipher characters (**YVIH**) decode to be **EANS**

HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M

We start by first filling in what has been given to us as the clue.

HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M
E ANS

Using the porta table, we need to determine what the key characters are. We start with the cipher text **Y** which decodes to be **E**. Since **E** is in the A-M range, we look for the E column in the table and the scan down until we find the **Y**. From there we look at the row header and find that it corresponds to **O,P** so we will put an **O** above the **Y** as the keyword since both **O** and **P** decode to the same thing.

O QKW
HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M
E ANS

We repeat this process. Next, looking in the **A** column for a **V** we find that it corresponds to **Q,R**. When we get to the Cipher text **I** decoding to **N** we have to change our strategy of lookup because **N** is in the N-Z range. Instead of looking in the **N** column (which doesn't exist in the table) for an **I**, we look in the **I** column for an **N** and find it in the **K,L** row. This is an important attribute of the Porta Cipher being reversable. You will always notice that any character that is in the A-M range will map to something in the N-Z range. This means when looking up the letters, you need to pay attention to which is in a and use it for the column and then find the other value (which will be N-Z) to determine the correct row. As such it is often easier to think about the cipher character and the plaintext character as a pair and always order it alphabetically. This way it becomes faster to look them up. We can look up the last **H/S** pair to come up with the **W,X** row by finding the **S** in the **H** column. This gives us the following:

O QKW
HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M
E ANS

We now have a couple of options. Since we are told that the key is a four letter word, we could try to figure out what it is by counting to find out where the word split would start. Since it starts at the 30th character, we do a quick 30 mod 4 of it to find out that the O would be the second character. This means that the W would be the first character because the keyword repeats. Putting them in order and including the alternates in the pairs (**W,X O,P Q,R K,L**) it doesn't take a rocket scientist to see that the four letter word must be **WORK**.

WOQK
XPRL

With that in mind we can just start from the beginning and fill in the keyword

WORKW ORKWO RKWOR KWORK WORKW ORKWO RKWOR K
HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M
E ANS

This technique works really well when you aren't given enough letters for the keyword, for example, if it was a 5 letter keyword and we were only given four clue letters. But if you are told how many letters were in the keyword, you can simply ignore figuring out the keyword and just start filling in forwards and backwards.

So if we didn't try to figure out the keyword, but we have four of the four letters, we just put the **W** in front of the **O** over the **S**, the **K** before that, **Q** and so on repeating until we get to the start.

```
WOQKW OQKWO QKWOQ KWOQK WOQKW OQKWO QKW
HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M
                                  E ANS
```

Then you also fill in from the **W** on to the end

```
WOQKW OQKWO QKWOQ KWOQK WOQKW OQKWO QKWOQ K
HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M
                                  E ANS
```

Next comes decoding the ciphers. With the keyword, you can go much faster as you have the row to work from. The easiest way to do this is to attack all of the cipher characters which use the same encoding letter. We can start with the **W** and look at the row of the table to make it easy for us.

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

With this in mind, we find all of the ones under a **W** and map them. The H column header has an **S** in the **W,X** row. **I** gets us a **T**. Another **H** maps again to **S**. When we get to the **U**, we have to do the reverse and find the column it is in giving us a **J**. When we see the **S**, we remember that **H** mapped to **S** earlier, so we do the reverse. Another **I** gives us the **T** again. For the **P** we have to find the column header for the **P** in the **W,X** row which is **E**. Another **S** is an **H** and we end up with:

```
WOQKW OQKWO QKWOQ KWOQK WOQKW OQKWO QKWOQ K
HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M
S   T    S    J    H    T   E    HE ANS
```

Now that one letter is done, we proceed to the O row which is

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

We follow the same strategy. This time the **H** column maps to **O**. We find the **P** in the **I** column. **E** maps to **Y**. The **A** column gives us **U**, and the fortunate next **U** is the reverse giving us an **A**. Another **H** maps to **O**. **S** is in the **L** column, and the last **C** column gives us **W**

```
WOQKW OQKWO QKWOQ KWOQK WOQKW OQKWO QKWOQ K
HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M
SO  T I  SY   JU   HA   TO  E S  HE ANSW
```

Looking at what we have so far, a couple of words are obvious at the end so we fill them in.

```
WOQKW OQKWO QKWOQ KWOQK WOQKW OQKWO QKWOQ K
HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M
SO  T I  SY   JU   HA   TO  E S THE ANSWE R
```

We still have more to solve, so next we take the **Q,R** row

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

Looking up **U** finds it in **M** column and the next one is exactly the same – you should be able to see the benefit of doing all one row at a time now. The **G** column gives us **O**, followed by the **K** column that gives us **S**. Continuing on, the **A** column gives us **V**, and since we know **G** gave us **O**, we just enter **O** for **G** to find another **K** that we already mapped to **S**. This gives us a mostly complete one at:

```
WOQKW OQKWO QKWOQ KWOQK WOQKW OQKWO QKWOQ K
HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M
SOM T IM SY O JUS  HAV  TOG E S THE ANSWE R
```

At this point, some of the letters are obvious. It must start out as **SOMETIMES** and the **JUS? HAV?** Must be **JUST HAVE**. That only leaves the **C** mapped by the **K,L** row which we can either leave blank (don't forget the up to two wrong rule) or look it up and see that it is a **U**.

```
WOQKW OQKWO QKWOQ KWOQK WOQKW OQKWO QKWOQ K
HHUWI PUWHE GCUAK BSUAW IHOCP LKBSY VIHCZ M
SOMET IMESY O JUS THAVE TOGUE SSTHE ANSWE R
```

Now that everything is done, you can see that the cipher decoded to be:

SOMETIMES YOU JUST HAVE TO GUESS THE ANSWER

13. Baconian

Div B Div C

13.a. General Baconian strategies

There are two forms of a Baconian: 24 and 26 character. Science Olympiad uses the 24-character form, and the corresponding Baconian table will be provided as a resource for the test and looks like this:

AAAAA	A	**AABBA**	G	**ABBAA**	N	**BAABA**	T
AAAAB	B	**AABBB**	H	**ABBAB**	O	**BAABB**	U/V
AAABA	C	**ABAAA**	I/J	**ABBBA**	P	**BABAA**	W
AAABB	D	**ABAAB**	K	**ABBBB**	Q	**BABAB**	X
AABAA	E	**ABABA**	L	**BAAAA**	R	**BABBA**	Y
AABAB	F	**ABABB**	M	**BAAAB**	S	**BABBB**	Z

1. The first step is to identify the type of Baconian.
   - A letter for letter will generally have a random set of patterns such as:
     ^^^^_^^^^^^^^_^^__^_^__^^^_^^^^^^^^^__^^
   - A Pattern Baconian will have a general repeating set of symbols like:
     HellŏHelloHelḷoHĕḷlŏHĕḷloHĕlloHelloHĕḷlo
   - A Word Baconian will have a semi-readable sentence like:
     I SAID GO SAY A SONG UNTIL ONLY A SNAKE WAKES UP TOO
2. Break the pattern into groups of 5. One thing to be aware of is that a Baconian cipher may have nulls. An example of this is:
   WHEŅ IƝ ŢHÉ CÓÚRŞÉ ÓF HÚMAŃ ĘVÉŃŢŚ IŢ BÉCÓMÉŚ
   In this case, only the characters with accents are part of the Baconian alphabet.
3. For the first two types, you need to identify what corresponds to an **A** or a **B**. A key factor to remember for all Baconian ciphers is that if you look at the table, no Baconian letter starts with **BB**. As such in any group of 5 if the first two letters are the same, you know that they must stand for an **A**.
4. Another way to identify which is **A** and which is **B** is to count the number of each type. The **A**'s will greatly outnumber the **B**'s
5. Also, a sequence of 5 characters all the same must be composed of all **A** characters and correspond to the plain text letter A.

13.b. Letter for Letter Baconian

Counting is your best friend here. If you can identify two different types of symbols, then you will be able to pick what is A and what is B. Sometimes it is a bit complicated. For example, the 2018 NC State test had a Baconian like:

↑⤓↖↘⇲↓↘⤒⤓↓↘⤓↑↘⇲↓↘⤓↓↘↓⤓↘↸⇲↓↘⤒↑↖↘⤓↑↘⇲⤒↘↖↓↸↓↑↘↘⇲↖↘↘⤓↖

This contains several obvious options to consider here:

1. Up vs down: ↑⤒↖↸ ↓⤓↘⇲
2. Straight vs Angled: ↑⤒↓⤓ ↖↸↘⇲
3. Line vs No Line: ↑↓↖↘ ⤒⤓↸⇲

The first step is to break it into groups of 5

↑⤓↖↘⇲ ↓↘⤒⤓↓ ↘⤓↑↘⇲ ↓↘⤓↓↘ ↓⤓↘↸⇲ ↓↘⤒↑↖ ↘⤓↑↘⇲ ⤒↘↖↓↸ ↓↑↘↘⇲ ↖↘↘⤓↖

Next for the groups of 5 take a small set and identify the type of character. Since we have three possibilities, we should write the options down to distinguish them:

ududd ddudd ddudd ddddd dddud dduuu ddudd ududu duddd udddu
ssaaa sasss assaa sassa ssaaa sassa assaa saasa ssaaa aaasa
nlnnl nnlln nlnnl nnlnn nlnll nnlnn nlnnl lnnnl nnnnl nnnln

Looking at the second set, we see groups that start bout as **ss** and **aa** which means we can immediately reject that option without any further looking.

For the second set we see that the second word starts with **dd** which would mean **d=A**. A quick lookup of the first few letters:

ududd ddudd ddudd ddddd dddud
babaa aabaa aabaa aaaaa aaaba
W E E A C

Comes out as **WEEAC** which seems productive, so we quickly try the last choice. Since we see a group that starts out as **nn**, we must conclude that **n=A** and quickly try out the first few letters to discover that they come out as gibberish with **KGK**.

nlnnl nnlln nlnnl
abaab aabba abaab
K G K

13.c. Pattern Baconian

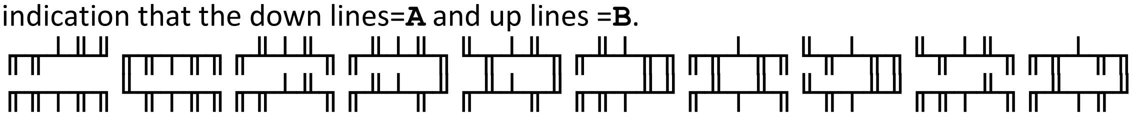
Pattern Baconian ciphers are attacked in the same manner as for the Letter for Letter Baconian. For example, if we had:

ĬTS CŎĹD OŬTSIDE IŤS ČŎLĎ OŬTSIDE ITŠ COLD OŬTŜĬDE ITS CŎĹD OUTSĬĎE ĬŤŠ CŎLD OUTSIDE ITS ČOĹD OŬTSĬDE ITS CŎLD OŬTSIDE ĬTS ČOLD OŬŤŜIDE ĬŤŠ COĹD OŬTSIDĚ ITS Č

It is quite apparent that the accented vs non-accented characters indicate the difference. A quick counting shows that only 10 out of the first 31 characters are accented which gives up accent=**B**. Applying this logic and breaking up into groups of 5 we get which starts out to decode as **STAY WARM**.

BAAAB BAABA AAAAA BABBA BABAA AAAAA BAAAA ABABB
S T A Y W A R M

Another style may be symbols such as were encoded in a tweet. Fortunately, in this case they are grouped into sets of 5. A quick look at the group shows the second group where all the lines are pointing down. Furthermore, counting them shows that 20 out of the first 35 characters are point down which is a strong indication that the down lines=**A** and up lines =**B**.



Applying that logic, the first 5 groups come out as:

aabbb aaaaa abbba abbba babba
H A P P Y

13.d. Word Baconian

The strategy for attacking a Word Baconian is slightly different. There will be multiple letters which map to **A** and **B**. For example, given the sample below with a hint that it starts out as **EVER**:

> **Maria built movie house badly.**
> **Super quick clock wrong.**
> **Board loose since chase begun**
> **Music buyer being movie extra.**
> **Heavy urban tower built worse since Maria began visit.**

The first step is to map the letters that we know.

```
Maria built movie house
AABAA BAABB AABAA BAAAA
  E     V     E     R
```

With that, we build a table showing what the letters all map to

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	B			A			B	A			B	A		A			B	A	B	A					

Looking at the table, we see that it starts out as **AB** and under **RSTU** we have a run of **BABA**. The most logical pattern in this case would just be alternating **A** and **B** mappings. A quick check of the next word **badly** maps it as **BA?B?** and if our guess is right, it is **BABBA** which maps to **Y** making our phrase start out as **EVERY**.
Filling in the rest of the table gives us:

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	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B

which we can use to decode the reminder of the phrase.
Note that it is very unlikely that the pattern will be as simple as **ABAB**… but it is reasonable to expect a pattern. Some additional techniques that you can use:

1. Even if there weren't a pattern (or you can't figure out the pattern), you can fill in the table by looking for groups of 5 that you know 4 of the mappings and identifying the possible letter choices which make sense in the decrypted text.
2. If you know one of the first two letters in a group are a **B**, (i.e., it starts out as **B?** or **?B**) then you can guarantee that the other letter maps to an **A**.

14. Morbit

The Morbit cipher uses Morse Code to encode the text.

There will be a Morse code table in forward and reverse on the resources page:

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

0	-----	2	●●---	4	●●●●-	6	-●●●●	8	---●●
1	●----	3	●●●--	5	●●●●●	7	--●●●	9	----●

●	E
●●	I
●●●	S
●●●●	H
-●●●	B
-----	0
●●●●●	5

-	T
●-	A
●●-	U
●●●-	V
-●●-	X
●----	1
-●●●●	6

-●	N
●-●	R
●●-●	F
-●-●	C
●●---	2
--●●●	7

--	M
●--	W
●-●●	L
-●--	Y
●●●--	3
---●●	8

-●●	D
●--●	P
--●●	Z
●●●●-	4
----●	9

-●-	K
●---	J
--●-	Q

--●	G

---	O

14.a. A Morbit problem to solve

For example, given the following cipher text to decode of and being told that it starts out as **CODE**:

99232572585158186858

The first thing to do is to map out what **CODE** would be in Morse code. Note that we use X to represent spaces.

```
-.-.X---X-..X.X
```

Next, we split it up into groups of 2 and map it to the cipher text

```
-. -. X- -- X- .. X. X-
9  9  2  3  2  5  7  2
```

We then build a table of mapping for what we know:

```
 1   2   3   4   5   6   7   8   9
 ??  X-  --  ??  ..  ??  X.  ??  -.
```

Based on the crib, we know the mapping of 5 of the 9 characters and are left looking for **.-**, **.X**, **-X** and **XX.**

```
9  9  2  3  2  5  7  2  5  8  5  1  5  8  1  8  6  8  5  8
-. -. X- -- X- .. X. X- .. ?? .. ?? .. ?? ?? ?? ?? ?? .. ??
C        O        D        E
```

Looking at the next letter in the sequence, we know that it starts out as -.. and that 8 must have an X in it (otherwise you would have at least 7 characters in a row without an X. Since we only have -X and XX left we can try them both.

First with 8=**-X** we get

```
9  9  2  3  2  5  7  2  5  8  5  1  5  8  1  8  6  8  5  8
-. -. X- -- X- .. X. X- .. -X .. ?? .. -X ?? -X ?? -X .. -X
C     O     D     E  X        ?     ?     ?     ?     U
```

Which doesn't seem likely, particularly with the U at the end

With 8=**.X** we get

```
9  9  2  3  2  5  7  2  5  8  5  1  5  8  1  8  6  8  5  8
-. -. X- -- X- .. X. X- .. .X .. ?? .. .X ?? .X ?? .X .. .X
C     O     D     E  B        ?     ?     ?     ?     S
```

Which looks promising and tells us that 1 must be either the remaining **-X** or **XX**, so we try **-X**

```
9  9  2  3  2  5  7  2  5  8  5  1  5  8  1  8  6  8  5  8
-. -. X- -- X- .. X. X- .. .X .. -X .. .X -X .X ?? .X .. .X
C     O     D     E  B        U     S     T  E  ?     S
```

Based on this, the only logical choice for 6 is to be **.-** which gives us

```
9  9  2  3  2  5  7  2  5  8  5  1  5  8  1  8  6  8  5  8
-. -. X- -- X- .. X. X- .. .X .. -X .. .X -X .X .- .X .. .X
C     O     D     E  B        U     S     T  E  R     S
```

Which means our table ends up as below (4 wasn't used and **XX** was left over so we get to put that in).

```
 1  2  3  4  5  6  7  8  9
-X X- -- XX .. .- X. .X -.
```

15. Pollux

A Pollux cipher is like a Morbit cipher, using the same Morse Table which should be on the resources page:

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

0	–––––	2	●●–––	4	●●●●–	6	–●●●●	8	–––●●
1	●––––	3	●●●––	5	●●●●●	7	––●●●	9	––––●

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

15.a. A Pollux problem to solve

Someone has encoded a phrase using the Pollux cipher and told you that 2,3 are Dots, 5,6 are Dashes and 8,9 are spaces (x). What does it say?

12059811012278473374491805946698143393935026296198313
0455866718756946591628223037761517666963203

15.b. Background on Solving Pollux

The Pollux cipher works by first converting the text into Morse code which is written as a series of dots (●), dashes (–), and spaces. To make it more convenient to solve, we typically represent the spaces as an ×. A single space is used at the end of a Morse code letter and a pair of spaces is used at the end of a word.

The person encoding the text then decides with digits will stand for dots/dashes/spaces with no restriction on that choice. For example, all the spaces could be represented by a 2, all of the dots by a 1 and all the other digits stand for a dash. Given the mapping of the digits, the Morse code is translated to the cipher text by picking a digit for the dash/dot/space. Since more than one digit can stand for a dash/dot/space, the encoding can choose whatever digit they would like.

Decoding a Pollux applies the process in reverse. It starts by mapping the known digits to their corresponding dot/dash/space and looking for complete Morse code characters. A complete Morse code character is one where an uninterrupted series of dots/dashes are delimited by a space. For example: ●●●× at the beginning represents the very familiar letter S (three dots). Finding ×●●× in the middle would represent the letter I (two dots). However, if we had ×● × (with an unmapped digit after the dot), we wouldn't know what the plain text is until we figured out the mapping for the digit.

With that in mind, the strategy for solving a Pollux consists of a set of steps:

1) Build a table of the possibilities for the digits.

0	1	2	3	4	5	6	7	8	9

2) Fill in the table with the known mappings and then just put ●–× for everything else since we don't know what they map to.
3) Underneath the digits of the cipher, fill in the known mappings with the corresponding Morse code character (●, –, ×).
4) Solve. As digits are eliminated, removed them from the possibility table and fill in known mappings under the cipher text. One special case that makes it easier to solve. If you eliminate × as a possibility, leaving ● or –, filling in the corresponding cipher spot with ? makes it easier to find places where a × belongs.

Some good solving rules that help quickly solve a Pollux

1) The first character will never be an ×. If the cipher digit at the start could map to an ×, you can eliminate that choice.
2) There will never be three spaces (×××) in a row. Hence if you find a cipher digit that is tripled, you know that it can't map to a ×.
3) Also looking for three spaces, if you have digits that already map to × and either have a doubled digit next to it which is unknown or ×× next to an unknown, you can eliminate × from that unknown.
4) No Morse letter is more than 4 dots/dashes and all numbers are exactly 5 dots/dashes. If there is a sequence of 6 characters with an unknown and all the remainder are known to be a dot/dash (●–?) then you know that the unknown must be a ×.
5) Not all sequences of 4 dots/dashes are legal Morse characters. (●●––, ●–●–, –––●, and ––––). If you have a pattern that would map to it, you know that you can eliminate it.

15.c. How to solve

Since we are told the mapping of 235689 ciphertext, we can build the following table:

0	1	2	3	4	5	6	7	8	9
●–×	●–×	●	●	●–×	–	–	●–×	×	×

Based on that Information we can map the cipher text as:

```
12059811012278473374491805946698143393935026296198313
 ● –××    ●● ×  ●●   × × –× ––××  ●●×●×●– ●–●×– ××● ●
   /                    /     E          /
04558667187569465916282230377615176669632O3
  ––×––  × ––× ––× –●×●●● ●  – –  –––×–●● ●
```

The first Morse code character can never be an ×,

0	1	2	3	4	5	6	7	8	9
●–×	●–	●	●	●–×	–	–	●–×	×	×

Based on that Information we can map the cipher text as:

```
12059811012278473374491805946698143393935026296198313
?● –××?? ?●● ×  ●●   ×?× –× ––××? ●●×●×●– ●–●×–?××●?●
```

```
    /                      /     E         /
0455866718756946591628223037761517666963203
  --×--  ?× --× --×?-●×●●● ● -?-? ---×-●● ●
```

At this point in time, 4 ciphertext characters still need to be mapped. Looking at the ciphertext, we see the sequence 449 which would result in three ×s in a row if 4 were an ×.

0	1	2	3	4	5	6	7	8	9
●–×	●–	●	●	●–	–	–	●–×	×	×

Based on that Information we can map the cipher text as:

```
12059811012278473374491805946698143393935026296198313
?● -××?? ?●● ×? ●● ??×?× -×?--××??●●×●×●- ●-●×-?××●?●
    /                      /     E         /
0455866718756946591628223037761517666963203
 ?--×-- ?× --×?--×?-●×●●● ●  -?-? ---×-●● ●
```

At this point in time, 4 ciphertext characters still need to be mapped. Based on the sequence 350262 with 0 possibly being one of ●–×, only × results in a legal Morse code character, so we can mark 0 as being ×.

0	1	2	3	4	5	6	7	8	9
×	●–	●	●	●–	–	–	●–×	×	×

Based on that Information we can map the cipher text as:

```
12059811012278473374491805946698143393935026296198313
?●×-××??×?●● ×? ●● ??×?××-×?--××??●●×●×●-×●-●×-?××●?●
   T/                /  T   /     E A  R    /
0455866718756946591628223037761517666963203
×?--×-- ?× --×?--×?-●×●●●×●  -?-? ---×-●●×●
                    S            D   E
```

At this point in time, 3 ciphertext characters still need to be mapped. Since 1 can still map to ●– we simply try them and look at the first word or two to see if it makes sense. Trying ● for 1 gives us a chunk: EARN S. Trying – for 1 gives us a chunk: EARM R. Which means we know that 1 must map to ●

0	1	2	3	4	5	6	7	8	9
×	●	●	●	●–	–	–	●–×	×	×

Based on that Information we can map the cipher text as:

```
12059811012278473374491805946698143393935026296198313
●●×-××●●×●●● ×? ●● ??×●××-×?--××●?●●×●×●-×●-●×-●××●●●
I  T/ I             E/ T   /     E A  R  N / S
0455866718756946591628223037761517666963203
×?--×-- ●× --×?--×●-●×●●●×●  -●-● ---×-●●×●
                R   S             D   E
```

At this point in time, 2 ciphertext characters still need to be mapped. Based on the sequence 37761517666 with 7 possibly being one of ●–×, only × results in a legal Morse code character, so we can mark 7 as being ×.

0	1	2	3	4	5	6	7	8	9
×	●	●	●	●–	–	–	×	×	×

Based on that Information we can map the cipher text as:

12059811012278473374491805946698143393935026296198313

●●×−××●●×●●●××?×●●×??×●××−×?−−××●?●●×●×●−×●−●×−●××●●●

I T/ I S / I E/ T / E A R N / S

0455866718756946591628223037761517666963203

×?−−×−−×●××−−×?−−×●−●×●●●×●××−●−●×−−−×−●●×●

M E/ M R S E/ C O D E

At this point in time, 1 ciphertext characters still need to be mapped. Since 4 can still map to ●– we simply try them and look at the first word or two to see if it makes sense. Trying ● for 4 gives us a chunk: IT IS EIIE TW HEARN SWME MWRSE COD. Trying – for 4 gives us a chunk: IT IS TIME TO LEARN SOME MORSE CODE. Which means we know that 4 must map to –

0	1	2	3	4	5	6	7	8	9
×	●	●	●	–	–	–	×	×	×

Based on that Information we can map the cipher text as:

12059811012278473374491805946698143393935026296198313

●●×−××●●×●●●××−×●●×−−×●××−×−−−××●−●●×●×●−×●−●×−●××●●●

I T/ I S / T I M E/ T O / L E A R N / S

0455866718756946591628223037761517666963203

×−−−×−−×●××−−×−−−×●−●×●●●×●××−●−●×−−−×−●●×●

O M E/ M O R S E/ C O D E

Now that we have mapped all the ciphertext characters, the decoded Morse code is the answer:

IT IS TIME TO LEARN SOME MORSE CODE

16. Fractionated Morse

Div B Div C

A Fractionated Morse cipher is a combination between a Pollux/Morbit and the K1/K2 alphabet from an Aristocrat. It is important to understand how they are encoded in order to be able to quickly decode them.

The first step is to pick a keyword and construct the alphabet. For example, if the keyword were DULCIMERS, then the alphabet is constructed by removing any duplicate letters in the phrase (of which we have none) and then adding the remainder of the alphabet in order after it. We end up with:

DULCIMERSABFGHJKNOPQTVWXYZ

Placing them into the table to map the morse characters we get:

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

From this we can see that **D** will correspond to ●●● to and **E** will correspond to ●×●. Since the letter **Z** was not used in the keyword, it ends up mapping to ××–.

Given this, we can then encode a simple phrase such as CODEBUSTERS by first converting it to Morse code:

C O D E B U S T E R S
−●−●×−−−×−●●×●×−●●●×●●−×●●●×−×●×●−●×●●●

Next we take the morse code and break it into groups of 3 padding with × as necessary, but in this case we got lucky and didn't need any padding. We can then look up the groups of 3 in the table above to generate the cipher text. We already knew that ●●● is **D** to and ●×● is **E** with the others pretty quick to look up.

C O D E B U S T E R S
−●− ●×− −−× −●● ×●× −●● ●×● ●−× ●●● ×−× ●×● −●× ●●●
B R J A T A E M D X E F D

As you should be able to see, the key to solving a Fractionated Morse cipher is to figure out the keyword and recognizing the patterns in the remainder of the alphabet after the keyword. For example if you lean that **N** is −×− and **Q** is ×●− then you immediately know that since there are two slots between them and likewise two letters, you know the mapping of **O** and **P**.

16.a. Solving a Fractionated Morse cipher

Someone has encoded a phrase using the Fractionated Morse cipher and told you that it ends with **EARS**. What does it say?

KMUPKSGHPDWWKDMUVDHVIZSRKPGBILTVORTNLJMXXEWOMIRDBQIVGCKQ QIS

●	●	●	●	●	●	●	●	●	●	–	–	–	–	–	–	–	–	×	×	×	×	×	×	×	×
●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×
●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–

The first step is to covert the **EARS** phrase to morse code and break it into groups of three.

●×● –×● –●× ●●●

Since the cipher text ends in **KQQIS** we immediately see that since all four groups of three are different, there is no way it will map which means that there must be one or two × at the end. Generating the two versions adding the × at the beginning and the end gives us

?×● ×●– ×●– ●×● ●●×
×●× ●–× ●–● ×●● ●××

We can see that the one with a single × at the end has a repeated pattern of ×●– which gives us the mapping:

?×● ×●– ×●– ●×● ●●×
K Q Q I S

allows us to fill in the table:

		S				I													Q						
●	●	●	●	●	●	●	●	●	●	–	–	–	–	–	–	–	–	×	×	×	×	×	×	×	×
●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×
●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–

K M U P K S G H P D W W K D M
?×●??????????×●●●×????????????????????●??????

U V D H V I Z S R K P G B I L
???????????????●×●???●●×????×●?????????●×●???

T V O R T N L J M X X E W O M
???

I R D B Q I V G C K Q Q I S
●×●?????????×●–●×●???????????×●×●–×●–●×●●●×
E A R S

The first thing that we can do is determine where **K** is since we know two of the morse characters. As ?×● it can be one of ●×●, –×●, or ××●. However since we already know that I maps to ●×●, we are left with two choices. Given the position of Q near the end of the list and S already being used, we are left with 8 characters (**RTUVWXYZ**) to take up the 6 slots after it including ××● which means that where **K** must be –×● so we can fill it in and update the cipher:

		S				I									K			Q							
●	●	●	●	●	●	●	●	●	●	−	−	−	−	−	−	−	−	×	×	×	×	×	×	×	×
●	●	●	−	−	−	×	×	×	●	●	●	−	−	−	×	×	×	●	●	●	−	−	−	×	×
●	−	×	●	−	×	●	−	×	●	−	×	●	−	×	●	−	×	●	−	×	●	−	×	●	−

K M U P K S G H P D W W K D M

−×●?????????−×●●●×??????????????????−×●??????

T S

U V D H V I Z S R K P G B I L

???????????????●×●???●●×???−×●?????????●×●???

T V O R T N L J M X X E W O M

???

I R D B Q I V G C K Q Q I S

●×●?????????×●−●×●?????????−×●×●−×●−●×●●●×

R E A R S

Next we have a couple of paths to take. We could assume that **Z** doesn't appear in the keyword and map it to **××−**. We can also notice that the cipher starts with the letter **T** and the next morse character is a ● which happens to be the first part of the letter **H** with **THE** being one of the most common words to start a phrase. This gives us

K M U P K S G H P D W W K D M

−×●●●●×●×???−×●●●×??????????????????−×●???●●●

T H E S

U V D H V I Z S R K P G B I L

×●×????????????●×●××−●●×???−×●?????????●×●???

E E/ D

T V O R T N L J M X X E W O M

????????????????????????●●●???????????????●●●

I R D B Q I V G C K Q Q I S

●×●?????????×●−●×●?????????−×●×●−×●−●×●●●×

R E A R S

And we can update the table as:

M		S				I									K			Q	U						
●	●	●	●	●	●	●	●	●	●	−	−	−	−	−	−	−	−	×	×	×	×	×	×	×	×
●	●	●	−	−	−	×	×	×	●	●	●	−	−	−	×	×	×	●	●	●	−	−	−	×	×
●	−	×	●	−	×	●	−	×	●	−	×	●	−	×	●	−	×	●	−	×	●	−	×	●	−

This turns out to be a really big break since we can see can assume that **QU** is not part of the keyword and the remaining 5 slots correspond to the last 5 letters in the alphabet after **U**. This gives us a table below that fills in a lot of the cipher:

M		S				I									K				Q	U	V	W	X	Y	Z
●	●	●	●	●	●	●	●	●	●	–	–	–	–	–	–	–	–	×	×	×	×	×	×	×	×
●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×
●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–

K M U P K S G H P D W W K D M
–×●●●●×●×???–×●●●×???????????×––×–––×●???●●●
T H   E   S   M O
U V D H V I Z S R K P G B I L
×●××–●??????×–●●×●××–●●×???–×●?????????●×●???
E/   E/ D
T V O R T N L J M X X E W O M
???×–●??????????????????●●●×–××–×???×––???●●●
T/ T
I R D B Q I V G C K Q Q I S
●×●?????????×●–●×●×–●??????–×●×●–×●–●×●●●×
R   E   E A R   S

This also leads to another lucky discovery since we believe that **THE** is the first word, we can guess that **P** must start with an **×**, there is only one slot left and it happens to be right before the letter **Q** which strengthens our guess:

M		S				I									K			P	Q	U	V	W	X	Y	Z
●	●	●	●	●	●	●	●	●	●	–	–	–	–	–	–	–	–	×	×	×	×	×	×	×	×
●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×
●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–

K M U P K S G H P D W W K D M
–×●●●●×●××●●–×●●●×??????×●●???×––×–––×●???●●●
T H   E/ U   S   M O
U V D H V I Z S R K P G B I L
×●××–●??????×–●●×●××–●●×???–×●×●●???????●×●???
E/   D   E/ D   E
T V O R T N L J M X X E W O M
???×–●??????????????????●●●×–××–×???×––???●●●
T/ T
I R D B Q I V G C K Q Q I S
●×●?????????×●–●×●×–●??????–×●×●–×●–●×●●●×
R   E   E A R   S

Seeing the start of the phrase as **THE US?** certainly sounds like it starts out **THE USE**, so we can assume that G maps to ●××. Filling that in gives us:

M		S				I		G							K			P	Q	U	V	W	X	Y	Z
●	●	●	●	●	●	●	●	●	●	–	–	–	–	–	–	–	–	×	×	×	×	×	×	×	×
●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×
●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–

```
 K  M  U  P  K  S  G  H  P  D  W  W  K  D  M
–×●●●●×●××●●–×●●●×●××???×●●???×––×–––×●???●●●
T H    E/ U   S   E/           M  O
 U  V  D  H  V  I  Z  S  R  K  P  G  B  I  L
×●××–●??????×–●●×●××–●●×???–×●×●●●××???●×●???
 E/          D   E/ D          E S   /
 T  V  O  R  T  N  L  J  M  X  X  E  W  O  M
???×–●??????????????????●●●×–××–×???×––???●●●
                              T/ T          H
 I  R  D  B  Q  I  V  G  C  K  Q  Q  I  S
●×●?????????×●–●×●×–●●××???–×●×●–×●–●×●●●×
             R   E D   /       E A  R   S
```

Looking at the end, we have a word that ends in **EARS** and has either four morse symbols ending in – or is two letters. A quick look at the four symbol morse characters ending with – gives us either **V J** or **Y** with **YEARS** being a very good choice. Filling the mapping for C gives us:

M		S				I		G		C					K			P	Q	U	V	W	X	Y	Z
●	●	●	●	●	●	●	●	●	●	–	–	–	–	–	–	–	–	×	×	×	×	×	×	×	×
●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×
●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–

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

Also seeing the **H???RED YEARS** only leaves us with one word that fits there – **HUNDRED YEARS** – so we get the mapping of a few more letters:

M		S			R	I		G	B	C	D				K			P	Q	U	V	W	X	Y	Z
●	●	●	●	●	●	●	●	●	●	–	–	–	–	–	–	–	–	×	×	×	×	×	×	×	×
●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×
●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–

K M U P K S G H P D W W K D M

–×●●●●×●××●●–×●●●×●××???×●●???×––×–––×●–●×●●●

T H E/ U S E/ M O R S

U V D H V I Z S R K P G B I L

×●××–●–●×???×–●●×●××–●●×●–×–×●×●●●××–●●●×●???

E/ C D E/ D A T E S / B

T V O R T N L J M X X E W O M

???×–●?××●–×????????????●●●×–××–×???×––?××●●●

T/ T / H

I R D B Q I V G C K Q Q I S

●×●●–×–●×–●●×●–●×●×–●●××–●––×●×●–×●–●×●●●×

U N D R E D / Y E A R S

This tells us that **O** must map to **?××** and since **G** already maps to **●××** it only leaves **–××** for **O** which is conveniently right next to **P**. (Remember that **×××** isn't mapped to anything). As we fill that in, we see **T?O HUNDRED YEARS** at the end which could only be **TWO HUNDRED YEARS** giving us the mapping for **E**. Likewise **MORSE C?DE** must be **MORSE CODE**:

M		S		E	R	I		G	B	C	D		H		K		O	P	Q	U	V	W	X	Y	Z
●	●	●	●	●	●	●	●	●	●	–	–	–	–	–	–	–	–	×	×	×	×	×	×	×	×
●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×
●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–

K M U P K S G H P D W W K D M

–×●●●●×●××●●–×●●●×●××–––×●●–●××––×–––×●–●×●●●

T H E/ U S E/ O F / M O R S

U V D H V I Z S R K P G B I L

×●××–●–●×–––×–●●×●××–●●×●–×–×●×●●●××–●●●×●???

E/ C O D E/ D A T E S / B

T V O R T N L J M X X E W O M

???×–●–××●–×????????????●●●×–××–×●––×–––××●●●

K / A T/ T W O / H

I R D B Q I V G C K Q Q I S

●×●●–×–●×–●●×●–●×●×–●●××–●––×●×●–×●–●×●●●×

U N D R E D / Y E A R S

We can pretty much sight read the rest of the cipher and fill in the remaining letters:

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

●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×	×	●	●	●	–	–	–	×	×
●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–	×	●	–

K M U P K S G H P D W W K D M

–×●●●●×●××●●–×●●●×●××–––×●●–●××––×–––×●–●×●●●

T H E/ U S E/ O F /M O R S

U V D H V I Z S R K P G B I L

×●××–●–●×–––×–●●×●××–●●×●–×–×●×●●●××–●●●×●–×–

E/ C O D E/ D A T E S /B A C

T V O R T N L J M X X E W O M

●–●×–●–××●–×●–●●×––×–––×●●●×–××–×●––×–––××●●●

K /A L M O S T/ T W O /H

I R D B Q I V G C K Q Q I S

●×●●–×–●×–●●×●–●×●×–●●××–●––×●×●–×●–●×●●●×

U N D R E D /Y E A R S

This gives us the mapping of every letter except A and F and we can see that the keyword would have to be **MASTERING**.

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

17. Cryptarithm

Div B Div C

17.a. General Solving Rules

In general, the strategy for an Aristocrat is:
Fill in letters from any clues you are given
Look for single letter words which will generally be **A** or **I**
Check the frequency. The most common letters in English are **ETAOIN**.
Look for contractions (**DON'T**, **DOESN'T**)
Look for two and three letter words
Look for patterns "**IT IS**" and "**THAT**" are good ones
Look for double letters

A much more detailed guide can be found on Puzzle Baron's Cryptograms site at
https://cryptograms.puzzlebaron.com/tutorial.php

17.b. Solving a Cryptarithm

SOCIAL	social		
+ SOLAR	solar	24687	95310
VEHICLE	vehicle		

	0	1	2	3	4	5	6	7	8	9
S										
O										
L										
A										
R										
T										
M										
P										
N										
E										

Immediately we know that **V** must be 1 because you can only carry a single digit from the previous column addition. Furthermore, since there is only one digit in the previous column, it must be a 9 in order to carry from the column before that which means that the first two digits of the final result must be 10 telling us the mappings of **V** and **E**. We can mark that in the table.

SOCIAL	9ocial		S V
+ SOLAR	9olar	24687	95310
VEHICLE	10hicl0		

	0	1	2	3	4	5	6	7	8	9
S										✓
O										
C										
I										
A										
L										
R										
V		✓								
E	✓									
H										

Some quick observations we can learn from what we have filled in so far:

In the first column we have: **L+R=10** which because of the numbers already mapped can only be 2+8, 3+7 or 4+6 in either order.

In the second column we add the carry from the first column to **A+A** giving us L which must be odd. Based on what we learned in the first column, we know that **L** must be either 3 or 7 which means **A** must be one of 1, 3, 6 or 8. We can quickly try all 4 options

A=1 won't work since **V=**1 already

For **A=**3 we end up with **L=**7, but **L+R=**10 means that **R** would also be 3 so we can't use that.

For **A**=6 we have **L**=3 which forces **R**=7 which means it is a possibility.

For **A**=8 we get **L**=7 which forces **R**=3 leaving it as a possibility.

Either way we know that either **L** or **R** is 3 and the other is 7 and that **A** must be either 6 or 8, so we mark it in the table. We also know that since **A** > 5 there is a carry into the next column

SOCIAL	9ocial		S V
+ SOLAR	+ 9olar	24687	95310
VEHICLE	10hicl0		

	0	1	2	3	4	5	6	7	8	9
S										✓
O										
C										
I										
A										
L										
R										
V		✓								
E	✓									
H										

With the next column, we know that (carry from previous column) 1+I+L=C. Since **L** must be either 3 or 7 and **I** and **C** are both limited to only five possible values, we look at the ten possible combinations to see which work.

L	I	1+I+L=C	Notes
3	2	6	
3	4	8	
~~3~~	~~5~~	~~9~~	S=9
~~3~~	~~6~~	~~(carry)0~~	E=0
3	8	(carry)2	
~~7~~	~~2~~	~~(carry)0~~	E=0
7	4	(carry)2	
~~7~~	~~5~~	~~(carry)3~~	C≠3
7	6	(carry)4	
7	8	(carry)6	

Immediately this eliminates C=5 and **I**=5 leaving only **H** or O to be 5.

With H and O in mind, we notice that (possible carry)+O+9=**H** (with a carry). This tells us that O>**H** and that either O−1=**H** or O−2=**H** depending on the carry from the previous column. Since one of them must be 5 we either have O=5 and **H**=4 or **H**=5 and O=6. This means that there can not be a carry from C+O and C+O<9.

With this information in hand, we fill in our table and eliminate quite a few options:

SOCIAL	9ocial		S V
+ SOLAR	+ 9olar	24687	95310
VEHICLE	10hicl0		

	0	1	2	3	4	5	6	7	8	9
S										✓
O										
C										
I										
A										
L										
R										
V		✓								
E	✓									
H										

The only column we haven't looked at is the (possible carry from previous column+)C+O=**I**. Taking into account what we learned with the **I+L** column and knowing that there are only two possible values for **I** and 4 possible values for C or **I** we can test them out quickly in a table.

C	O	Carry+C+O=I	Notes
2	5	8	
2	6	9	S=9
4	5	(carry)0	E=0
4	6	(carry)1	V=1
6	5	(carry)2	
6	5	(carry)3	I≠3
8	5	(carry)3	I≠3
8	6	(carry)4	

Since we previously determined that C+O<9 this tells us that the only possible answer is that C=2, O=5 and **I**=8. Since we know what when O=5, **H**=4 we can fill that in too.

This leaves **A**=6 as the only option. Previously we also determined that for **A**=6, **L**=3 and **R**=7 which gives us the final table and we can fill in the letters for the answer.

	SOCIAL		952863
+	SOLAR	+	95367
	VEHICLE		1048230

CHAIR	SOLVE
24687	95310

	0	1	2	3	4	5	6	7	8	9
S										✓
O						✓				
C			✓							
I									✓	
A							✓			
L				✓						
R								✓		
V		✓								
E	✓									
H					✓					

18. Tap Code Cipher

Div A

The Tap Code cipher is an easy cipher to remember and can be solved in two ways. One way is to write down the letters in a table and then use the sets of taps to look up the entry in the table. The other way is to remember five letters and solve it on the fly.

18.a. A Tap Code Cipher to decode

Your friend just gave you this message written in a Tap Code Cipher. What does it say?

●●● ● ● ●●●●● ●●●● ●●●● ●●●● ●●● ●● ●●● ● ● ●●●●● ● ● ●●●●●
●●●● ●●● ●●● ●●●● ●●● ●● ● ●●●●● ●● ● ●●●● ●●●●● ●●● ●●●

18.b. Solving a Tap Code Cipher it with a table

The first thing to do is create a table to map the letters. Just draw a grid of 5x5 boxes:

Then fill in the table with the letters A-Z remembering that C and K go into the same spot. You can also put numbers across the rows and columns to aid in decoding.

	1	2	3	4	5
1	A	B	CK	D	E
2	F	G	H	I	J
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

With the table in hand, the next step is to count the number of taps and group them in sets of 2

●●● ● ● ●●●●● ●●●● ●●●● ●●●● ●●● ●● ●●● ● ● ●●●●● ● ● ●●●●●
3 1/1 5 /4 4 /4 3 /2 3 /1 1/5 1/1 5
●●●● ●●● ●●● ●●●● ●●● ●● ● ●●●●● ●● ● ●●●● ●●●●● ●●● ●●●
4 3 /3 4 /3 2 /1 5 /2 1/4 5 /3 3

It is important that the count ends with a pair of two numbers. If there is only one, then carefully go back and find where a set got skipped.

With the numbers in hand, it is a matter of using the first in the pair to look up the row and the second to pick the column and then put the letter in place:

```
●●● ● ● ●●●●● ●●●● ●●●● ●●●● ●●● ●● ●●● ● ● ●●●●● ● ● ●●●●●
3   1/1 5     /4    4   /4    3  /2  3  /1 1/5     1/1 5
L     E       T         S        H      A   V        E

●●●● ●●● ●●● ●●●● ●●● ●● ● ●●●●● ●● ● ●●●● ●●●●● ●●● ●●●
4     3  /3   4   /3   2 /1 5    /2  1/4    5    /3   3
S        O        M      E       F    U          N
```

This gives us the answer, The only thing that you may have to do is choose whether something was a **C** or a **K** based on the word.

18.c. Solving a Tap Code Cipher on the fly by remembering 5 letters

Another way to solve pretty quickly without the table is to remember the five letters in the first column: **AFLQV** Just start with the first set and put the letters under each tap until you get the end. Then for the next set, you start with the letter you ended up with and advance it for each one.

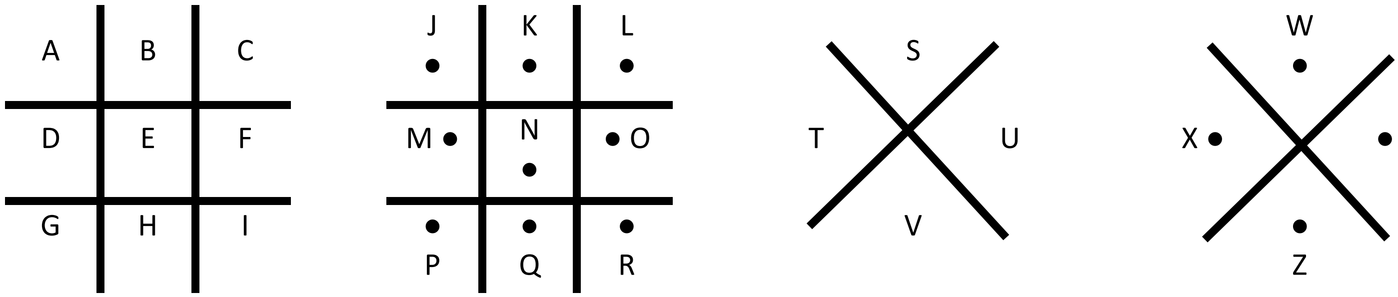
```
●●● ● ● ●●●●● ●●●● ●●●● ●●●● ●●● ●● ●●● ● ● ●●●●● ● ● ●●●●●
AFL L A ABCDE AFLQ QRST AFLQ QRS AF FGH A A AFLQV V A ABCDE
    L       E          T          S      H   A       V      E

●●●● ●●● ●●● ●●●● ●●● ●● ● ●●●●● ●● ● ●●●● ●●●●● ●●● ●●●
AFLQ QRS AFL LMNO AFL LM A ABCDE AF F AFLQ QRSTU AFL LMN
       S        O      M       E    F          U       N
```

The last letter in each set is the one to use. Once again, you may have to change a **C** to be a **K** based on the word.

19. PigPen/Masonic Cipher

Div A

The PigPen cipher is an easy cipher requires remembering a simple setup in order to create the key. With that key, decoding is just looking up the symbols in the key. All the students need to do is create two tic-tac-toe grids and two X grids and then fill them in with the alphabet putting dots on the second of the grids as so:



19.a. Solving A PigPen Cipher

Given this simple cipher to solve:

The symbols can be decoded by looking at the letters in the corresponding spot in the grids. The first symbol ⊐ corresponds to the left center of the first grid, hence the letter **D**. The next symbol ☐ Corresponds to the center of the first grid giving us the letter **E**. The next ∟ is the upper right for the letter **C**. With the fourth letter ⊏ we have a dot in it, so the letter comes from the second grid in the right middle for the letter **O**. The next two are repeats of the first to leaving us with **DECODE** so far.

DECODE

The second word starts with > which corresponds to the left half of the cross for the letter **T**. This is followed by ⊓┌ which is **HI**. Lastly we have V which corresponds to the letter **S** giving us **THIS** for the second word.

DECODE THIS

The same pattern repeats for the last letters resulting in

DECODE THIS CIPHER

20. Nihilist Cipher

Div B Div C

The Nihilist cipher uses two keywords to encode the plaintext. The first is used to construction a Polybius Square which is used to map all of the letters to numbers. The second keyword is used to encode the plaintext, applying each letter in the keyword in sequence in just the same way that the Vigenère cipher works.

Constructing the Polybius Square works in a manner like a K1/K2 alphabet with the exception that the letter **J** is not used. In general, if you had a keyword with the letter **J**, you would substitute the letter **I**. Once you have the Polybius keyword, you then fill out the remainder of the alphabet skipping any letters in the keyword. For example, if we take the keyword **FASHIONED** we can start the Polybius square as:

	1	2	3	4	5
1	F	A	S	H	I
2	O	N	E	D	
3					
4					
5					

We then fill the remaining squares row by row with the unused letters of the alphabet (skipping **J** of course): **BCGKLMPQRTUVWXYZ** giving us:

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

From this Polybius Square we now have the mappings for all the letters we would use in the cipher. **F** being in the first row of the first column maps to **11** while **E** in the third column of the second row maps to **23** and **Z** all at the end maps to **55**.

Now that we have the mapping, we need to pick an encryption keyword. This needs to be a different keyword from the Polybius keyword but is often related. For this example, we will pick **SENSE**. For convenience we can go ahead and map it to the values as **13 23 22 13 23**.

With our Square and encryption keyword in hand we can take on the task of encrypting the plaintext. We will demonstrate it with the plaintext of **EASY CIPHER EXAMPLE**. The Keyword is repeated for each letter in the plaintext starting over when all the letters in the keyword are used up. The Polybius Values are determined for each letter and the final ciphertext is simply the sum of the two numbers.

Plaintext	E	A	S	Y		C	I	P	H	E	R		E	X	A	M	P	L	E
Polybius Value	23	12	13	54		31	15	41	14	23	43		23	53	12	35	41	34	23
Keyword	S	E	N	S		E	S	E	N	S	E		S	E	N	S	E	S	E
Keyword Value	13	23	22	13		23	13	23	22	13	23		13	23	22	13	23	13	23
Ciphertext	36	35	35	67		54	28	64	36	36	66		36	76	34	48	64	47	46

The encrypted ciphertext is **36 35 35 67 54 28 64 36 36 66 36 76 34 48 64 47 46**. It is worth noting that both the **E/S** combinations and **H/N** combinations both encode to the same value **36**, much as you see with the Vigenère cipher.

20.a. Solving a Nihilist Cipher given the keys

Problem to solve:

Given a Polybius key of **SCIENCE OLYMPIAD** and an encoding key of **FUN** decode the following quote by Criss Jami that has been encoded using the Nihilist Cipher.

79 92 29 67 64 50 69 65 26 79 63 56 65 73 37 48 66 50

48 82 49 79 65 59 45 102 27 46 65 26 45 64 26 45 92 66

79 96 28 49 86 66 59 82 48 55 102 60 47 96

How to solve it:

The first step is to use the key to construct the Polybius square. We do this by eliminating all the duplicate letters and then adding all the remaining letters of the alphabet (remembering that I and J count as the same letter) in the same way that we would construct a K1/K2 alphabet. This gives us:

SCIENOLYMPADBFGHKQRTUVWXZ

We can use this to fill in the Polybius square row by row giving us:

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

This tells us the mapping values for all of the letters. For example, in row 3, column 4 we have the first letter of our encoding key **F** which gives us the value **34**. The second letter **U** is in Row 5, column 1 mapping to **51** and **N** is in Row 1 column 5 mapping to **15**. Putting them all together gives us an encoding key of **34 51 15**.

The next step is to write them under each of the cipher text values in the problem repeating when we hit the end of the key as you can see with the *italic* values below:

79 92 29 67 64 50 69 65 26 79 63 56 65 73 37 48 66 50
34 51 15 34 51 15 34 51 15 34 51 15 34 51 15 34 51 15

48 82 49 79 65 59 45 102 27 46 65 26 45 64 26 45 92 66
34 51 15 34 51 15 34 51 15 34 51 15 34 51 15 34 51 15

79	**96**	**28**	**49**	**86**	**66**	**59**	**82**	**48**	**55**	**102**	**60**	**47**	**96**
34	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*

Once we have the values in place we do a simple subtraction from the encoded value: For example the first one 79-34 = 45. All the values are highlighted yellow below:

79	**92**	**29**	**67**	**64**	**50**	**69**	**65**	**26**	**79**	**63**	**56**	**65**	**73**	**37**	**48**	**66**	**50**
34	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*
45	**41**	**14**	**33**	**13**	**35**	**35**	**14**	**11**	**45**	**12**	**41**	**31**	**22**	**22**	**14**	**15**	**35**

48	**82**	**49**	**79**	**65**	**59**	**45**	**102**	**27**	**46**	**65**	**26**	**45**	**64**	**26**	**45**	**92**	**66**
34	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*
14	**31**	**34**	**45**	**14**	**44**	**11**	**51**	**12**	**12**	**14**	**11**	**11**	**13**	**11**	**11**	**41**	**51**

79	**96**	**28**	**49**	**86**	**66**	**59**	**82**	**48**	**55**	**102**	**60**	**47**	**96**
34	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*
45	**45**	**13**	**15**	**35**	**51**	**25**	**31**	**33**	**21**	**51**	**45**	**13**	**45**

From this point it is just a matter of using the numbers in the Polybius square to select the row/column and get the letter. The first value **45** is Row 4, Column 5 which has the letter **T**. The next value **41** is Row 4, Column 1 which is the letter **H**. Repeat the process for all the remaining values to get:

79	**92**	**29**	**67**	**64**	**50**	**69**	**65**	**26**	**79**	**63**	**56**	**65**	**73**	**37**	**48**	**66**	**50**
34	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*
45	**41**	**14**	**33**	**13**	**35**	**35**	**14**	**11**	**45**	**12**	**41**	**31**	**22**	**22**	**14**	**15**	**35**
T	**H**	**E**	**B**	**I**	**G**	**G**	**E**	**S**	**T**	**C**	**H**	**A**	**L**	**L**	**E**	**N**	**G**

48	**82**	**49**	**79**	**65**	**59**	**45**	**102**	**27**	**46**	**65**	**26**	**45**	**64**	**26**	**45**	**92**	**66**
34	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*
14	**31**	**34**	**45**	**14**	**44**	**11**	**51**	**12**	**12**	**14**	**11**	**11**	**13**	**11**	**11**	**41**	**51**
E	**A**	**F**	**T**	**E**	**R**	**S**	**U**	**C**	**C**	**E**	**S**	**S**	**I**	**S**	**S**	**H**	**U**

79	**96**	**28**	**49**	**86**	**66**	**59**	**82**	**48**	**55**	**102**	**60**	**47**	**96**
34	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*	*15*	*34*	*51*
45	**45**	**13**	**15**	**35**	**51**	**25**	**31**	**33**	**21**	**51**	**45**	**13**	**45**
T	**T**	**I**	**N**	**G**	**U**	**P**	**A**	**F**	**T**	**E**	**R**	**I**	**T**

Reading the letters off and adding in spaces gives us the plain text: **`The biggest challenge after success is shutting up about it`**.

20.b. Solving a Nihilist Cipher via cryptanalysis

Problem to solve:

The following quote by Abhijit Naskar has been encoded using the Nihilist Substitution cipher. You have been told that the decoded text starts with **`SOMETIMES`**. What does it decrypt to?

97 82 57 45 98 74 57 45 97 86 84 78 64 52 88 66 64 86

86 85 98 73 97 75 104 55 94 78 98 86 94 77 88 63 64 67

88 65 88 75 98 73 54 75 75 93 76 75 104 85

Steps to solution

The process of cryptanalysis is straightforward with the information given.

1. Determine the keyword length.
2. Map the keyword values to determine possible positions in the Polybius square.
3. Subtract the known keyword values to determine the plaintext square positions.
4. Fill in the Polybius square with known information.
5. Iterate over the cipher and Polybius square filling in information until complete.

Determining the Keyword Length

Fortunately for the Nihilist, you are guaranteed that for each mapping letter, there can only be 5 unique values for the last digit, so we can do a count of the number of times that the last digit occurs. Here's a simple way to think about it. Since the last digit for the plaintext mapping comes from the column and there are only 5 columns, adding 5 unique digits to whatever digit is for the keyword at that position can only produce 5 different values. If we had a 3-letter keyword, the mapping looks like this:

K1	K2	K3	**K1**	K2	K3	**K1**	K2	K3	**K1**	K2	K3	**K1**	K2	K3	**K1**	K2	K3
97	82	57	45	98	74	57	45	97	86	84	78	64	52	88	66	64	86

K1	K2	K3	**K1**	K2	K3	**K1**	K2	K3	**K1**	K2	K3	**K1**	K2	K3	**K1**	K2	K3
86	85	98	73	97	75	104	55	94	78	98	86	94	77	88	63	64	67

K1	K2	K3	**K1**	K2	K3	**K1**	K2	K3	**K1**	K2	K3	**K1**	K2
88	65	88	75	98	73	54	75	75	93	76	75	104	85

We can build a little table to track the digits for the 3-letter keyword and just make a mark if we find the last digit in that position. It is worth observing that there is no possibility that the last digit will be a 1 since the smallest column number in the Polybius square is 1 and we all know that 1+1 = 2.

9**7** 82 57 4**5** 98 74 5**7** 45 97 8**6** 84 78 6**4** 52 88 6**6** 64 86

8**6** 85 98 7**3** 97 75 10**4** 55 94 7**8** 98 86 94 77 88 63 64 67

88 65 88 75 98 73 54 75 75 93 76 75 104 85

	2	3	4	5	6	7	8	9	0
K1		X	X	X	X	X	X		
K2									
K3									

As you can see, once we get to the 78 on in the middle of the second line, we have already found 6 different values which tells us the keyword can't be 3 letters long. So, let's try the same table with every 4th position to see if the keyword is 4 letters long.

9**7** 8**2** 5**7** 4**5** 9**8** 7**4** 5**7** 4**5** 9**7** 8**6** 8**4** 7**8** 6**4** 5**2** 8**8** 6**6** 6**4** 8**6**

8**6** 8**5** 9**8** 7**3** 9**7** 7**5** 10**4** 5**5** 9**4** 7**8** 9**8** 8**6** 9**4** 7**7** 8**8** 6**3** 6**4** 6**7**

8**8** 6**5** 8**8** 7**5** 9**8** 7**3** 5**4** 7**5** 7**5** 9**3** 7**6** 7**5** 10**4** 8**5**

	2	3	4	5	6	7	8	9	0
K1			X	X		X	X		
K2	X	X	X	X	X				
K3			X		X	X	X		
K4				X	X	X	X		

This looks really promising as we get all the way through all the letters without having more than 5 results. We could assume that the keyword is 4 letters long and go on to the next step, but it is worth doing a quick sanity check for 5 letters:

9**7** 82 57 45 98 7**4** 57 45 97 86 8**4** 78 64 52 88 6**6** 64 86

86 85 9**8** 73 97 75 104 5**5** 94 78 98 86 9**4** 77 88 63 64 6**7**

88 65 88 75 9**8** 73 54 75 75 9**3** 76 75 104 85

	2	3	4	5	6	7	8	9	0
K1		X	X	X	X	X	X		
K2									
K3									
K4									
K5									

As we can see, when we get to the **93** we have 6 different values for the last digit so we know that the keyword can't be 5 letters long. We can also do the same thing to test for a 6-letter keyword and discover that for the last position there are 6 different values:

	2	3	4	5	6	7	8	9	0
K1			X		X	X	X		
K2	X			X		X			
K3			X	X		X	X		
K4		X		X	X		X		
K5			X		X	X	X		
K6		X	X	X	X	X	X		

This gives us a high confidence that the keyword is 4 letters long and we can proceed to the next step with that assumption.

Mapping the Keyword Values

With our table we created when checking the possibilities for a 4-letter keyword, we also learned a couple of things about what column the keyword must be in by looking at the spread of the digits we found.

9**7** 8**2** 5**7** 4**5** 9**8** 7**4** 5**7** 4**5** 9**7** 8**6** 8**4** 7**8** 6**4** 5**2** 8**8** 6**6** 6**4** 8**6**

8**6** 8**5** 9**8** 7**3** 9**7** 7**5** 10**4** 5**5** 9**4** 7**8** 9**8** 8**6** 9**4** 7**7** 8**8** 6**3** 6**4** 6**7**

8**8** 6**5** 8**8** 7**5** 9**8** 7**3** 5**4** 7**5** 7**5** 9**3** 7**6** 7**5** 10**4** 8**5**

	2	3	4	5	6	7	8	9	0
K1			X	X		X	X		
K2	X	X	X	X	X				
K3			X		X	X	X		
K4				X	X	X	X		

Since the 1st keyword letter adds to produce a 4, 5, 7 and 8, this tells us that the first keyword must end in 3. Likewise for the 2nd keyword letter it must be a 1 for the first column. The 3rd keyword must also be a 3. The last keyword letter can either be a 3 or a 4 since we only observe 4 positions.

We can learn the row that the keyword must be in by doing the same process with the remainder of the values.

97 82 57 45 98 74 57 45 97 86 84 78 64 52 88 66 64 86

86 85 98 73 97 75 104 55 94 78 98 86 94 77 88 63 64 67

88 65 88 75 98 73 54 75 75 93 76 75 104 85

	2	3	4	5	6	7	8	9	0
K1					X	X	X	X	X
K2				X	X	X	X	X	
K3				X	X	X	X	X	
K4			X		X	X	X		

This gives us a clear indication of the row that the keywords must be in. The 1st keyword must be in Row 5, 2nd in Row 4, 3rd in Row 4 and the 4th in Row 3. Putting this together with what we know about the columns, the values for our keyword must be either:

53 41 43 33 or **53 41 43 34**

Since we aren't completely sure about the last one, we will proceed with the other three for now. Write the known values under the corresponding encoded values and do the appropriate subtraction. We can also fill in the known crib text which we were told occurs at the beginning.

97 82 57 45 98 74 57 45 97 86 84 78 64 52 88 66 64 86
53 41 43 3? 53 41 43 3? 53 41 43 3? 53 41 43 3? 53 41
44 41 14 1? 45 33 14 1? 44 45 41 4? 11 11 45 3? 11 45
S O M E T I M E S

86 85 98 73 97 75 104 55 94 78 98 86 94 77 88 63 64 67
43 3? 53 41 43 3? 53 41 43 3? 53 41 43 3? 53 41 43 3?
43 5? 45 32 54 4? 51 14 51 4? 45 45 51 4? 35 22 21 3?

88 65 88 75 98 73 54 75 75 93 76 75 104 85
53 41 43 3? 53 41 43 3? 53 41 43 3? 53 41
35 24 45 4? 45 32 11 4? 22 52 33 4? 51 44

Filling in the Polybius Square

We can start to fill in the Polybius square with known values:

	1	2	3	4	5
1				M	
2					

3			I/J K4?	K4?	
4	O		K3	S	T
5			K1		

This tells us that the 1st of the keyword must be either **W** or **X** because there are 6 letters after T in the alphabet and one of them must be in the keyword. We know the 2nd letter is **O** based on the position already being mapped. The 3rd letter must be either **Q** or **R** because there are three letters between **O** and **S** and one of them must be in the keyword. The last letter must be either **I/J**, **K** or **L** since there are only three unused letters between **I/K** and **O** (**M** is already known to be in the Polybius keyword as **14**). Putting it together we have:

W X	O	Q R	I/J K L

Since the **X** and **Q** don't make any sense, the keyword obviously is **WORK**, so we can fill it in the Polybius square. This tells us that **K** must map to be **34** giving us the remaining values. We also notice that there are only two letters between **T** and **W** so we can fill them in the Polybius square.

	1	2	3	4	5
1				M	
2					
3			I/J	K	
4	O		R	S	T
5	U	V	W		

Iterating over the problem

With this in mind, we go back through the encrypted text filling in the information we have now learned. As we fill in the letters, we also see that **11** maps to **E**, so we can fill that in along the way.

97	**82**	**57**	**45**	**98**	**74**	**57**	**45**	**97**	**86**	**84**	**78**	**64**	**52**	**88**	**66**	**64**	**86**
53	41	43	34	53	41	43	34	53	41	43	34	53	41	43	34	53	41
44	41	14	11	45	33	14	11	44	45	41	44	11	11	45	32	11	45
S	**O**	**M**	**E**	**T**	**I**	**M**	**E**	**S**	**T**	**O**	**S**	**E**	**E**	**T**		**E**	**T**

86	**85**	**98**	**73**	**97**	**75**	**104**	**55**	**94**	**78**	**98**	**86**	**94**	**77**	**88**	**63**	**64**	**67**
43	34	53	41	43	34	53	41	43	34	53	41	43	34	53	41	43	34
43	51	45	32	54	41	51	14	51	44	45	45	51	43	35	22	21	33
R	**U**	**T**			**O**	**U**	**M**	**U**	**S**	**T**	**T**	**U**	**R**				**I/J**

88	**65**	**88**	**75**	**98**	**73**	**54**	**75**	**75**	**93**	**76**	**75**	**104**	**85**
53	41	43	34	53	41	43	34	53	41	43	34	53	41
35	24	45	41	45	32	11	41	22	52	33	41	51	44

		T	**O**	**T**		**E**	**O**		**V**	**I/J**	**O**	**U**	**S**

While we can mostly read this, some obvious letters stand out. **32** must be the letter **H** and **54** must be **Y**. The last word clearly is **OBVIOUS** telling us that **22** is **B**. We can fill that into the Polybius square. This tells us that **55** must be the letter **Z** (which is pretty common). Also given that **U** isn't in the keyword, **Q** won't be either making it **42** and **P** will be in the keyword.

	1	2	3	4	5
1	E			M	
2		B			
3		H	I/J	K	
4	O	Q	R	S	T
5	U	V	W	Y	Z

With this in mind, we go back through the encrypted text filling in the information we have now learned. As we fill in the letters, we also see that **11** maps to **E**, so we can fill that in along the way.

97	**82**	**57**	**45**	**98**	**74**	**57**	**45**	**97**	**86**	**84**	**78**	**64**	**52**	**88**	**66**	**64**	**86**
53	41	43	34	53	41	43	34	53	41	43	34	53	41	43	34	53	41
44	41	14	11	45	33	14	11	44	45	41	44	11	11	45	32	11	45
S	**O**	**M**	**E**	**T**	**I**	**M**	**E**	**S**	**T**	**O**	**S**	**E**	**E**	**T**	**H**	**E**	**T**

86	**85**	**98**	**73**	**97**	**75**	**104**	**55**	**94**	**78**	**98**	**86**	**94**	**77**	**88**	**63**	**64**	**67**
43	34	53	41	43	34	53	41	43	34	53	41	43	34	53	41	43	34
43	51	45	32	54	41	51	14	51	44	45	45	51	43	35	22	21	33
R	**U**	**T**	**H**	**Y**	**O**	**U**	**M**	**U**	**S**	**T**	**T**	**U**	**R**		**B**		**I/J**

88	**65**	**88**	**75**	**98**	**73**	**54**	**75**	**75**	**93**	**76**	**75**	**104**	**85**
53	41	43	34	53	41	43	34	53	41	43	34	53	41
35	24	45	41	45	32	11	41	22	52	33	41	51	44
		T	**O**	**T**	**H**	**E**	**O**	**B**	**V**	**I**	**O**	**U**	**S**

We are only left with a few letters to figure out. **35** must be either **L** or **N**. **N** works out great to give us the word **TURN** which tells us that **L** must be in the keyword. We have only two more to figure out: **21** and **24** in the sequence that maps to a word:

22	21	33	35	24
B		**I/J**	**N**	

If we count the letters between **B** and **H** in the Polybius square we can see that because **E** was already used at **11**, we get **C D F G** to fill the four empty squares giving us.

	1	2	3	4	5
1	E			M	
2		B	C	D	F
3	G	H	I/J	K	N
4	O	Q	R	S	T
5	U	V	W	Y	Z

This fills in our word as

22	21	33	35	24
B		**I/J**	**N**	**D**

Since **21** is right before **B**, we see if that would make a word… **BAIND** or **BAJND**. This tells us **A** is in the keyword in the first row and that **21** must map to one of the remaining letters of the keyword. We know that the keyword uses **A**, **E**, **L**, **M**, **P**, and **X**, starting with **E** in the 1st place, **M** in the 4th place. Remembering that letters could repeat once that they are used, we can put together an order list

E	**A**	**A**	**M**	**A**	**E**
	L	**L**		**E**	**L**
	X	**X**		**L**	**M**
	P	**P**		**M**	**X**
	E	**E**		**X**	**P**
				P	

A quick sight-reading eliminating letters that don't make sense gives us the word **EXAMPLE**. Filling in the table tells us that the **21** maps to **P** and the last word must be **BLIND**.

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

97	**82**	**57**	**45**	**98**	**74**	**57**	**45**	**97**	**86**	**84**	**78**	**64**	**52**	**88**	**66**	**64**	**86**
53	41	43	34	53	41	43	34	53	41	43	34	53	41	43	34	53	41
44	41	14	11	45	33	14	11	44	45	41	44	11	11	45	32	11	45
S	**O**	**M**	**E**	**T**	**I**	**M**	**E**	**S**	**T**	**O**	**S**	**E**	**E**	**T**	**H**	**E**	**T**

86	**85**	**98**	**73**	**97**	**75**	**104**	**55**	**94**	**78**	**98**	**86**	**94**	**77**	**88**	**63**	**64**	**67**
43	34	53	41	43	34	53	41	43	34	53	41	43	34	53	41	43	34
43	51	45	32	54	41	51	14	51	44	45	45	51	43	35	22	21	33
R	**U**	**T**	**H**	**Y**	**O**	**U**	**M**	**U**	**S**	**T**	**T**	**U**	**R**	**N**	**B**	**L**	**I**

88 65 88 75 98 73 54 75 75 93 76 75 104 85

53	41	43	34	53	41	43	34	53	41	43	34	53	41
35	24	45	41	45	32	11	41	22	52	33	41	51	44
N	**D**	**T**	**O**	**T**	**H**	**E**	**O**	**B**	**V**	**I**	**O**	**U**	**S**

We can then read the plaintext as:

Sometimes to see the truth, you must turn blind to the obvious.

21. Complete Columnar Cipher

Div B Div C

The Complete Columnar Cipher shares a lot of similarities to the Railfence cipher. The biggest difference however is that it is taken out as columns instead of diagonally.

When encoding a Complete Columnar cipher, the first step is to determine the number of columns that it will be encoded as. For example, let us we use the phrase:

HERE IS A COMPLETE COLUMNAR EXAMPLE

Ignoring spaces, it is exactly 30 characters long. If we don't pad the example, we see that 30 is 6×5 or 3×10 which means that we can use either 3, 5 or 6 columns. While you can use any arbitrary number of columns, for invitationals and regionals, we are limiting it to 9 or fewer, bumping it up to 11 or fewer for state/national tests. In this case we will pick 5 columns and we write out the phrase row by row into the columns:

H	E	R	E	I
S	A	C	O	M
P	L	E	T	E
C	O	L	U	M
N	A	R	E	X
A	M	P	L	E

Next, we need to pick out the order to take out the columns. While you could just take them out from left to right, that would be way too easy for someone to decode. Often someone will use a keyword and then use the order of the letters to figure out the order. If we chose the keyword **ORDER**, it would give us an order of **34125**. When the same letter appears twice in the keyword, the earlier one is first in the order. Note that there is no requirement to use a keyword, but it can be helpful in helping pick the random order.

3	4	1	2	5
H	E	R	E	I
S	A	C	O	M
P	L	E	T	E
C	O	L	U	M
N	A	R	E	X
A	M	P	L	E

With the order of columns identified, we read off the ciphertext a column at a time grouping the letters five at a time resulting in:

RCELR PEOTU ELHSP CNAEA LOAMI MEMXE

21.a. Solving a Complete Columnar Cipher

Problem to solve:

The following quote by Kellie Elmore has been encoded using a complete columnar Cipher. cipher. You have been told that the decoded text has the word **EVER** in it somewhere. What does it decrypt to?

MSLOI RSCTS THWVD EOMLX SMOYR OFELL

ETYEN STOET OENTF UITEO IEAEY LGPYX

Steps to solution

The process of cryptanalysis is straightforward with the information given.

6. Determine the number of columns that it is likely to contain.
7. Write the cipher in the number of columns.
8. Look to determine if there is any padding to help order columns.
9. Look for the CRIB to help order the columns.
10. Pick an order for the columns.
11. Read out the answer by row in the column order determined.

Determining the Number of columns

First count the number of letters in the ciphertext. In this case, we have 12 groups of 5 which works out to be 60 characters. From there we factor it into possible column lengths. $60 = 5 \times 6 \times 2$ which tells us that we either have 5 or 6 columns since the number of columns must be 9 or less. With two choices we can afford to try them both out.

Write the ciphertext in the number of columns

Given the two possible values of 5 and 6 columns, we can write out the cipher text in both formats to see if anything stands out.

5 Columns				
M	W	R	T	E
S	V	O	O	O
L	D	F	E	I
O	E	E	T	E
I	O	L	O	A
R	M	L	E	E
S	L	E	N	Y
C	X	T	T	L
T	S	Y	F	G
S	M	E	U	P
T	O	N	I	Y
H	Y	S	T	X

6 Columns					
M	T	S	E	O	I
S	H	M	T	E	E
L	W	O	Y	N	A
O	V	Y	E	T	E
I	D	R	N	F	Y
R	E	O	S	U	L
S	O	F	T	I	G
C	M	E	O	T	P
T	L	L	E	E	Y
S	X	L	T	O	X

Look at the Padding

In this case we see two occurrences of the letter **X**. In the 5-column case one appears in the last row but the other occurs in the 8th row. In the 6-column case, both of the **X**s appear in the last row. This is a strong clue that the cipher uses 6 columns, however it is possible that the letter **X** occurs in the plain text in which case we don't want to jump to the conclusion too quickly.

5 Columns				
M	W	R	T	E
S	V	O	O	O
L	D	F	E	I
O	E	E	T	E

6 Columns					
M	T	S	E	O	I
S	H	M	T	E	E
L	W	O	Y	N	A
O	V	Y	E	T	E

I	O	L	O	A
R	M	L	E	E
S	L	E	N	Y
C	X	T	T	L
T	S	Y	F	G
S	M	E	U	P
T	O	N	I	Y
H	Y	S	T	X

I	D	R	N	F	Y
R	E	O	S	U	L
S	O	F	T	I	G
C	M	E	O	T	P
T	L	L	E	E	Y
S	X	L	T	O	X

Look for the CRIB

We are told that the word **EVER** occurs in the plaintext. Since there is only one letter **V** in the entire text we can mark the **V** and then look for any letters **E** and **R** in the rows above and below it. Technically we only need to look for an **R** in the same row of next row since it must come after the **V**.

5 Columns				
M	W	R	T	E
S	V	O	O	O
L	D	F	E	I
O	E	E	T	E
I	O	L	O	A
R	M	L	E	E
S	L	E	N	Y
C	X	T	T	L
T	S	Y	F	G
S	M	E	U	P
T	O	N	I	Y
H	Y	S	T	X

6 Columns					
M	T	S	E	O	I
S	H	M	T	E	E
L	W	O	Y	N	A
O	V	Y	E	T	E
I	D	R	N	F	Y
R	E	O	S	U	L
S	O	F	T	I	G
C	M	E	O	T	P
T	L	L	E	E	Y
S	X	L	T	O	X

In this case it is obvious that the cipher was encoded using 6 columns because there is no **R** anywhere close enough after the **V**.

Pick an order for the columns

There are some very strong clues to the order of the columns. To avoid confusion, we will label the columns *CA* through *CF*.

1. Because **R** is on the next row, we know that **EVER** is split at the **R** with **EVE** being the last three columns and **R** being the first column

2. Column *CC* must be the 1st column because the **R** is in the subsequent row

3. Column *CB* has to be the 5th column because it has the **V** in it.

4. Columns *CD* and *CF* remain to be the 4th and 6th columns. Since we have an **X** in the last row of column *CB* (the 5th column), the **X** in the last row of column *CF* means it should be 6th leaving *CD* to be 4th.

5. To order the last two columns, we can use the letters from the last row which will either be **LSOTXX** or **LOSTXX**. Since **LOST** makes sense as a word, we know that *CE* must be 2nd and *CA* is 3rd. There is nothing special about the last row, you can apply this test to any row.

CA	CB	CC	CD	CE	CF
3	5	1	4	2	6
M	T	S	E	O	I
S	H	M	T	E	E
L	W	O	Y	N	A
O	V	Y	E	T	E
I	D	R	N	F	Y
R	E	O	S	U	L
S	O	F	T	I	G
C	M	E	O	T	P
T	L	L	E	E	Y
S	X	L	T	O	X

Read out the answer in column order

To avoid making mistakes, the easiest thing to do is just copy the columns over in the new column order. From there you can read the answer off by rows.

1	2	3	4	5	6
S	O	M	E	T	I
M	E	S	T	H	E
O	N	L	Y	W	A
Y	T	O	E	V	E
R	F	I	N	D	Y
O	U	R	S	E	L
F	I	S	T	O	G
E	T	C	O	M	P
L	E	T	E	L	Y
L	O	S	T	X	X

SOMETI MESTHE ONLYWA YTOEVE RFINDY

OURSEL FISTOG ETCOMP LETELY LOSTXX

Splitting it up with spaces gives us the full answer:

Sometimes the only way to ever find yourself is to get completely lost

22. Hints for your team

- Get your calculators early (they are inexpensive) so that the students become comfortable with them. Note that they may NOT use a standard scientific calculator used at other Science Olympiad events.
- Watch the twitter feed @NCSO_cb (https://twitter.com/NCSO_cb)
- Do the Practice Exams.
- Pay attention to question scores to decide what to do.
- Take advantage of the 2-letter mistake rule to speed up. If you are down to two letters on an Aristocrat and you are sure of the answer, move on to the next question.
- The Timed question is critical.
- Make Practice Samples.
- Use a pencil and paper and the online tool.
- Learn to guess! Sometimes a quick guess gets you to a result faster. It is ok to backtrack if it doesn't work out.
- Split out the test among students.
- Bring pencils and erasers. Note that highlighters are legal and useful.
- Practice, Practice, Practice!
- Have Fun!

23. A strategy for Coaching

A good way to build a team is to start out with a few of the simpler ciphers to give the kids an early success. If you meet regularly, it helps to make a small test with a couple of ciphers that are new for that meeting. There are plenty of examples out there to draw from, building a sample test from the thousands of questions that are already out there means you don't have to know how to write the test, just find examples you like. If you use the toebes.com/codebusters site to build the test, you can use the Answers and Solutions output to guide you in solving many of the ciphers.

For Division B and Division C, it is useful to have a timed question at the start of the test and treat the first 10 minutes of the meeting time as an actual timed test, starting the timer promptly at the meeting time. This gets the team into the mode of how they would have to operate at an actual event and working together as a team. If you have both a varsity and a JV team, letting them compete against one another to see who gets the timed question first is a good motivator.

Having a few extra questions on the test gives the team something to go home and work on at their own pace and then be able to ask about solutions at the next meeting time. Below are some suggested orderings for teaching the ciphers to the team. Note that you may want to reorder them based on the interest level of the kids and their experience. In general, the strategy is to start with simple successes and then build on those learnings getting to harder ciphers.

It is also important to discuss picking what ciphers to solve. Sometimes it is faster to do several low scoring problems than to do one high value problem. Ultimately scoring is a time management problem as it isn't expected that they can solve all the ciphers on the test. The team should learn to pick the ones that have the highest return for their skills.

Don't ignore the special bonus questions. Solving two of them gives a bonus which equals a typical hard problem and three of them is almost double that.

23.a. Division A Div A

1. **PigPen/Masonic** – This is such a fun cipher for them to play with. It is easy to learn the decoding table and looks like secret writing for them.
2. **Caesar** – This is a simple letter shift of only three letters.
3. **Atbash** – Another simple cipher where A maps to Z and vice-versa. There are only 13 letter mappings.
4. **Tap Code** – Another fun cipher that requires only memorizing a couple of letters
5. **Aristocrat** – Once they understand the mapping of Atbash, they get to learn about random mappings and letter frequencies. Keep the examples simple with lots of hints at first to get them going.
6. **Vigenère** – This is just like the Caesar cipher except that the shift is controlled by the keyword.
7. **Dancing Men** – Another fun symbol substitution. Fortunately, the character mapping is on the resource page, so they don't have to memorize them, but it helps to know a few of the characters to make it go faster.

23.b. Division B Div B

1. **Aristocrat** – Starting out with something that should be familiar to them is a great introduction. Since this is also the timed question and will be the first question on the test it is good to get practice at it. The how to solve guide at Cryptograms.org is a good guide for the discussion.

2. **Caesar and Atbash** – These are simple substitution ciphers that look like an Aristocrat, but are pretty easy to solve.
3. **Porta** – A table-based cipher with a keyword. There are only 13 possible table mappings, but it does take a little learning to map the **A-M** vs the **N-Z** letters.
4. **Symbol Baconian** – A different form of table mapping. Start with a simple one symbol for A and a different symbol for B. Then progress to multiple letter mappings.
5. **Aristocrat K1/K2 Keywords** – When you have a K1/K2 keyword, it can often assist in solving an Aristocrat. Learn about guessing where the keyword might be and filling in obvious alphabet letters.
6. **Patristocrat** –This is not much different from the Aristocrat, except that all the spaces are gone. As a result, many of the word patterns don't work, but you can look for obvious words and high frequency letters. Since all Patristocrats are going to have a K1 or K2 keyword, this builds on the previous learning.
7. **Fractionated Morse** – This uses Morse code as well as the K1/K2 style keywords.
8. **Nihilist Substitution** – This is another table-based cipher, but a little addition is involved to map the letters.
9. **Word Baconian** – This uses the same Baconian table, except that multiple letters stand for A and B. The key is to figure out what pattern was used to construct the A/B table.
10. **Affine** – A completely math-based cipher. You can ignore the math and solve the cipher as an Aristocrat, but there is no guarantee that a letter can't map to itself.
11. **Complete Columnar** – The only transposition cipher. All the letters are there, they just need to learn how they are mixed up.
12. **Cryptarithm** – This can be introduced at any time that the kids are interested. This is more math and logic.
13. **Xenocrypt** – The rules are the same as for an Aristocrat, but the phrase is in Spanish. It will typically be a K1 or K2 alphabet and most test creators use phrases with lots of cognates to make it easier.

23.c. Division C Div C

1. **Aristocrat** – Starting out with something that should be familiar to them is a great introduction. Since this is also the timed question and will be the first question on the test it is good to get practice at it. The how to solve guide at Cryptograms.org is a good guide for the discussion.
2. **Porta** – A table-based cipher with a keyword. There are only 13 possible table mappings, but it does take a little learning to map the **A-M** vs the **N-Z** letters.
3. **Symbol Baconian** – A different form of table mapping. Start with a simple one symbol for A and a different symbol for B. Then progress to multiple letter mappings.
4. **Aristocrat K1/K2 Keywords** – When you have a K1/K2 keyword, it can often assist in solving an Aristocrat. Learn about guessing where the keyword might be and filling in obvious alphabet letters.
5. **Aristocrat K3 Keywords** – While the K3 keyword doesn't necessarily help in solving, learning how to recover a K3 keyword is important.
6. **Patristocrat** –This is not much different from the Aristocrat, except that all the spaces are gone. As a result, many of the word patterns don't work, but you can look for obvious words and high frequency letters. Since all Patristocrats are going to have a K1 or K2 keyword, this builds on the previous learning.
7. **Fractionated Morse** – This uses Morse code as well as the K1/K2 style keywords.

8. **Nihilist Substitution** – This is another table-based cipher, but a little addition is involved to map the letters.
9. **Word Baconian** – This uses the same Baconian table, except that multiple letters stand for A and B. The key is to figure out what pattern was used to construct the A/B table.
10. **Hill 2x2** – A completely math-based cipher. This is an opportunity to learn matrix math and how to efficiently do mod 26 on the calculator.
11. **Complete Columnar** – The only transposition cipher. All the letters are there, they just need to learn how they are mixed up.
12. **Cryptarithm** – This can be introduced at any time that the kids are interested. This is more math and logic
13. **Xenocrypt** – The rules are the same as for an Aristocrat, but the phrase is in Spanish. It will typically be a K1 or K2 alphabet and most test creators use phrases with lots of cognates to make it easier.
14. **Hill 3x3** – Very similar to the Hill 2x2 except that it is a larger math matrix. This only needs to be learned for States/National competition.

24. Resources

National Event Page

- https://www.soinc.org/codebusters-c
- https://www.soinc.org/codebusters-b

North Carolina Event Page

- Division A: https://ncscienceolympiad.ncsu.edu/resources/elementary/codebustersa/
- Division B: https://ncscienceolympiad.ncsu.edu/codebusters-b/
- Division C: https://ncscienceolympiad.ncsu.edu/resources/high-school/codebusters/

Cipher Tools

- https://toebes.com/codebusters/ – has lots of tools for writing exams and solving ciphers.
- http://www.cryptograms.org/tutorial.php – One of the best tutorials for solving Aristocrats.
- http://www.dcode.fr/tools-list#cryptography – Has a lot of tools for encoding/decoding ciphers.
- https://quipqiup.com/ – Solves any Aristocrat or Patristocrat.
- http://www.gregorybard.com/cryptogram.html – includes practice problems and suggested textbooks.

Practice Sample resources

- http://www.cryptogram.org/ – is the website of the American Cryptogram Association (ACA) if you are looking for even more resources or a fun organization to join. Note: I am a member of the ACA and ACA members will be contributing questions for the test and helping run the event.
- http://cryptograms.org/ – Puzzle Baron's site with tons of Aristocrats
- http://www.cryptoclub.org/ – Has sample ciphers to practice on
- https://www.brainyquote.com/quotes/topics.html – Is a great source of quotes to encode. Keep in mind the length of the quotes, however.

25. Creating a test

You can use the template from one of the tests on at https://toebes.com/codebusters/ and just replace the questions with your own. An overview of using the tool can be found at https://www.youtube.com/watch?v=pcz_3ql8ebM

25.a. For Aristocrats/Patristocrats

1. Search for Quotes/Phrases to use. Ideally you want something inspirational, topical or science related. A good quote will have around 20 words and about 100-120 characters. They should have a good distribution of letters nominally matching the standard frequency of English letters:

E	T	AO	NIR	SH	LD	CUPF	MWY	BGV	KQXJZ
13%	9%	8%	7%	6%	4%	3%	2%	1%	-

Table 1 - Frequency of English Letters

The tool automatically checks the phrase and gives a basic idea of difficulty based on a chi-square comparison to the English Frequency. Phrases that start with `it is`, have multiple occurrences of `the` or contain the words `these`, `there`, `little` or `people` tend to be easier. You will also want some samples which have repeated words to use for test questions providing hints. It is good to avoid quotes which are unattributed or by anonymous to allow the author of the quote to serve as an extra hint.

2. Using the Patristocrat or Aristocrat tool https://toebes.com/codebusters/AristocratEncrypt.html (Figure 1) as appropriate, enter the text for the cipher as well as the number of points and the text for the question.

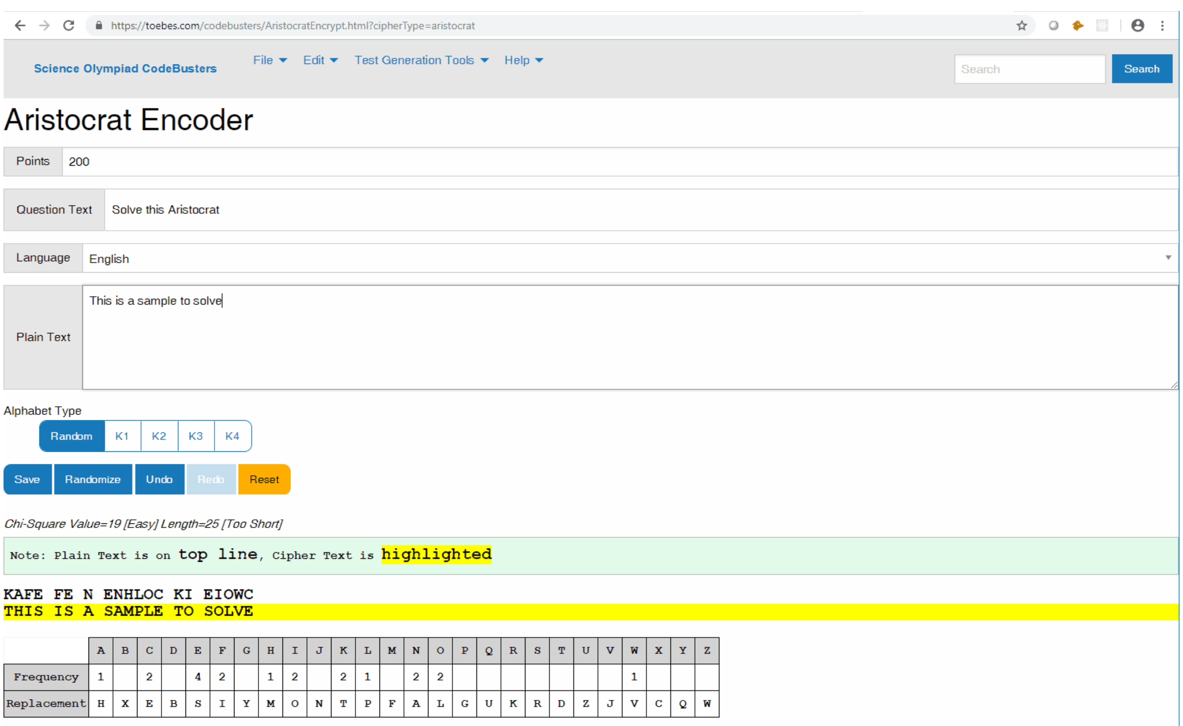
3. 

Figure 1 - Aristocrat Encrypt Tool

25.b. For the Spanish Xenocrypt

http://toebes.com/Ciphers/AristocratSpanishEncrypt.html

1. Pick a Spanish phrase which primarily consists of words which a second-year Spanish class would cover. Phrases which have both `la` and `las` present are good choices as well as phrases which contain `y` or Spanish words which are substantially like their English equivalent words are also good. Although it isn't strictly necessary, try to avoid phrases which depend on accented characters. As with the approach for the English Aristocrats, pay attention to the frequency of letters. You can use the Spanish frequency check tool to verify the difficulty.

E	A	O	SNR	IL	DTUC	MP	BHQ	YVGÓÍ	FJZÁÉÑXÚKWÜ
13%	12%	8%	7%	6%	5%	3%	2%	1%	-

2. Encode using the Spanish Aristocrat encoder at https://toebes.com/codebusters/AristocratSpanishEncrypt.html?cipherType=aristocrat. If the encoded string uses both N and Ñ, you will probably want to re-encode until you don't get them both to avoid confusion on the part of the teams. Although you can also try for an encoding that doesn't use Ñ at all, it is perfectly fine to generate a question which has one.

25.c. Hill Cipher

https://toebes.com/codebusters/HillEncrypt.html

1. Pick a phrase to encode. As a rule of thumb for a 2x2 matrix, every pair of letters is worth 20 points. Ideally you want an odd length string to force them to use a padding Z. For a 3x3 matrix, every group of three letters is worth 25 points. Again, you want a string which is not a multiple of 3 characters long so that they must add the appropriate number of padding characters.

2. Pick an encoding key. For a 2x2 it is 4 characters long and for a 3x3 it is 9 characters long. This is probably the hardest part to making the test because the matrix must be invertible (https://en.wikipedia.org/wiki/Invertible_matrix). Fortunately, the tool will tell you if it is not invertible. There is also a list of known valid keys at https://toebes.com/codebusters/HillKeys.html for both the 2x2 and 3x3 encodings. In general, it is more likely to be invertible if you use the letters B, D, F, H, L, N, R, T, X and Z. as they are odd and non-prime, but you can mix in some other letters. Just make sure that the keyword is not an inappropriate phrase. A total non-sense phrase is perfectly acceptable, but it helps the style of the test if it looks like a word.

3. Use the tool to encode the cipher. The tool can display the math for the problem so that teams can practice and understand what may be wrong with their answers.

25.d. Misspelleedd[sic] Aristocrat

1. Pick a phrase/quote to encode. Ideally this should contain words which have homophones available. The phrase should be about 120-150 characters long as the question is worth 3 points per letter.

2. Use a homophone generation tool (like http://homophonemachine.allaboutlearningpress.com/ or http://evanshort.name/homophone/) or even try dictating through Siri or Dragon type to get a phrase which has been slightly twisted. You may want to try a couple of times to get something that is appealing. Siri has gotten a lot smarter lately and doesn't make as many mistakes as it used to.

3. Encode like a normal Aristocrat using the Aristocrat tool.

25.e. Affine Cipher Basic Question

https://toebes.com/codebusters/AffineEncrypt.html

1. Pick a 5 or 6 letter word to encode which doesn't have the letter `A` in it.

2. Pick a value for *a* which is not coprime with 26 (1,3,5,7,9,11,15,17,19,21,23 or 25). The actual value doesn't matter, but larger ones tend to be slightly harder. If you are generating tests for multiple regions, pick numbers that are near each other. I.e., 7, 9 and 11 would be good to have as equivalent *a* values.

3. Pick a value for *b* between 1 and 25 inclusive. Unlike a where the larger values become slightly harder, the value of *b* can truly be any number and be the same level of difficulty.