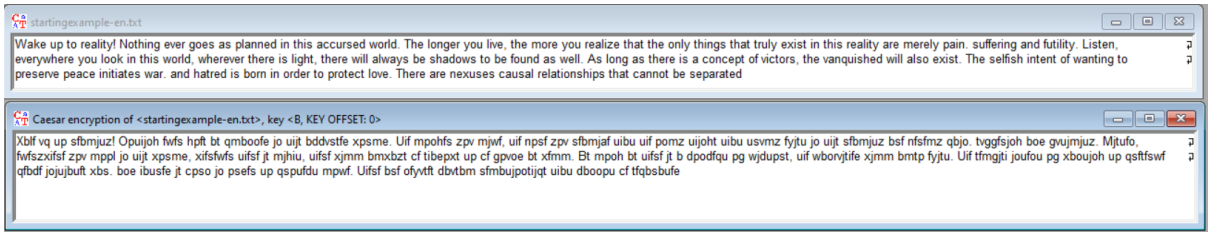
In order to analyze algorithms the following text will be used:

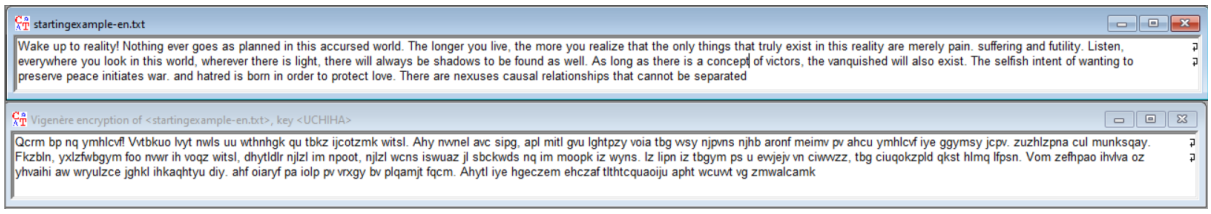
Wake up to reality! Nothing ever goes as planned in this accursed world. The longer you live, the more you realize that the only things that truly exist in this reality are merely pain. suffering and futility. Listen, everywhere you look in this world, wherever there is light, there will always be shadows to be found as well. As long as there is a concept of victors, the vanquished will also exist. The selfish intent of wanting to preserve peace initiates war. and hatred is born in order to protect love. There are nexuses causal relationships that cannot be separated

TASK 1.1

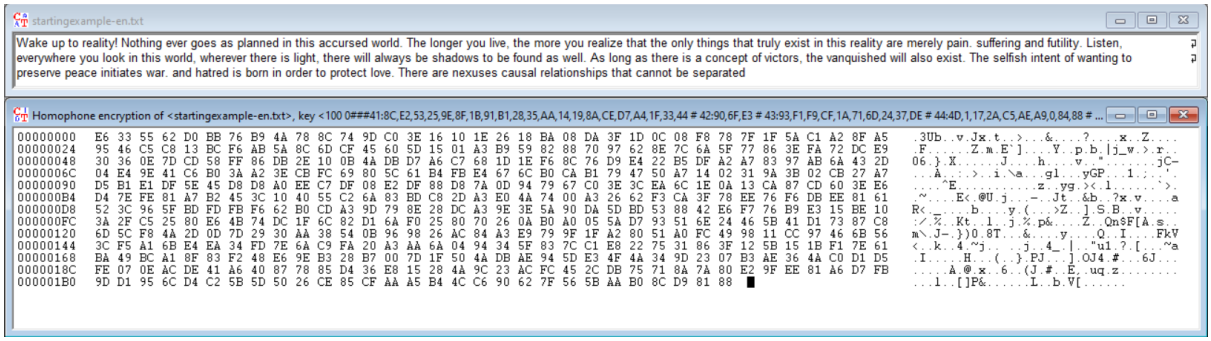
Caesar(key : “V”)



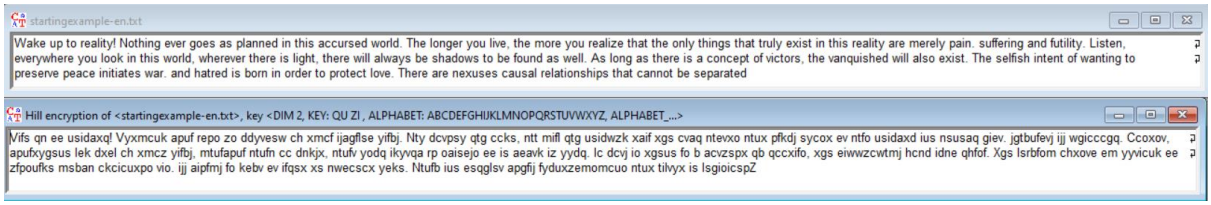
Vigenère(key : “UCHIHA”)



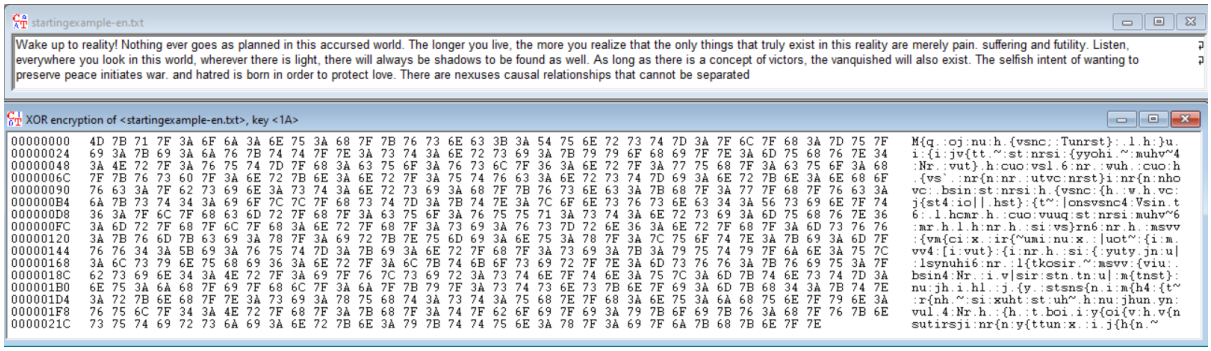
Homophone:



Hill(2x2 matrix, QUZI key)



XOR(key 1A)



TASK 1.2

Hill Cipher:

Space of Keys: The Hill cipher uses a matrix as the key. The size of the matrix depends on the length of the key phrase. Typically, it uses a 2x2 or 3x3 matrix for encryption.

Effect of Multiple Encryption: Repeated encryption using the Hill cipher can increase the security, but it also increases the complexity of the decryption process. However, multiple encryption with the same key matrix does not improve security, as it can be broken with known-plaintext attacks.

Weak Keys: The Hill cipher is vulnerable to attack when the key matrix is singular (i.e., its determinant is 0 or not relatively prime with the modular base). This can make it easier to break the encryption.

Caesar Cipher:

Space of Keys: The Caesar cipher uses a single integer key representing the shift value, which can range from 1 to 25 for the standard implementation.

Effect of Multiple Encryption: Repeated encryption using the Caesar cipher does not increase security. It is a simple substitution cipher and is vulnerable to brute-force attacks.

Weak Keys: The Caesar cipher only has 25 possible keys, making it susceptible to brute-force attacks.

Vigenère Cipher:

Space of Keys: The Vigenère cipher uses a keyword as the key. The space of keys depends on the length of the keyword, which can vary in length.

Effect of Multiple Encryption: Repeated encryption with the same keyword does not significantly improve security, as it still exhibits periodicity in the ciphertext. However, using longer and non-repeating keywords can increase security.

Weak Keys: The Vigenère cipher is vulnerable to frequency analysis when used with short or repeating keywords. Longer, non-repeating keywords are more secure.

ADFGVX Cipher:

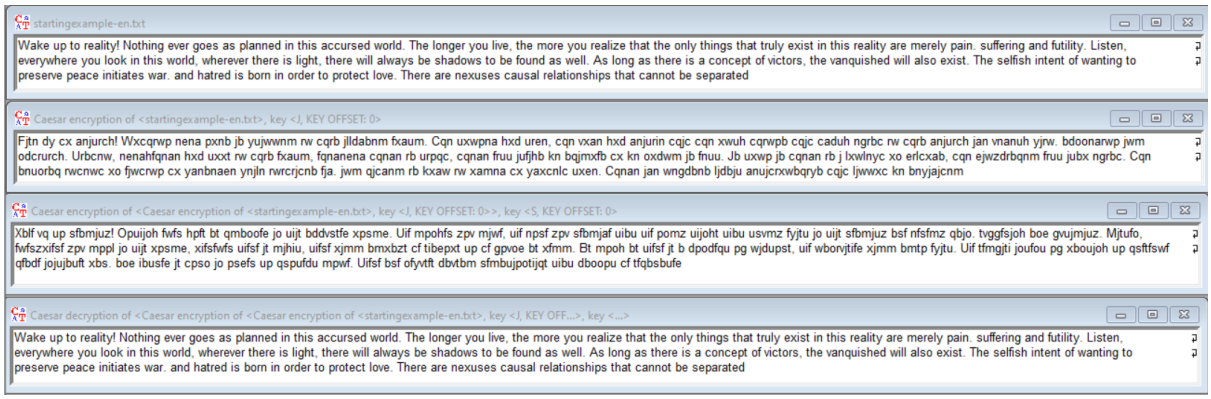
Space of Keys: The ADFGVX cipher uses a keyword for both transposition and substitution. The space of keys is determined by the keyword, which can be relatively large if a long and unique keyword is used.

Effect of Multiple Encryption: Repeated encryption with the same keyword can increase security, but it still maintains certain vulnerabilities, such as frequency analysis.

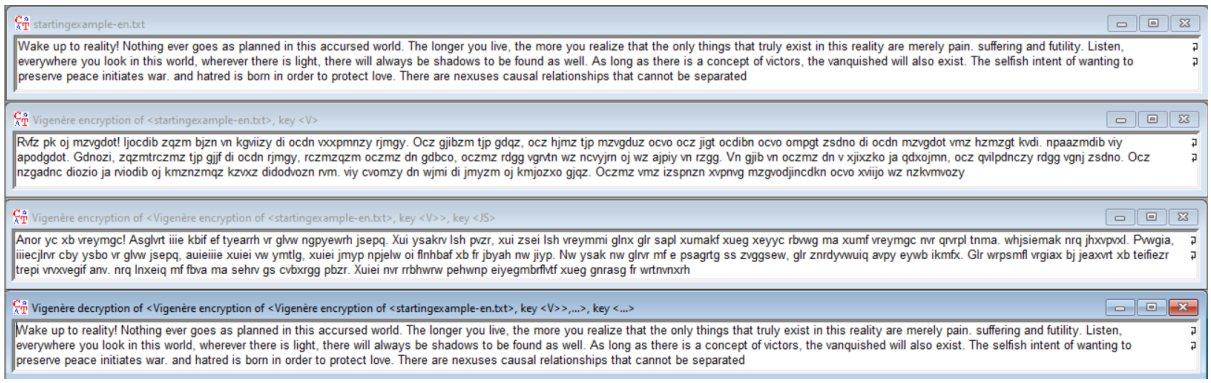
Weak Keys: The ADFGVX cipher can be vulnerable if the keyword is short or has repeating elements. A longer, unique keyword is more secure.

TASK 1.3

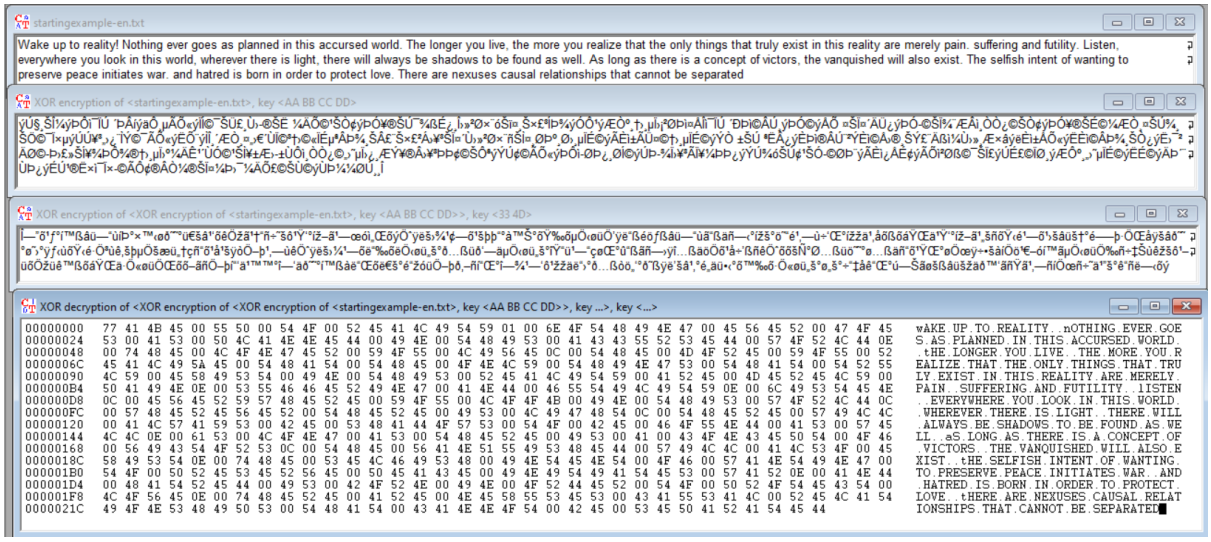
Caesar cypher(firstly with “J” key, then with “S” key and then deciphered with the use of “B” key)



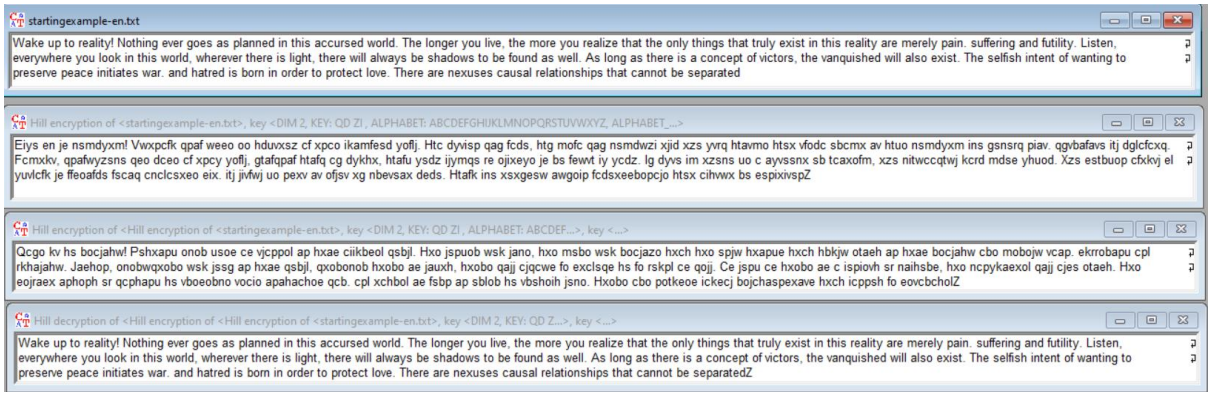
Vigenère cipher(Firstly encrypted with “V” key , then with “JS” key and then decrypted with “EN” key)



XOR cipher(firstly encrypted with “AA BB CC DD”, then with “33 4D” and decrypted with key “B9 D6 DF B0”



Hill(encrypted twice with the use of QDZI 2x2 matrix, and then decrypted with the use of CZZC key)



Multiple encryption does not significantly impede the decryption of text. In the case of mono-alphabetic ciphers, like Caesar's cipher, applying multiple encryptions is akin to a single encryption with a different key length. This holds true for classical substitution algorithms as well; assuming, for example, that A maps to 4 and then 4 maps to 8, is essentially equivalent to assuming A maps directly to 8. Therefore, multiple encryptions do not substantially affect the complexity of decrypting texts.

In the context of polyalphabetic ciphers, such as the Vigenère cipher, multiple encryption may introduce some additional complexity, primarily in terms of managing the decryption key. However, given the current computing power of computers, this added complexity does not pose a major challenge.

Conversely, in the case of ciphers like Hill or XOR cipher, multiple encryption can significantly complicate the decryption process, as it may render the decryption key considerably more intricate to determine.

TASK 1.4

In summary, among the encryption methods discussed, the ADFGVX cipher stands out as one of the strongest. This historical encryption technique employs a combination of transposition and substitution methods, enhancing its resistance against basic cryptanalysis approaches, such as frequency analysis. The ADFGVX cipher's ability to map plaintext characters to a larger set of possible ciphertext characters makes it a formidable choice for securing information.

For analysing the entropy the following text from Wikipedia was chosen:

Cyberpunk 2077 is a 2020 action role-playing video game developed by CD Projekt Red and published by CD Projekt, based on video game designer Mike Pondsmith's game series. Set in a dystopian Cyberpunk universe, the player assumes the role of "V" (played by Gavin Drea/Cherami Leigh), a mercenary in the fictional Californian city known as "Night City", where they deal with the fallout from a heist gone wrong that results in an experimental cybernetic "bio-chip" containing an engram of the legendary rock star and terrorist Johnny Silverhand (played by Keanu Reeves) threatening to slowly overwrite V's mind; as the story progresses V and Johnny must work together to find a way to be separated and save V's life.

TASK 2.1

Entropy values of plaintext		
English	Polish	French
4.22	4.33	4.15

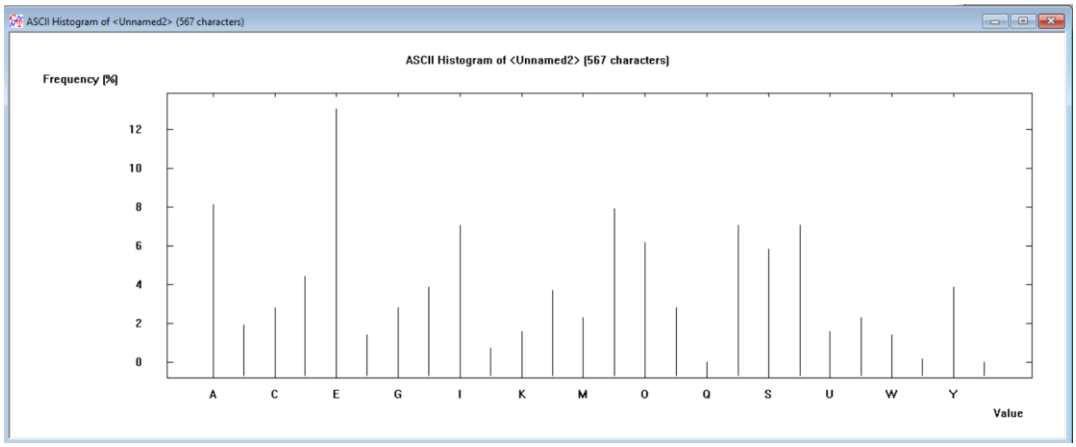
TASK 2.2 & 2.3

The entropy values of plaintext and ciphertext depending on the algorithm		
Algorithm	Plaintext	Ciphertext
Caesar	4.22	4.22
ADFGVX	4.22	2.47
Homophone	4.22	4.41(rescaled)
Permutations	4.22	4.22
Vigenere(V)	4.22	4.22
Vigenere(Somi)	4.22	4.52
Vigenere(Aterlife)	4.22	4.55
Vigenere(JOHNNYSILVERHAND)	4.22	4.61
Hill("KA RA")	4.22	4.6
Hill("FYP CAZ SME")	4.22	4.65
Hill("VRIF ESGE MNAN HDUO")	4.22	4.66

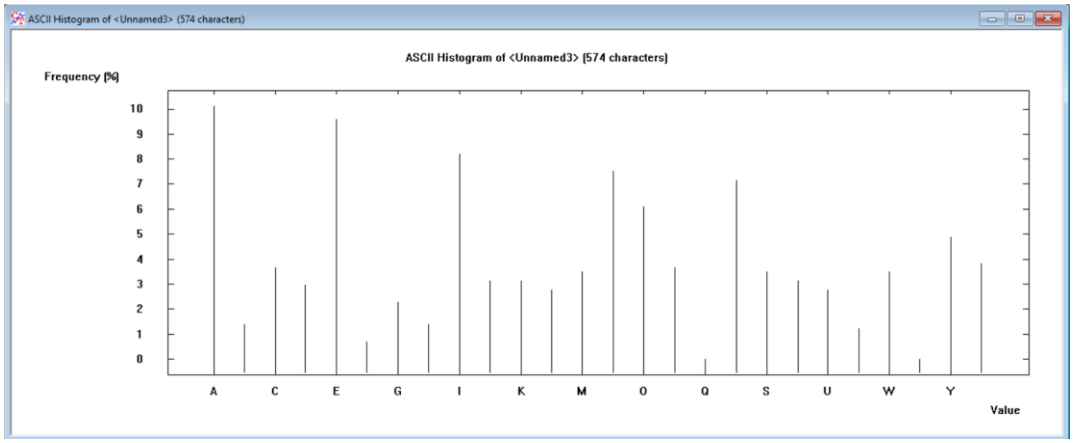
TASK 2.4

Histograms of plaintexts

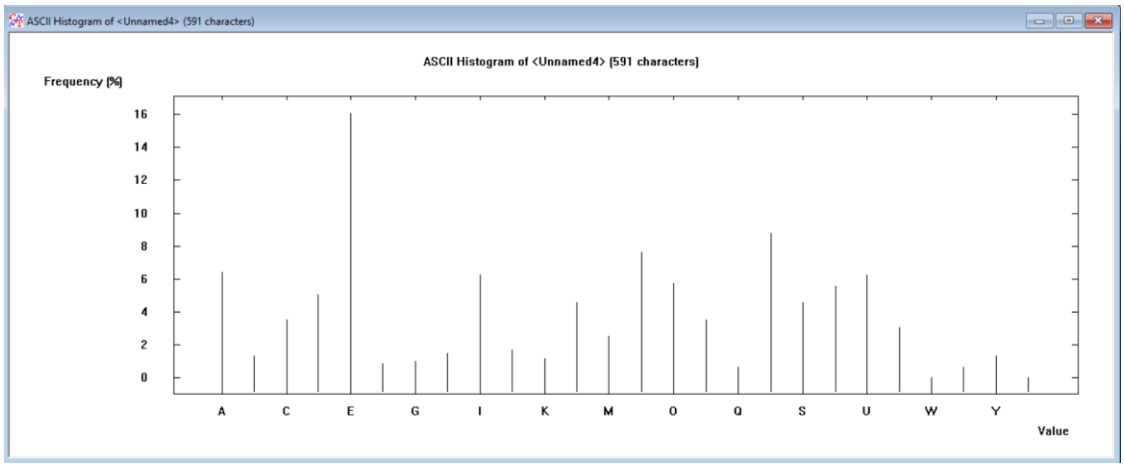
English:



Polish:



French:



TASK 2.5

n-grams (bi/tri/n) of plaintexts

English:

No.	Substring	Frequency (in %)	Frequency
1	E	13.0511	74
2	A	8.1129	46
3	N	7.9365	45
4	I	7.0547	40
5	R	7.0547	40
6	T	7.0547	40
7	O	6.1728	35
8	S	5.8201	33
9	D	4.4092	25
10	H	3.8801	22
11	Y	3.8801	22
12	L	3.7037	21
13	C	2.8219	16
14	G	2.8219	16
15	P	2.8219	16
16	M	2.2928	13
17	V	2.2928	13
18	B	1.9400	11
19	K	1.5873	9
20	U	1.5873	9
21	F	1.4109	8
22	W	1.4109	8
23	J	0.7055	4
24	X	0.1764	1

French:

No.	Substring	Frequency (in %)	Frequency
1	E	16.0745	95
2	R	8.7986	52
3	N	7.6142	45
4	A	6.4298	38
5	I	6.2606	37
6	U	6.2606	37
7	O	5.7530	34
8	T	5.5838	33
9	D	5.0761	30
10	L	4.5685	27
11	S	4.5685	27
12	C	3.5533	21
13	P	3.5533	21
14	V	3.0457	18
15	M	2.5381	15
16	J	1.6920	10
17	H	1.5228	9
18	B	1.3536	8
19	Y	1.3536	8
20	K	1.1844	7
21	G	1.0152	6
22	F	0.8460	5
23	Q	0.6768	4
24	X	0.6768	4

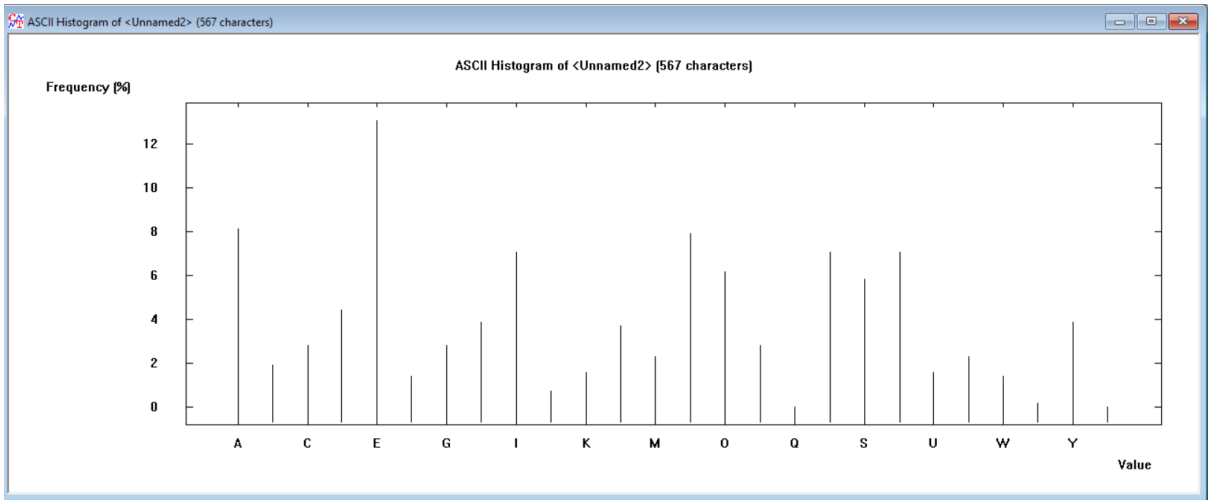
Polish:

No.	Substring	Frequency (in %)	Frequency
1	A	10.1045	58
2	E	9.5819	55
3	I	8.1882	47
4	N	7.4913	43
5	R	7.1429	41
6	O	6.0976	35
7	Y	4.8780	28
8	Z	3.8328	22
9	C	3.6585	21
10	P	3.6585	21
11	M	3.4843	20
12	S	3.4843	20
13	W	3.4843	20
14	J	3.1359	18
15	K	3.1359	18
16	T	3.1359	18
17	D	2.9617	17
18	L	2.7875	16
19	U	2.7875	16
20	G	2.2648	13
21	B	1.3937	8
22	H	1.3937	8
23	V	1.2195	7
24	F	0.6969	4

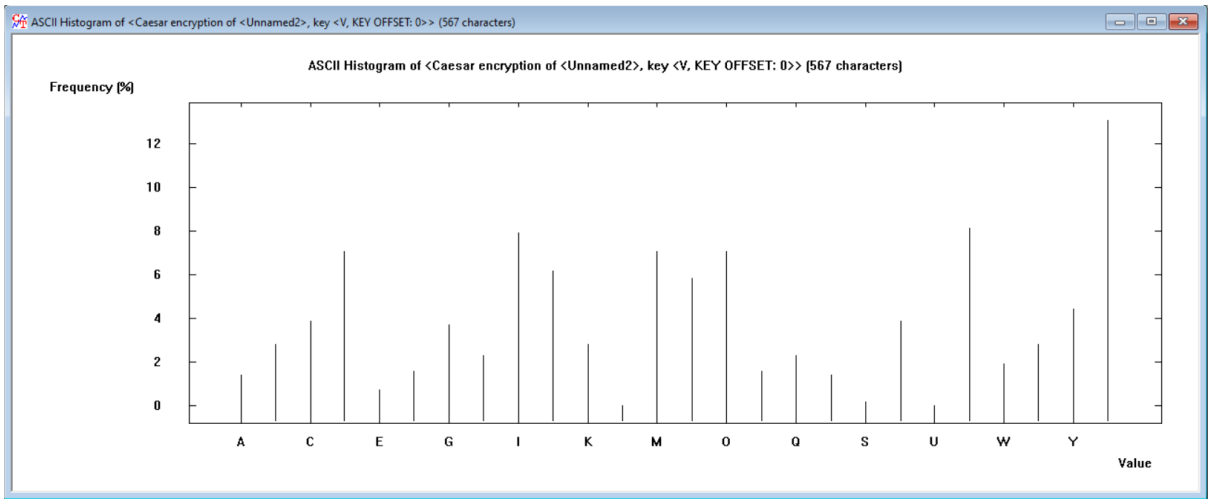
TASK 2.6

Histograms of plaintext and ciphertext depending on the algorithm

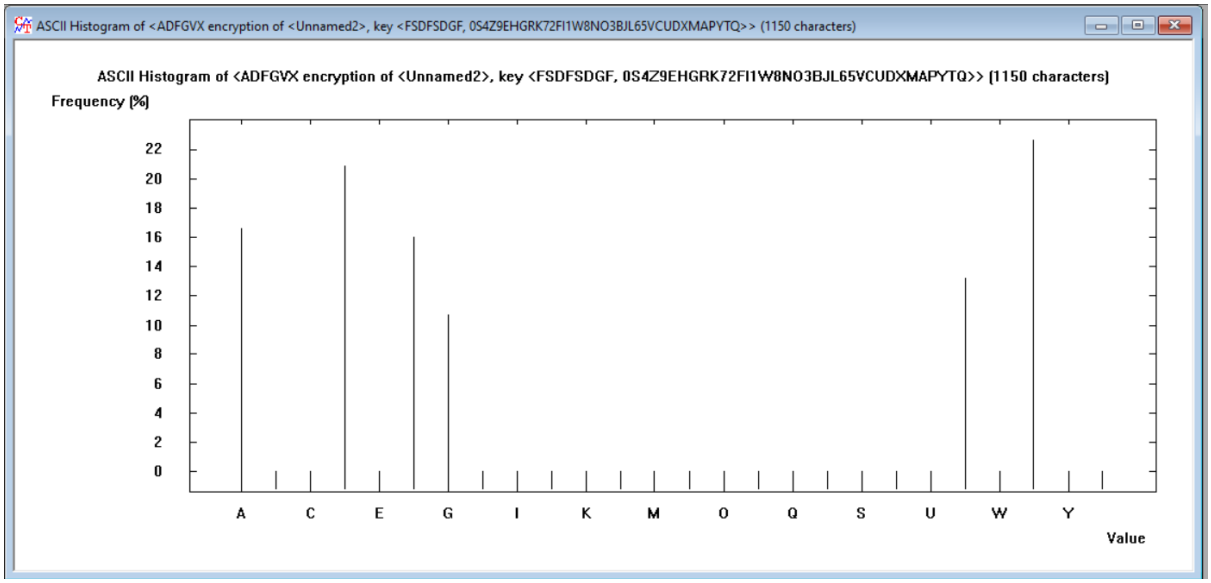
Plaintext:



Caesar:

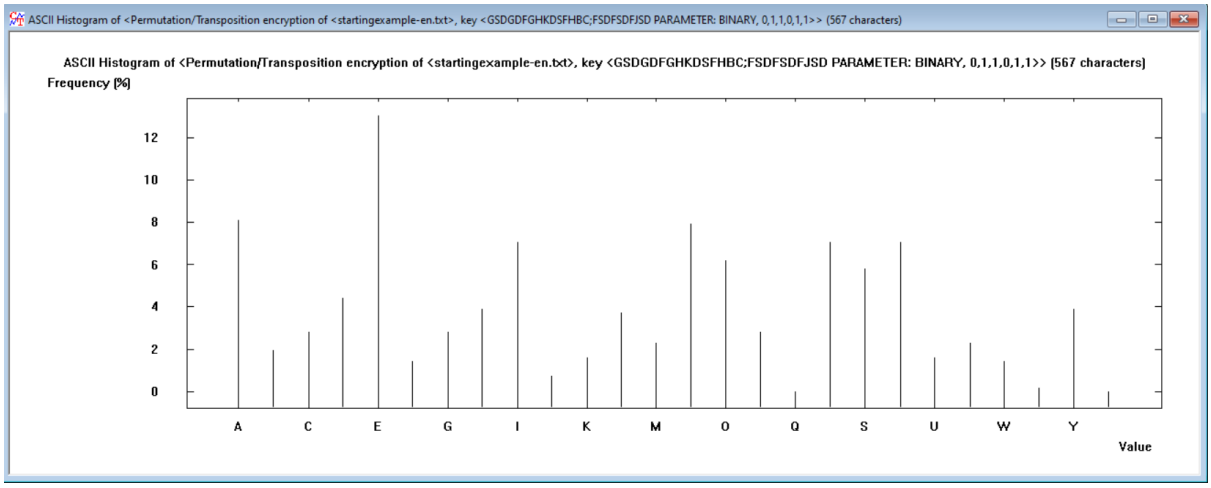


ADFGVX:

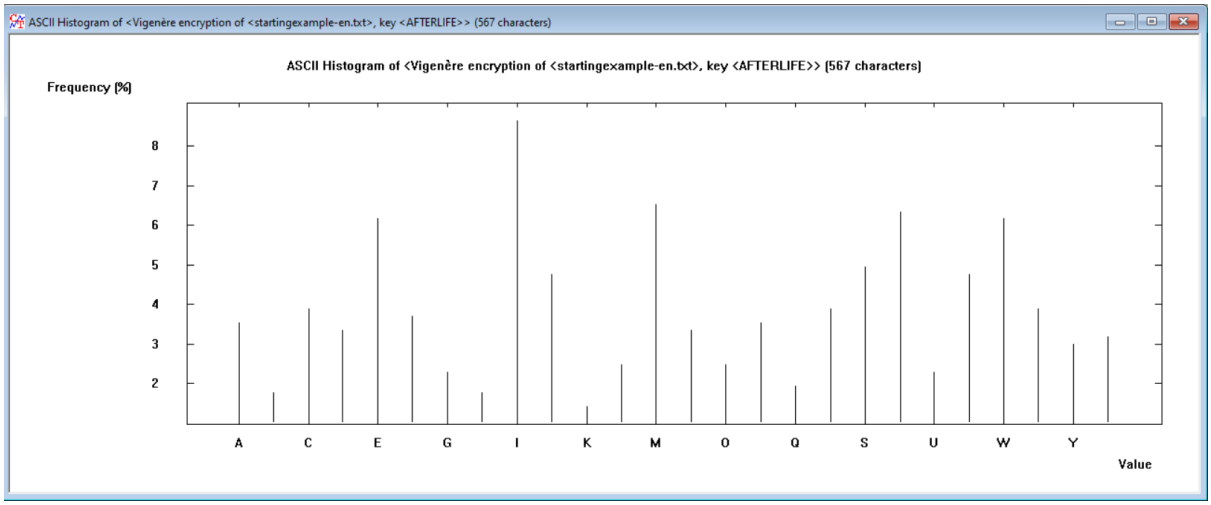


Homophone: the histogram for homophone cannot be generated in CrypTool

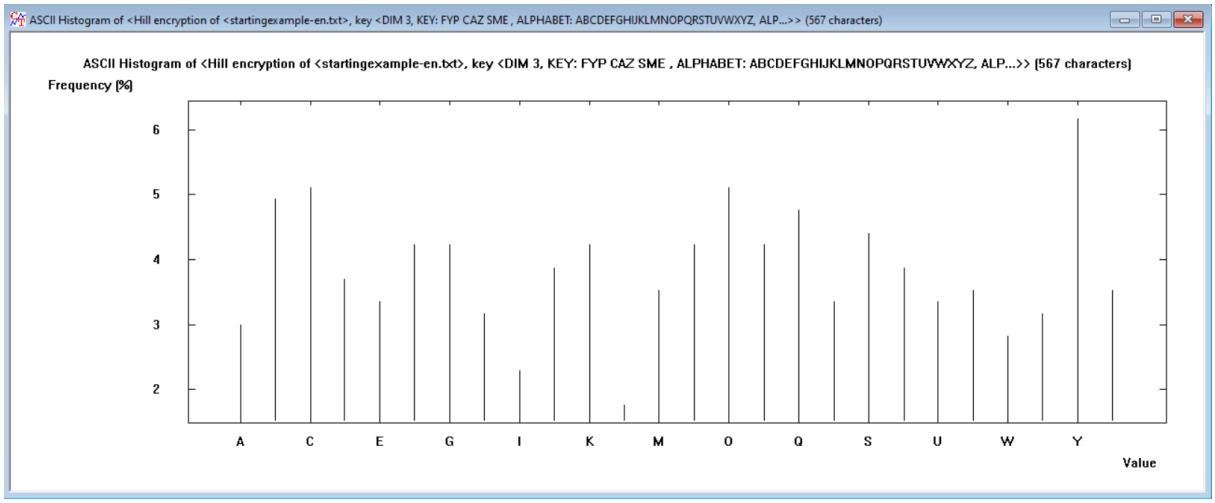
Permutations:



Vigenere(“Afterlife”):



Hill(“FYP CAZ SME”)



TASK 2.7

n-games (bi/tri/n) of plaintext and ciphertext

CAESAR(“V”)

No.	Substring	Frequency (in %)	Frequency
1	Z	13.0511	74
2	V	8.1129	46
3	I	7.9365	45
4	D	7.0547	40
5	M	7.0547	40
6	O	7.0547	40
7	J	6.1728	35
8	N	5.8201	33
9	Y	4.4092	25
10	C	3.8801	22
11	T	3.8801	22
12	G	3.7037	21
13	B	2.8219	16
14	K	2.8219	16
15	X	2.8219	16
16	H	2.2928	13
17	Q	2.2928	13
18	W	1.9400	11
19	F	1.5873	9
20	P	1.5873	9
21	A	1.4109	8
22	R	1.4109	8
23	E	0.7055	4
24	S	0.1764	1

Homophone:

No.	Substring	Frequency (in %)	Frequency
1	U	12.8205	15
2	G	7.6923	9
3	M	6.8376	8
4	F	5.9829	7
5	V	5.9829	7
6	Z	5.9829	7
7	D	5.1282	6
8	J	5.1282	6
9	O	5.1282	6
10	P	5.1282	6
11	B	4.2735	5
12	C	4.2735	5
13	K	3.4188	4
14	L	3.4188	4
15	N	3.4188	4
16	Q	2.5641	3
17	S	2.5641	3
18	A	1.7094	2
19	E	1.7094	2
20	I	1.7094	2
21	T	1.7094	2
22	W	1.7094	2
23	Y	1.7094	2

Vigenere(“Afterlife”)

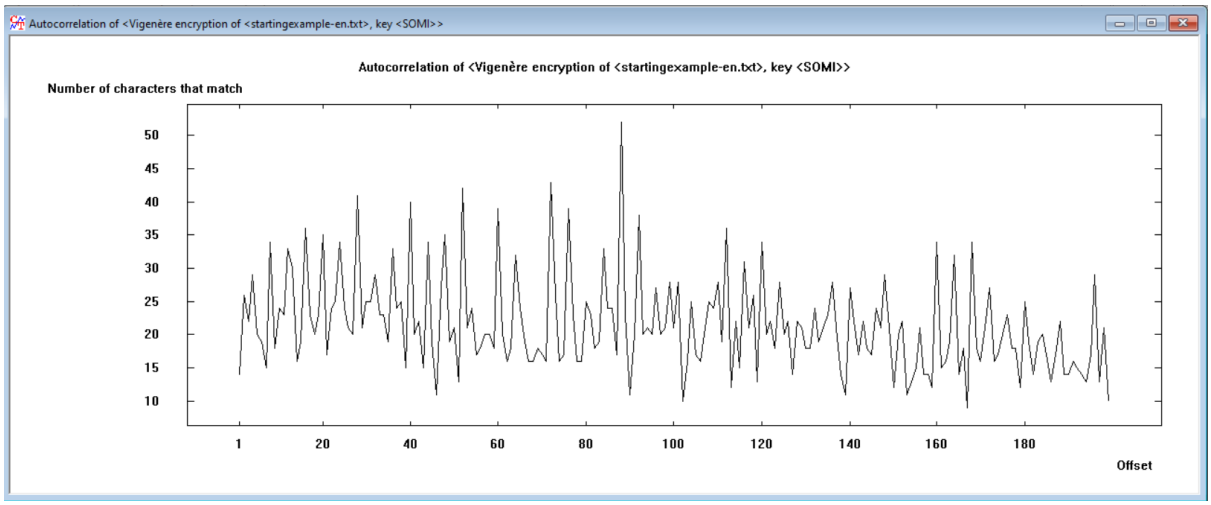
No.	Substring	Frequency (in %)	Frequency
1	I	8.6420	49
2	M	6.5256	37
3	T	6.3492	36
4	E	6.1728	35
5	W	6.1728	35
6	S	4.9383	28
7	J	4.7619	27
8	V	4.7619	27
9	C	3.8801	22
10	R	3.8801	22
11	X	3.8801	22
12	F	3.7037	21
13	A	3.5273	20
14	P	3.5273	20
15	D	3.3510	19
16	N	3.3510	19
17	Z	3.1746	18
18	Y	2.9982	17
19	L	2.4691	14
20	O	2.4691	14
21	G	2.2928	13
22	U	2.2928	13
23	Q	1.9400	11
24	B	1.7637	10
25	H	1.7637	10
26	K	1.4109	8

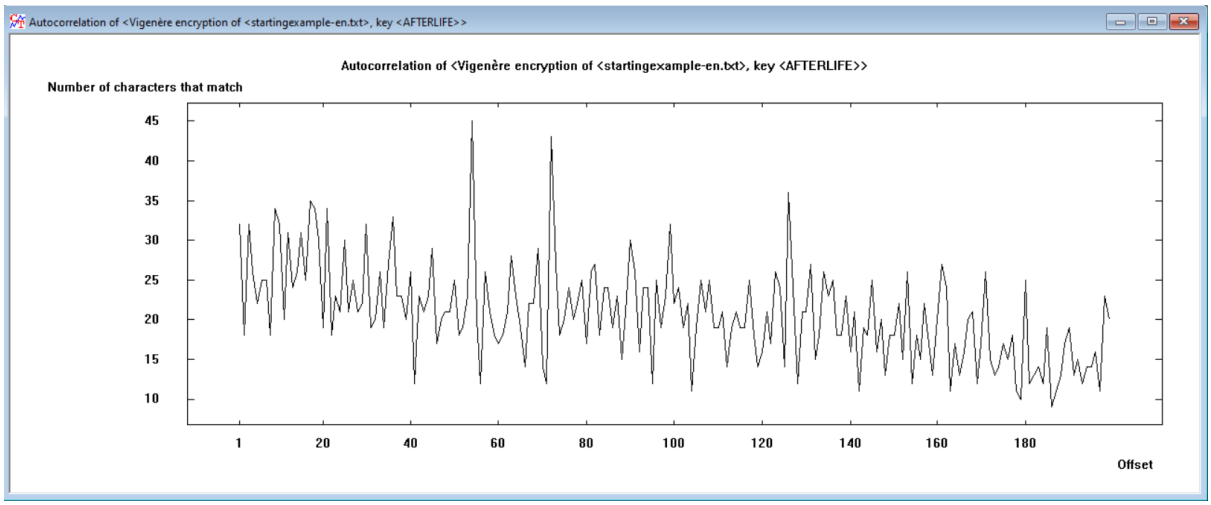
Hill(“FYP CAZ SME”)

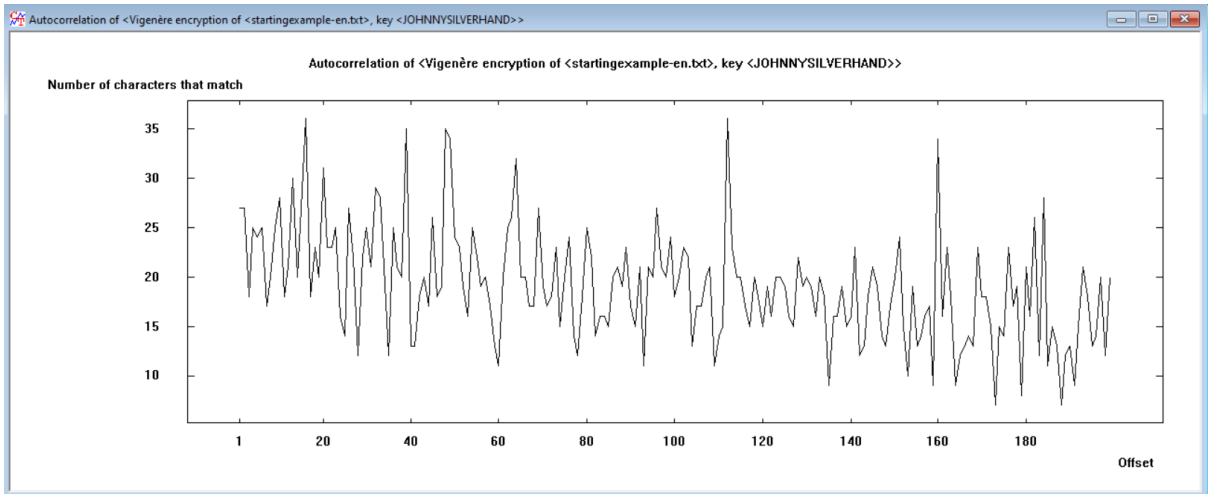
No.	Substring	Frequency (in %)	Frequency
1	Y	6.1728	35
2	C	5.1146	29
3	O	5.1146	29
4	B	4.9383	28
5	Q	4.7619	27
6	S	4.4092	25
7	F	4.2328	24
8	G	4.2328	24
9	K	4.2328	24
10	N	4.2328	24
11	P	4.2328	24
12	J	3.8801	22
13	T	3.8801	22
14	D	3.7037	21
15	M	3.5273	20
16	V	3.5273	20
17	Z	3.5273	20
18	E	3.3510	19
19	R	3.3510	19
20	U	3.3510	19
21	H	3.1746	18
22	X	3.1746	18
23	A	2.9982	17
24	W	2.8219	16
25	I	2.2928	13
26	L	1.7637	10

TASK 2.8

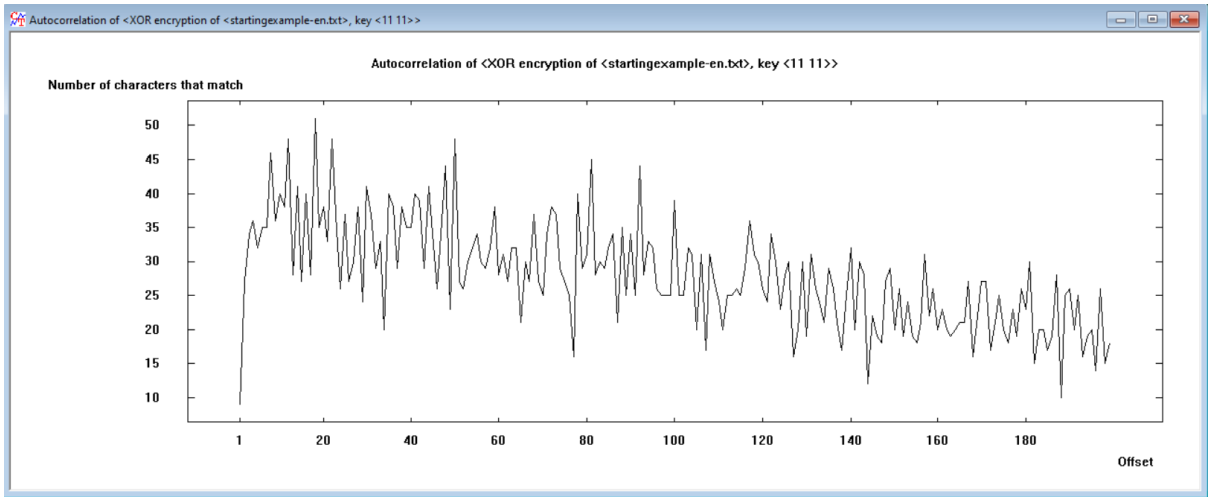
Autocorrelation for Vigenère encryption with different key length

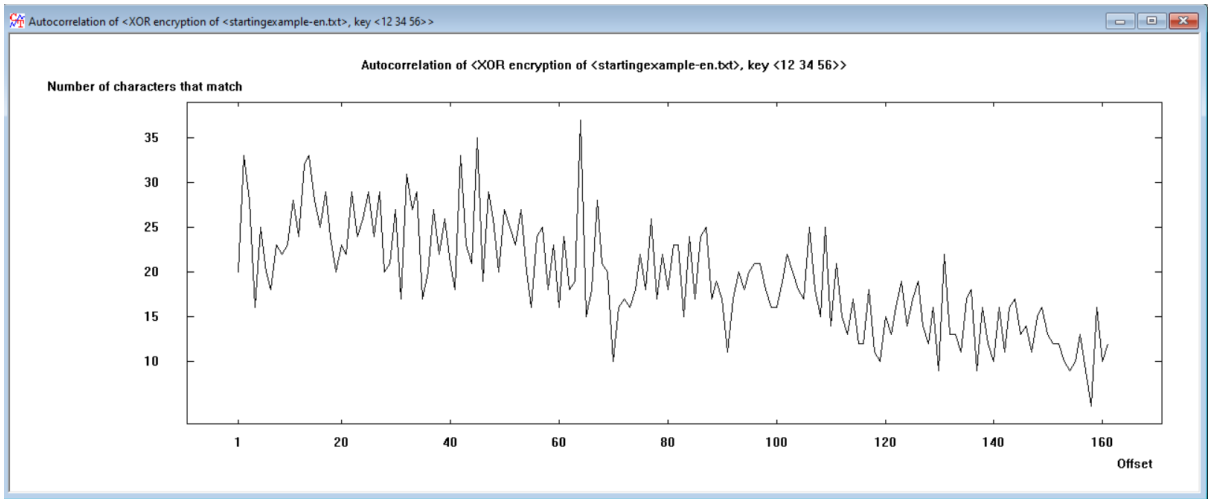


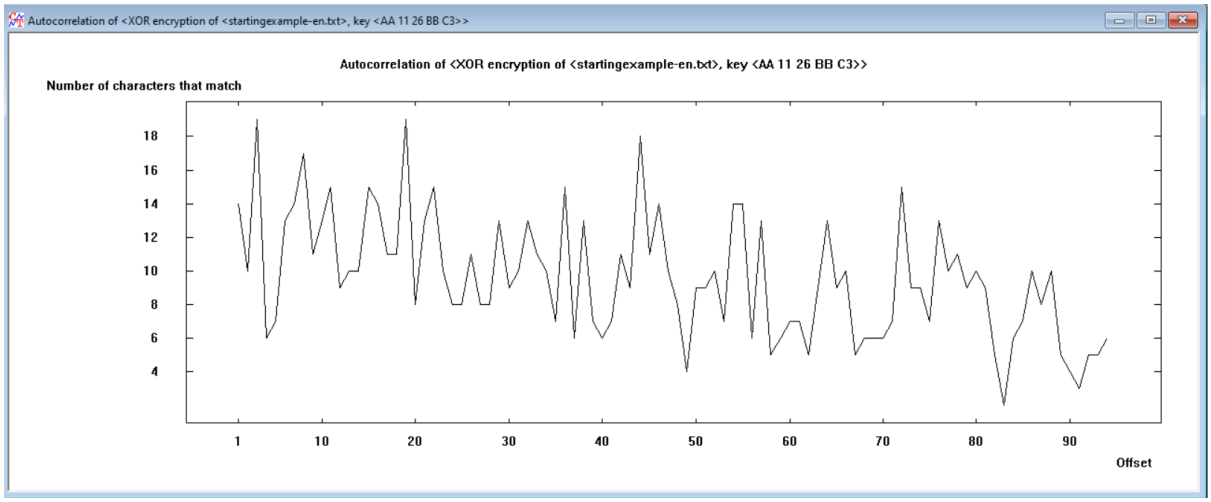




Autocorrelation for XOR encryption with different key length







TASK 2.9

Polish exhibits the highest entropy among the analyzed languages, while French demonstrates the lowest.

In the Caesar cipher, the entropy remains constant, with only character shifting and no loss of information. However, for the transposition and homophonic ciphers, entropy increases due to the appearance of additional characters.

Conversely, the ADFGVX cipher notably reduces entropy by employing only six characters. In the case of the Vigenère cipher, entropy escalates as the key length grows, resulting in fewer repeated characters. A similar pattern is observed in Hill's cipher, where entropy rises with an increasing matrix size.

In all languages under scrutiny, vowels were the most frequently occurring letters. In Polish, "a," "e," and "i" dominated among the consonants, with "n" and "r" also being prevalent. English primarily featured "e," accompanied by "a," "i," "r," and "t." In French, "e" stood out as the most common letter, significantly surpassing others, with "r," "n," and 'a' also exhibiting a relatively high occurrence.

In the context of the Caesar cipher, the histogram retains its shape but shifts along the horizontal axis. Meanwhile, for the ADFGVX cipher, only six repeated letters are observable.

TASK 2.10

Determining the Encryption Algorithm for a Given Ciphertext:

Frequency Analysis: Many encryption algorithms leave characteristic patterns in the ciphertext. You can use the frequency analysis tools in CrypTool to analyze the frequency of characters or n-grams in the ciphertext. For example, if you suspect a simple substitution cipher, you can perform frequency analysis to identify common letter frequencies and patterns.

Known-plaintext Attack: If you have access to a known plaintext-ciphertext pair (a piece of text that has been encrypted), you can use CrypTool to perform a known-plaintext attack. You would try different encryption algorithms until you find one that produces the same ciphertext when given the same plaintext.

Automatic Analysis: CrypTool provides automated analysis tools that can try different encryption algorithms on the ciphertext and evaluate the results based on various criteria. This can help you identify the encryption algorithm used.

Pattern Recognition: You can visually inspect the ciphertext and look for patterns or characteristics that might suggest a specific encryption method. CrypTool's visualization tools can help with this.

TASK 2.11

Determining the Password Used for Encryption:

Brute Force Attack: CrypTool can assist in conducting brute force attacks by trying all possible passwords until the correct one is found. This is typically only feasible for weak passwords or encryption methods with short keys.

Dictionary Attack: You can use a dictionary or wordlist attack to try common passwords. CrypTool allows you to import wordlists and perform dictionary attacks.

Password Cracking Tools: CrypTool integrates various password cracking tools and techniques. For example, you can use it for offline attacks on hashed passwords if you have access to a hash and the corresponding algorithm.

Rainbow Tables: If the password is hashed, you can use CrypTool to generate and search rainbow tables, which are precomputed tables for reversing hash functions.

It's important to note that the success of these methods depends on the strength of the encryption and the password. Strong encryption methods and complex passwords can significantly increase the time and effort required to determine both the algorithm and the password.

TASK 3

Entropy Assessment:

- High entropy, indicating a scarcity of repeating characters, is often associated with the extensive use of Hill's algorithm with a large matrix or Vigenere's algorithm with a lengthy key.

- Conversely, low entropy, characterized by numerous repeated characters, is typically linked to the application of the adfgvx or Caesar algorithms.

Histogram Examination:

- When histograms display frequent repetition of letters, it suggests that the text may have been encrypted using the Caesar or adfgvx algorithm.

- On the other hand, when histograms show a scarcity of repeated letters, it hints at the possibility of the Hill or Vigenere cipher being used.

N-gram Observation:

- If specific character sequences stand out as notably more common than others, it's likely that a Caesar or adfgvx cipher was employed.

- In cases where there are no distinct 3- or 4-character sequences, it's probable that the text is encoded using a Playfair cipher.

- If there's no significant variation in the occurrence of character sequences within groups, it may indicate the presence of a Vigenere or Hill cipher.

Autocorrelation Graph Analysis:

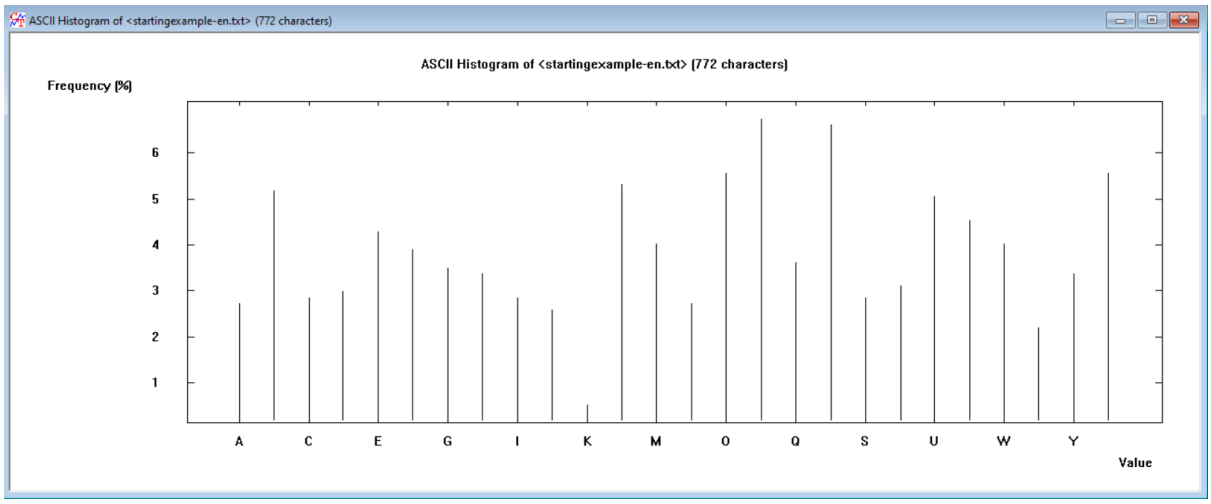
- The presence of a clearly visible periodic pattern in the autocorrelation graph suggests the likelihood of a Vigenere or Caesar cipher.

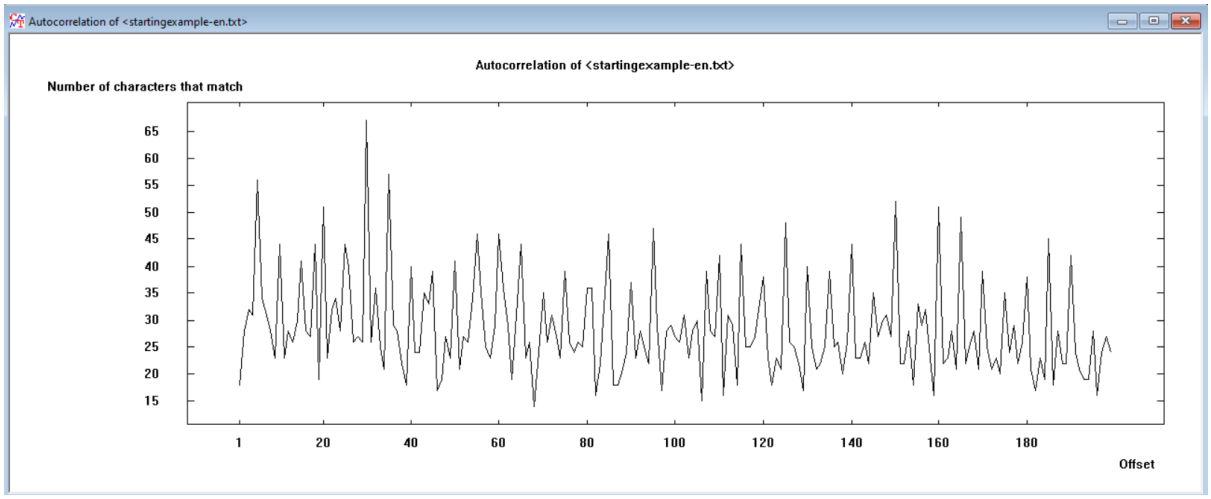
- In contrast, if no periodicity is apparent, the cipher might be a Hill or XOR cipher.

FILE 1_1

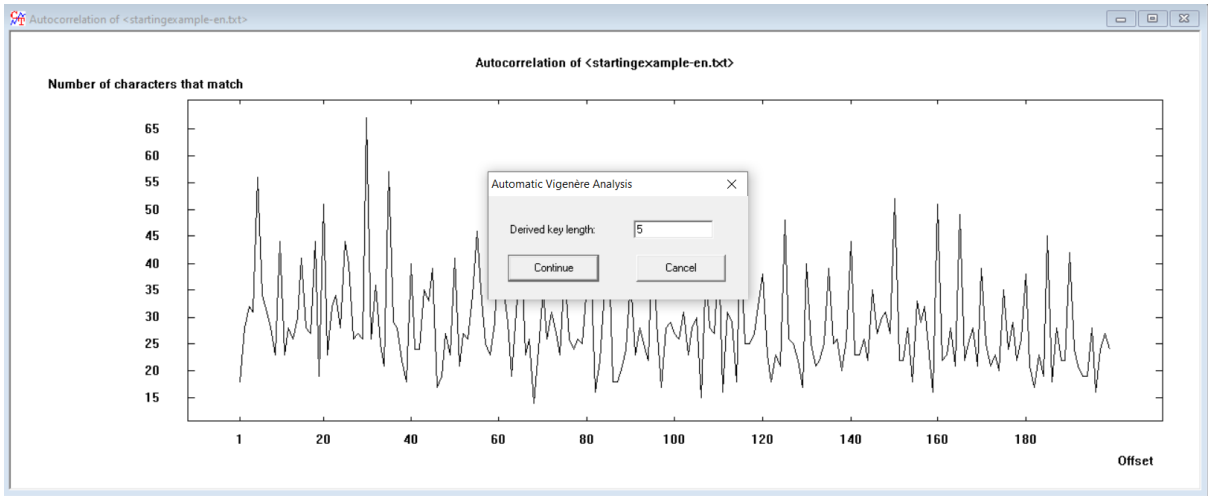
1_1

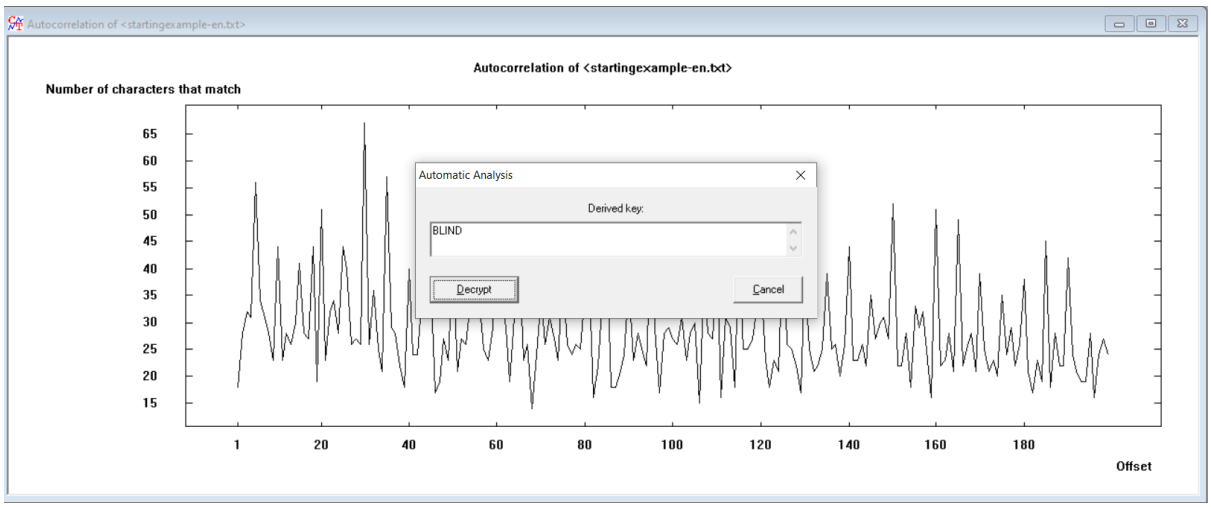
Entropy: 4.59/4.7

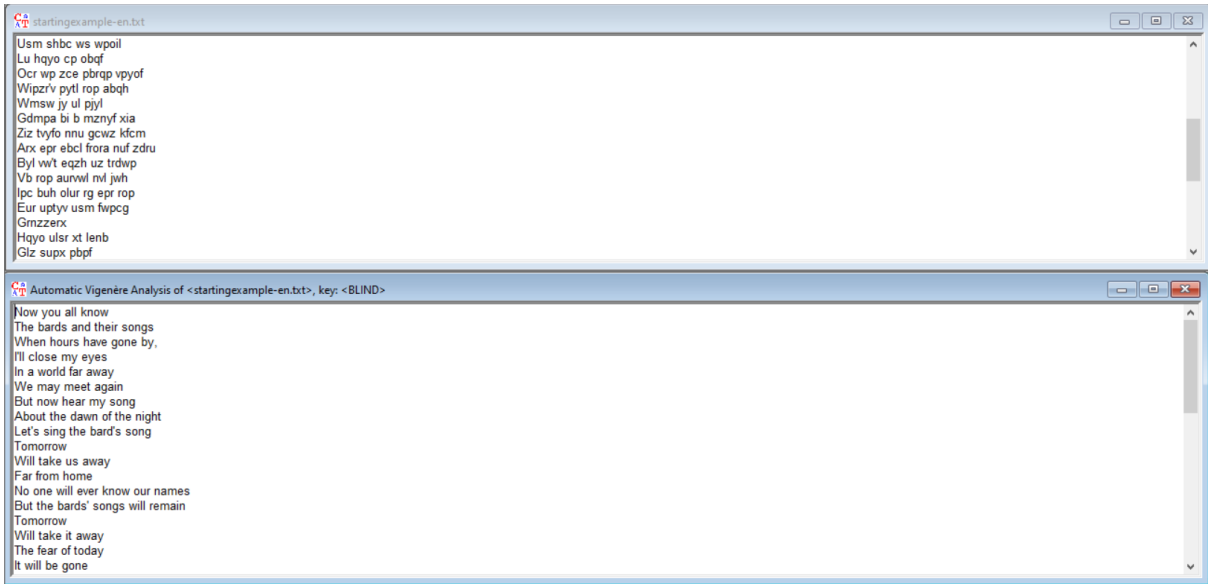




Based on the information displayed I can conclude with high probability that this cypher text is encrypted with the use of Vigenère cipher.

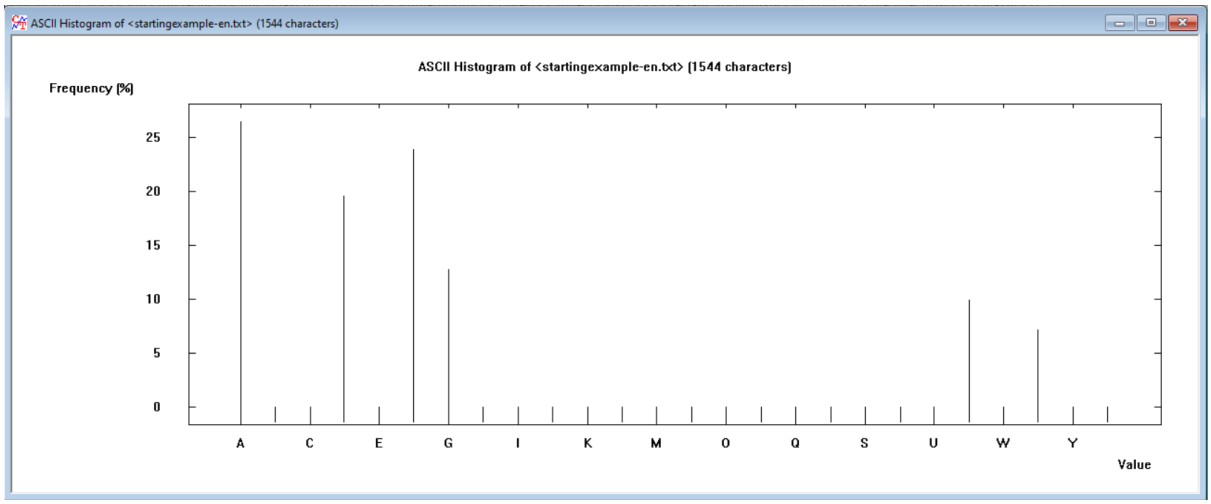

Autocorrelation of <startingexample-en.txt>
Autocorrelation of <startingexample-en.txt>
Number of characters that match
65
60
55
50
45
40
35
30
25
20
15
1
20
40
60
80
100
120
140
160
180
Offset
Automatic Vigenère Analysis
Derived key length:
5
Continue
Cancel


Autocorrelation of <startingexample-en.txt>
Autocorrelation of <startingexample-en.txt>
Number of characters that match
65
60
55
50
45
40
35
30
25
20
15
1
20
40
60
80
100
120
140
160
180
Offset
Automatic Analysis
Derived key:
BLIND
Decrypt
Cancel

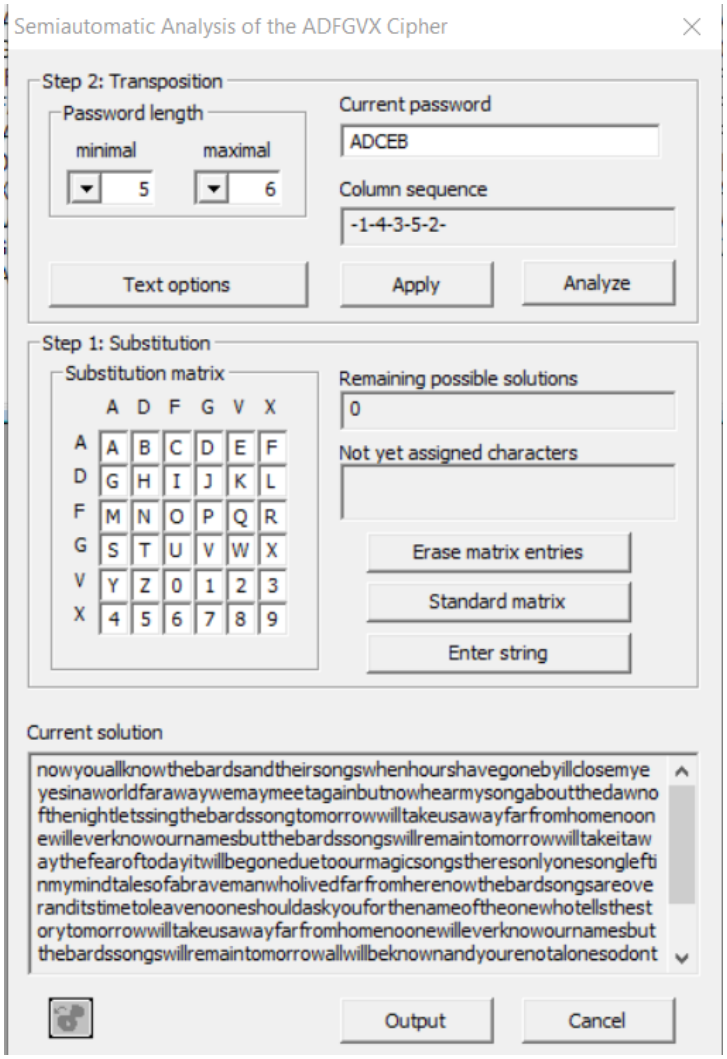

startingexample-en.txt
Usm shbc ws wpoil
Lu hqyo cp obqf
Ocr wp zce pbrqp vpyof
Wipzr'v pytl rop abqh
Wmsw jy ul pjyl
Gdmpa bi b mznyf xia
Ziz tvyfo nnu gcwz kfcm
Arx epr ebcl frora nuf zdru
Byl vw't eqzh uz trdwp
Vb rop aurvwl nvl jwh
Ipc buh olur rg epr rop
Eur uptyv usm fwpcg
Grnzzerx
Hqyo ulsr xt lenb
Glz supx pbpf
Automatic Vigenère Analysis of <startingexample-en.txt>, key: <BLIND>
Now you all know
The bards and their songs
When hours have gone by,
I'll close my eyes
In a world far away
We may meet again
But now hear my song
About the dawn of the night
Let's sing the bard's song
Tomorrow
Will take us away
Far from home
No one will ever know our names
But the bards' songs will remain
Tomorrow
Will take it away
The fear of today
It will be gone

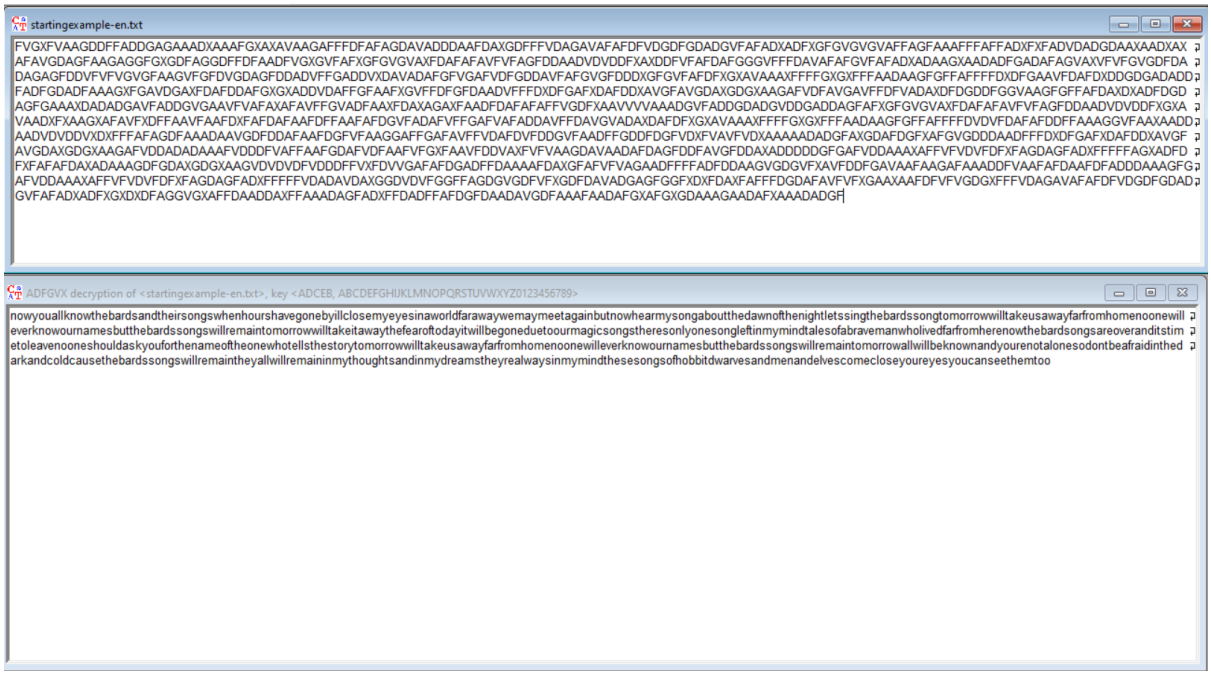
FILE 1_2

Entropy: 2.44 /4.7



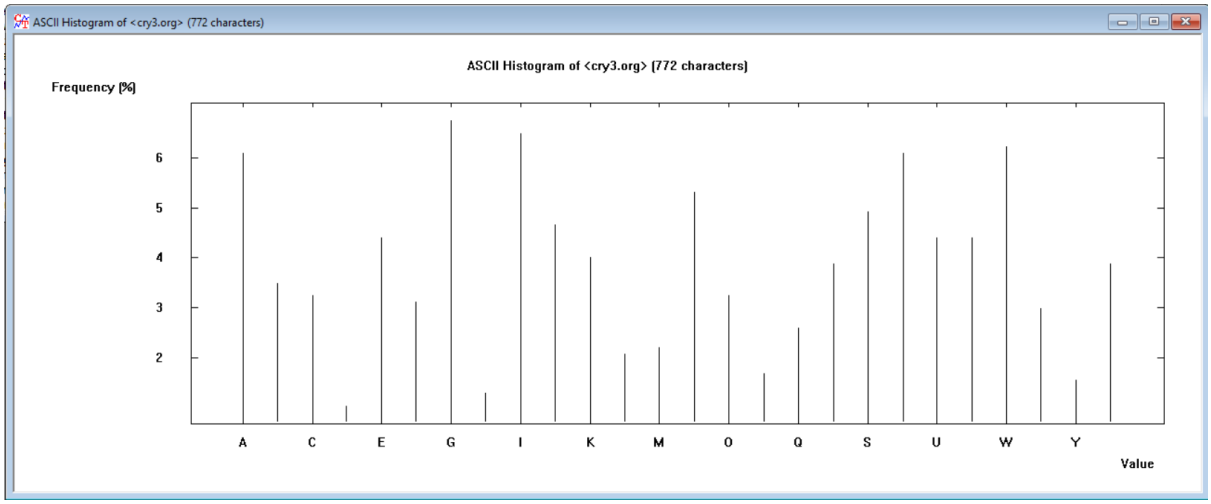
Based on the entropy and graph I can conclude that the algorithm used is ADFGVX

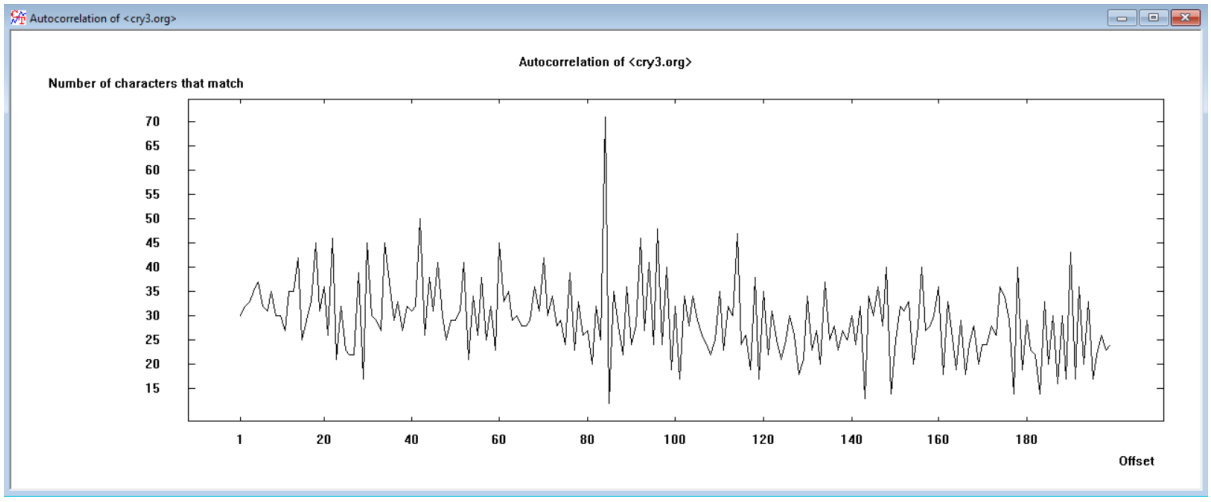




FILE 1_3

Entropy:4.56/4.7

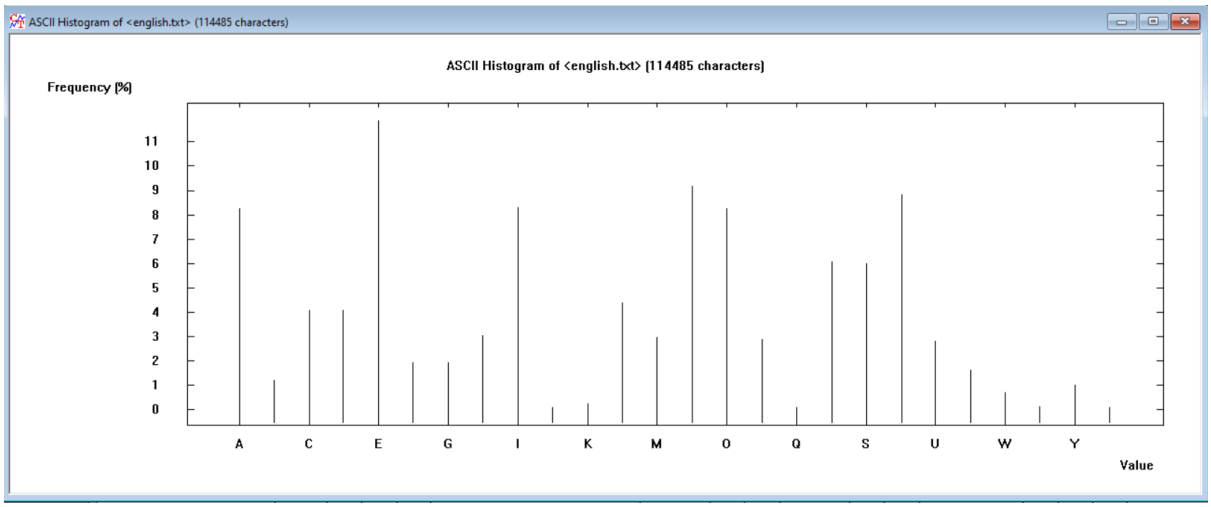


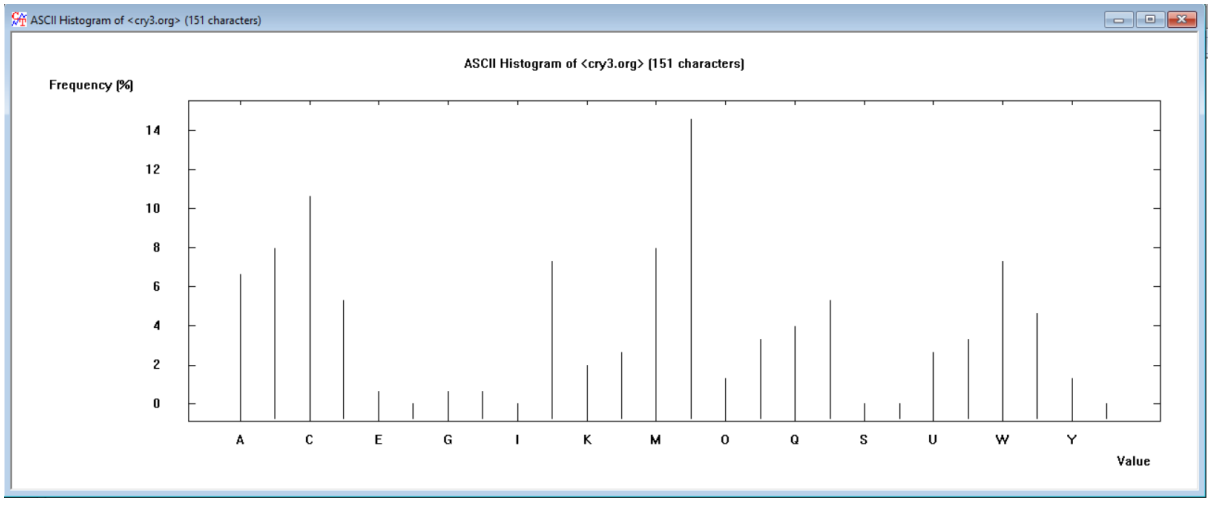


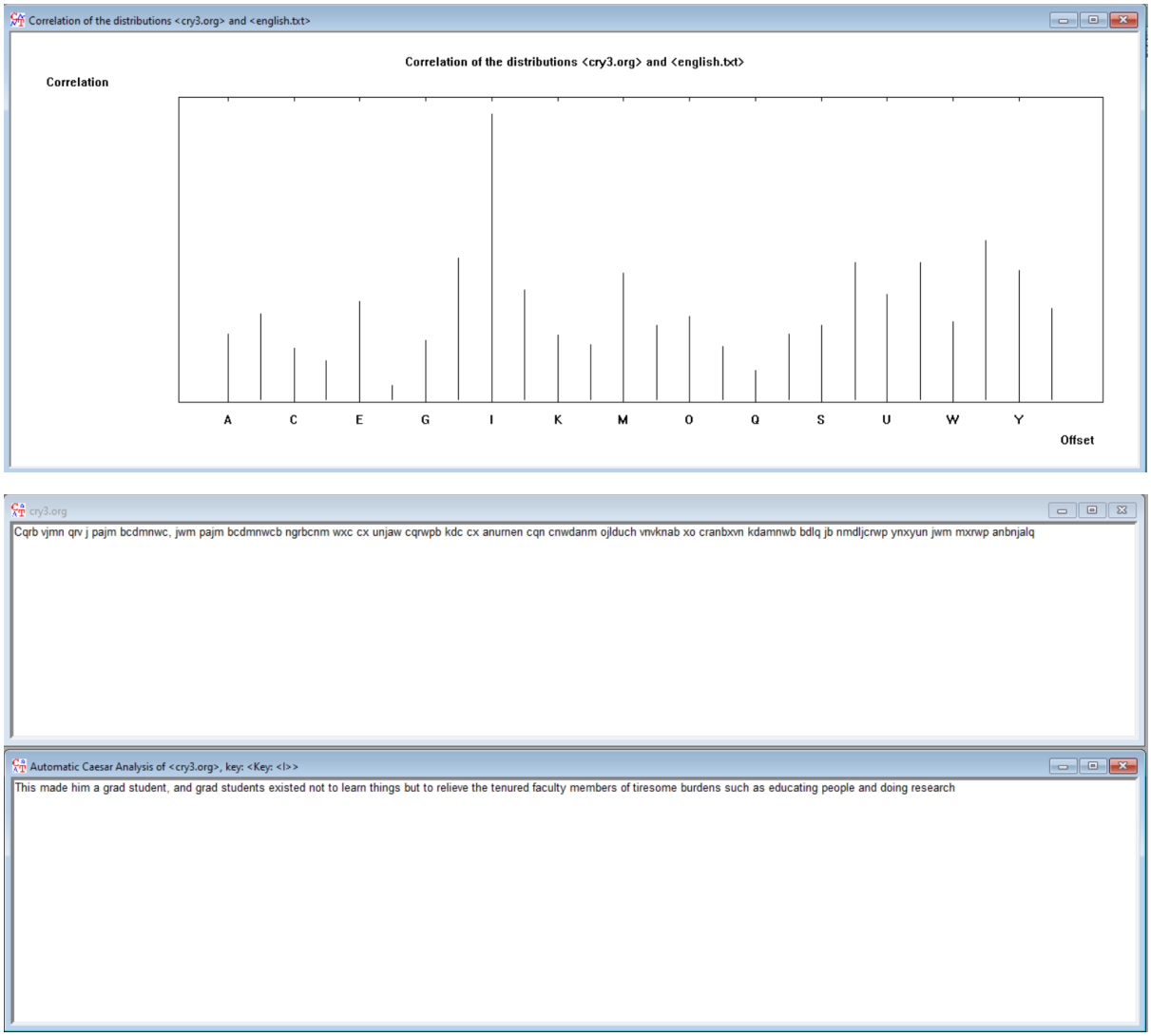
Based on the information collected I can assume that algorithm used is Hill

FILE 2_1

Entropy: 4/4.7







So this is Caesar with shift “I” if we start with A=1(shift “J” if we start with A=0)

FILE 2_2

Based on the format of the text I can assume that this is Playfair cipher was used

FILE 2_3

Based on the format of the data I can assume that XOR cipher was used

TASK 3.4

The strength of encryption hinges on a multitude of factors, encompassing:

- **Key Confidentiality:** The degree of secrecy surrounding the encryption key.

- **Key Guessing Difficulty:** The challenge posed by attempting to guess the key or performing a brute-force search for all potential keys. Longer keys typically present a greater challenge for guessing or discovery.

- **Algorithm Inversion Difficulty:** The level of complexity involved in reversing the encryption algorithm without possessing the encryption key, effectively breaking the encryption.

- **Existence of Back Doors:** The presence or absence of alternative methods for more straightforward decryption of an encrypted file without the knowledge of the key.

- **Known Text Attack:** The capability to decrypt an entire encrypted message by exploiting knowledge of how a portion of it decrypts, known as a known-text attack.

- **Plaintext Characteristics:** The characteristics of the plaintext and an attacker's awareness of these characteristics. For example, a cryptographic system might be susceptible to an attack if all messages encrypted with it consistently begin or end with a known fragment of plaintext.

TASK 3.5

To increase the encryption strength for known ciphers, you can consider the following measures:

Eliminate Outdated Encryption Ciphers: Remove or discontinue the use of outdated or deprecated encryption ciphers. These ciphers may have known vulnerabilities or weaknesses that could be exploited. Ensure that you are using up-to-date, secure encryption algorithms.

Use the Longest Possible Encryption Keys: Whenever feasible, use the longest encryption keys supported by the chosen cipher. Longer keys generally provide stronger encryption and make it more difficult for attackers to break the encryption through brute-force methods.

Encrypt in a Layered Approach: Implement a layered approach to encryption, which involves using multiple encryption methods or ciphers in sequence. This adds an extra layer of security, as even if one layer is compromised, the data remains protected by subsequent layers.

Hold Secret Keys Securely: Ensure that the secret encryption keys used to encrypt and decrypt data are held securely. Use robust key management practices, such as secure key storage and access control, to prevent unauthorized access to the keys.

Make Sure Your Encryption Method Is Used Correctly: Properly implement and use the chosen encryption method. This includes using appropriate modes of operation, strong random number generation for keys and initialization vectors, and secure implementation practices. Even the strongest encryption can be compromised if not used correctly.