Assignment 4

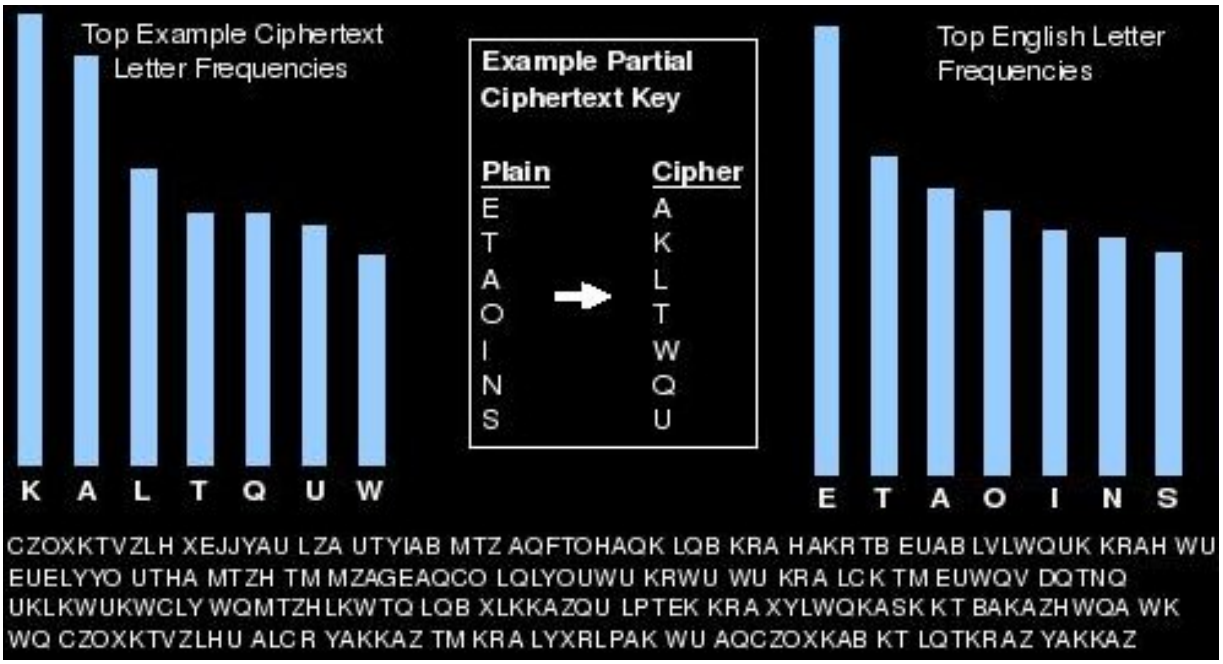
Task 1: Frequency Analysis (100 points)

The cryptanalyst can benefit from some inherent characteristics of the plaintext language to launch a statistical attack. For example, we know that the letter E is the most frequently used letter in English text. The cryptanalyst finds the mostly-used character in the ciphertext and assumes that the corresponding plaintext character is E. After finding a few pairs, the analyst can find the key and use it to decrypt the message. To prevent this type of attack, the cipher should hide the characteristics of the language. Table 1 contains frequency of characters in English.

Table 1 Frequency of characters in English

Letter	*Frequency*	*Letter*	*Frequency*	*Letter*	*Frequency*	*Letter*	*Frequency*
E	12.7	H	6.1	W	2.3	K	0.08
T	9.1	R	6.0	F	2.2	J	0.02
A	8.2	D	4.3	G	2.0	Q	0.01
O	7.5	L	4.0	Y	2.0	X	0.01
I	7.0	C	2.8	P	1.9	Z	0.01
N	6.7	U	2.8	B	1.5		
S	6.3	M	2.4	V	1.0		

Cryptogram puzzles are solved for enjoyment and the method used against them is usually some form of frequency analysis. This is the act of using known statistical information and patterns about the plaintext to determine it. In cryptograms, each letter of the alphabet is encrypted to another letter. This table of letter-letter translations is what makes up the key. Because the letters are simply converted and nothing is scrambled, the cipher is left open to this sort of analysis; all we need is that ciphertext. If the attacker knows that the language used is English, for example, there are a great many patterns that can be searched for. Classic frequency analysis involves tallying up each letter in the collected ciphertext and comparing the percentages against the English language averages. If the letter "M" is most common then it is reasonable to guess that "E"-->"M" in the cipher because E is the most common letter in the English language. These sorts of clues can be bounced off each other to derive the key and the original plaintext. The more collected cipher text the attacker has, the better this will work. As the amount of information increases, its statistical profile will draw closer and closer to that of English (for example). This sort of thing can also be applied to groups of characters ("TH" is a very common combination in English for example). The example frequency analysis image above was performed on the first three sentences of this paragraph turned into a cryptogram. As you can see, the English language is very predictable with regard to letter frequency and this can exploited in some situations to break ciphers.



The goal of this lab is to gain a better understanding of a statistical attack by programming some of the important components to analyze/manipulate arrays of characters.

Lab Questions

1. What type of cipher is this program useful for breaking?
2. In this type of cipher, the relationship between characters in the plaintext and characters in the ciphertext is _____.
3. List the frequencies for the top 4 characters found in the given ciphertext:

   MKLAJZHAIUQWKHJABZNXBVHAGKFASDFGALQPIWRYIOQYWIERMASVZMNBZXCKJASDFGLKJFH
   WQERYIOQWTYIOASUDYFLASKJDHFZMZVBCXMVQLWERYIQRASDFQIWUERYIHKMFMAKHLSDFY
   UIOQWYREIORYIWQEUFHAKDFHLKASHFKVBBBNASMDFSADFWQEUYRUUEYRUUUQKASJHFKJDS
   HFSNBNBNBNBABABAAASKJFHLKJSADHFIDUASFOYDASIYFQWERBQWBRKLJLKASSADFDFDASDA

4. Break the cipher text given in the following. What is the plaintext? What is the key?

OTWEWNGWCBPQABIZVQAPMLJGZWTTQVOBQUMAPMIDGZCAB

EQVBMZLZIXMLAXZQVOQVLMMXAVWEIVLLIZSNZWAB

JQZLWNLMTQOPBVIUMLGWCBPAEQNBTGTMNBBPMVMAB

ITIAKWCTLVBBQUMQBEPQTMQBEIAQVUGBZCAB

What To Turn in:

A zip file named lastname_firstname_Assignment3.zip, containing:

- Source code
- A text file or Word document containing yours answers to the Lab Questions
- If you changed any other files in your project, please include them as well

Task 2: Lab on encryption using binary/byte addition (practice)

Under this encryption algorithm, the key entered is added character by character (byte by byte) to the data to be encrypted. Here addition modulo 256 is used, i.e. so that any carry-overs are ignored. The key is applied cyclically (as under the Vigenère encryption algorithm and also with the Exclusive-OR), i.e. once all the characters (bytes) of the key have been used, the algorithm reverts to the first character until the text has been completely encrypted.

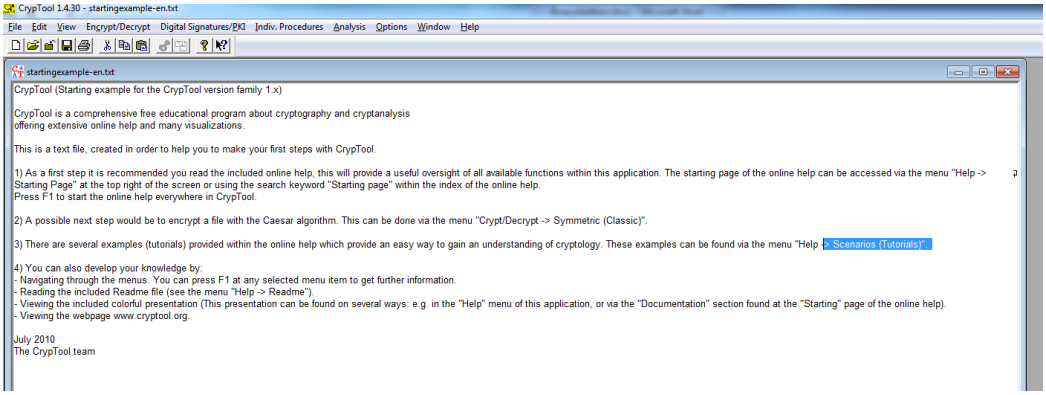
To decrypt the text, the characters of the key have to be subtracted from the encrypted text modulo 256.

If one knows the characters which occur most frequently in the plaintext, it is then possible to work out the key with the aid of a computer (and hence also the plaintext) (see Automatic analysis, Byte Addition).

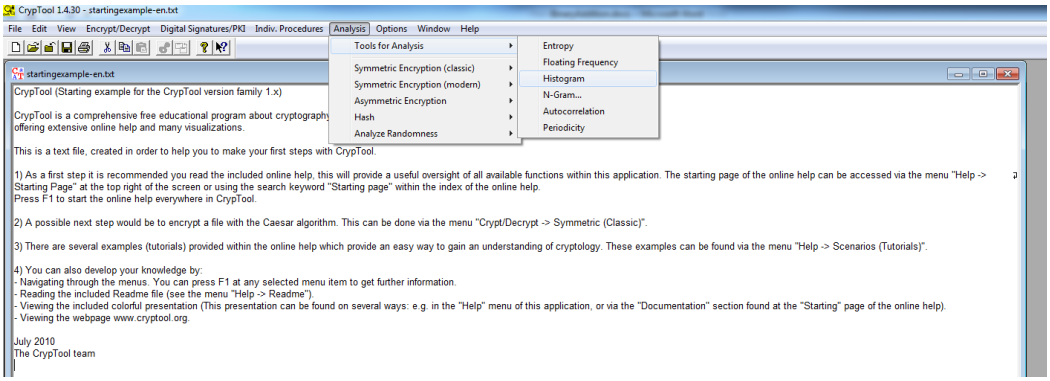
The key used for Binary Addition is entered in the Key entry dialog.

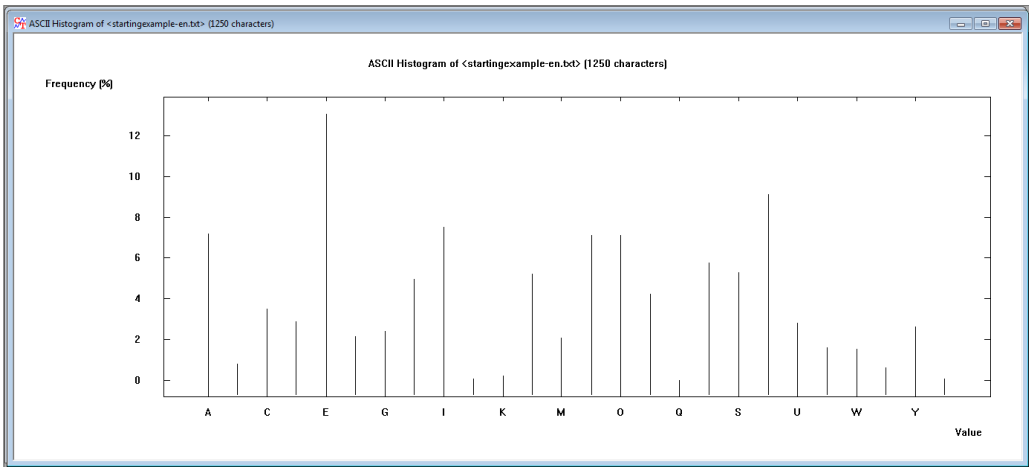
This encryption algorithm can be easily broken with a Ciphertext-Only attack (see Automatic analysis, Byte Addition). An example of this will be found in the Examples chapter.

1. Open the file **CrypTool-en.txt** under **C:\Program Files (x86)\CrypTool\examples**.



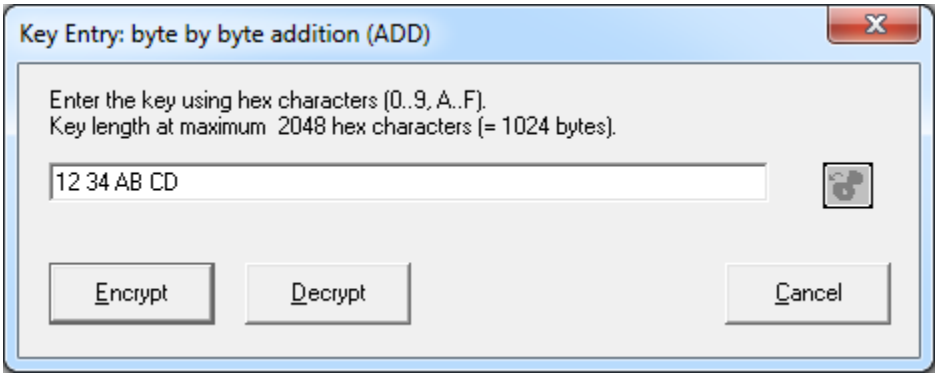
2. Click "**Analysis\Tools for Analysis\Histogram**".





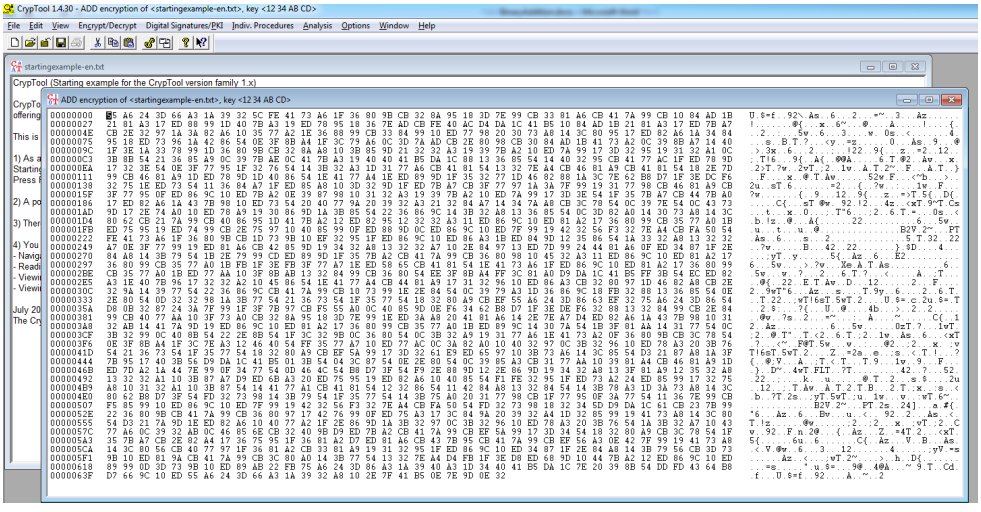
We can see from the histogram that the character which occurs most frequently is the letter E. This is true of many German and English texts. This information will be used later on during our attack.

3. Close the histogram dialog. Choose from menu "**Encrypt/Decrypt\Symmetric\Byte Addition**".

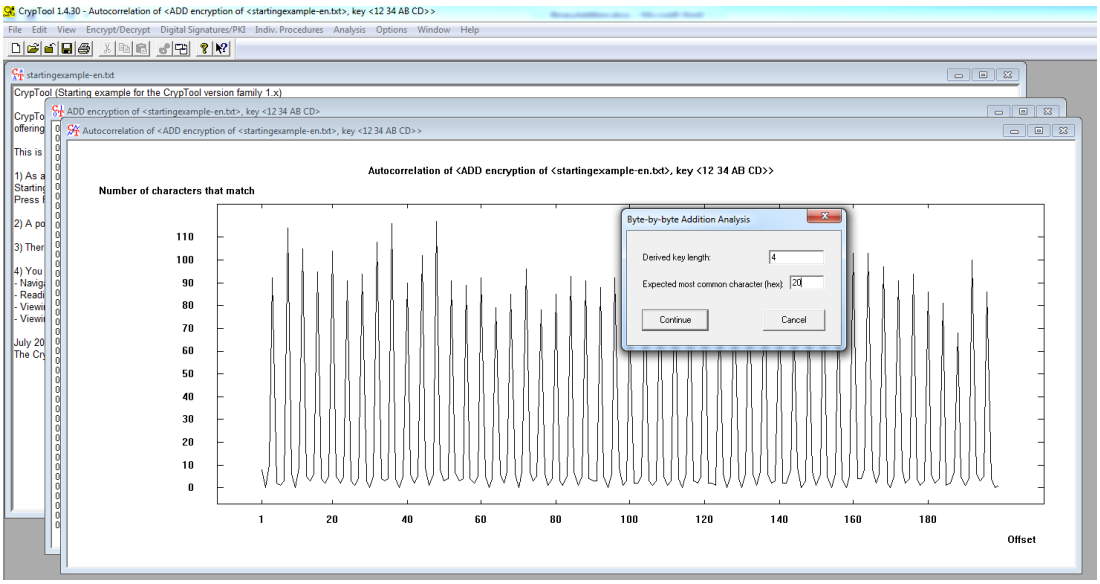


4. Enter **12 34 AB CD** as the key and click **Encrypt**.

The encrypted message shows up:

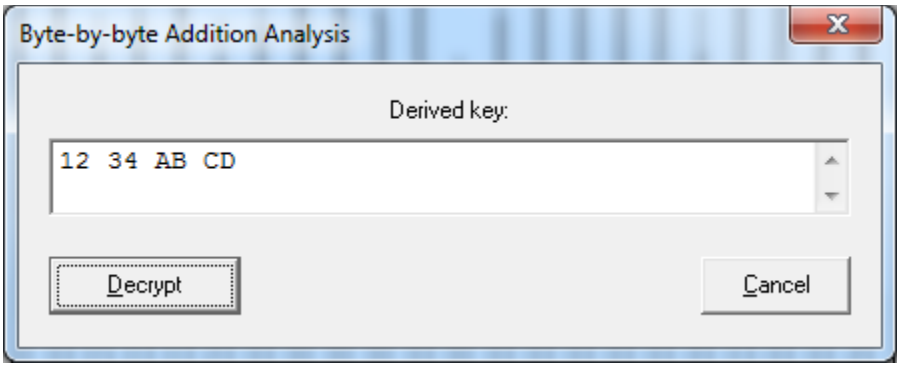


5. cipher text only attack will be performed. Choose from menu "**Analysis\Symmetric\Ciphertext-only\Byte Addition**".

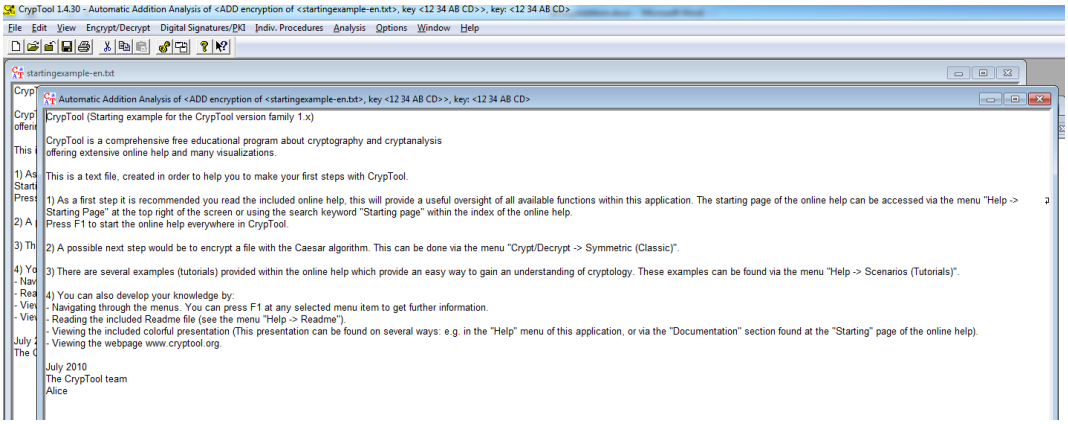


We are told that key length is calculated to be 4. The commonest character is E with hexadecimal value of 45. If we look at the plaintext, the most frequently character is e with hexadecimal value of 65. We enter into the Expected most common character field in the Byte-by-byte Addition Analysis box 20 (=65-45).

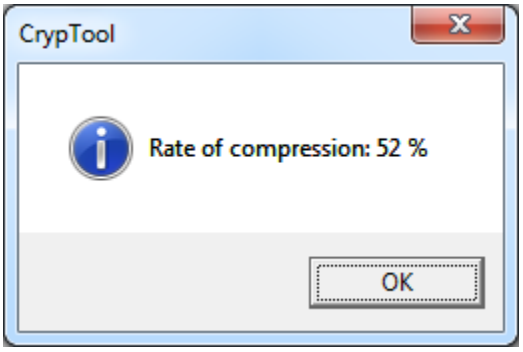
6. Click "**Continue**", CrypTool has been able to find the key. The only information was needed to do this was the fact that the character which occurred most frequently in the plaintext was the lower case letter e.



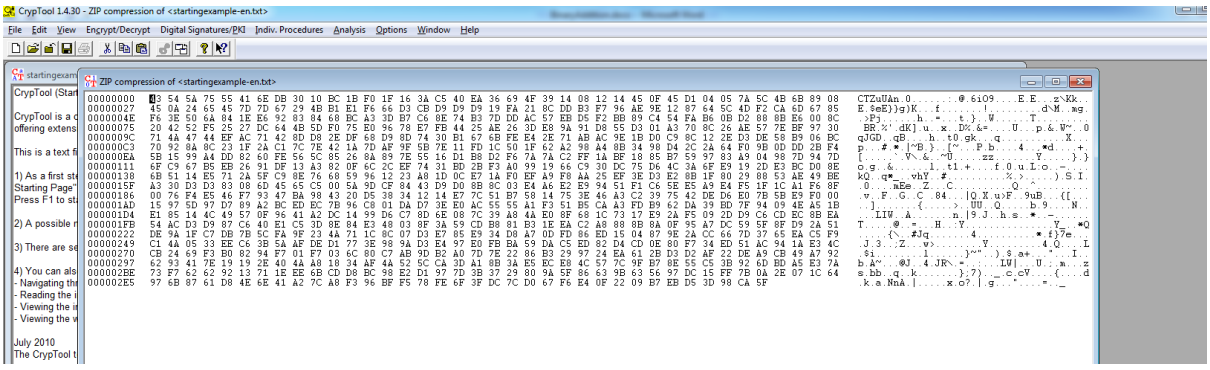
7. Click the "**Decrypt**" button shows the plaintext.



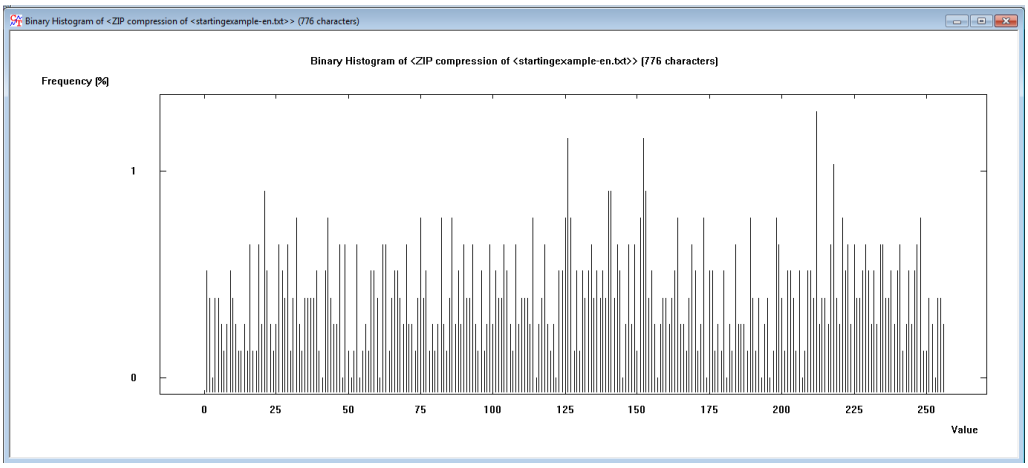
8. If the text is compressed prior to encryption then we will not be able to draw any conclusions from the frequency distribution of the characters in the text about the frequency distribution of the compressed text, since the compression process not only reduces size of a file but alters the frequencies of the individual characters so that they no longer reflect the frequencies of the characters in the original text. To compress the document, we make startingexample-en.txt active again. And select "**Indiv. Procedure\Tools\Compress\Zip**", the rate of compression is displayed.



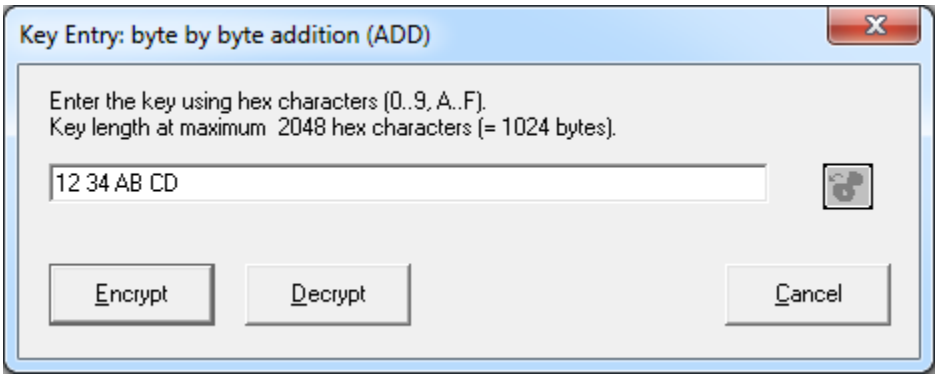
9. Click "**OK**", the compressed document is shown.



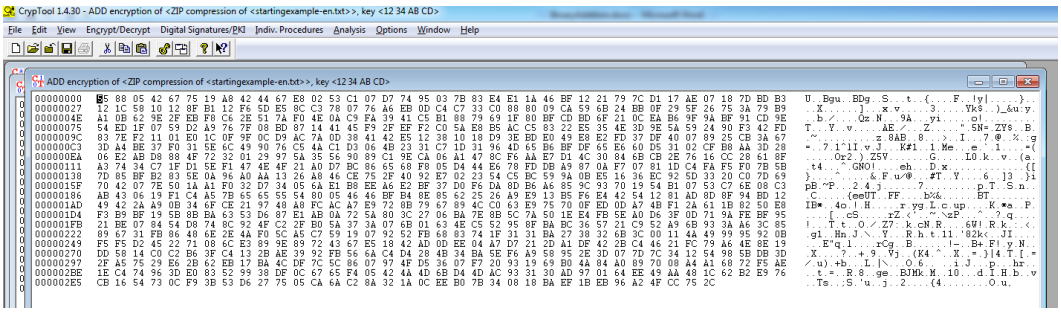
10. Click "**Analysis\Tools for Analysis\Histogram**" to see its histogram. The compression produces a quite different histogram profile from the one previously obtained for the uncompressed document. The characters are much more evenly distributed than in the unencrypted document.



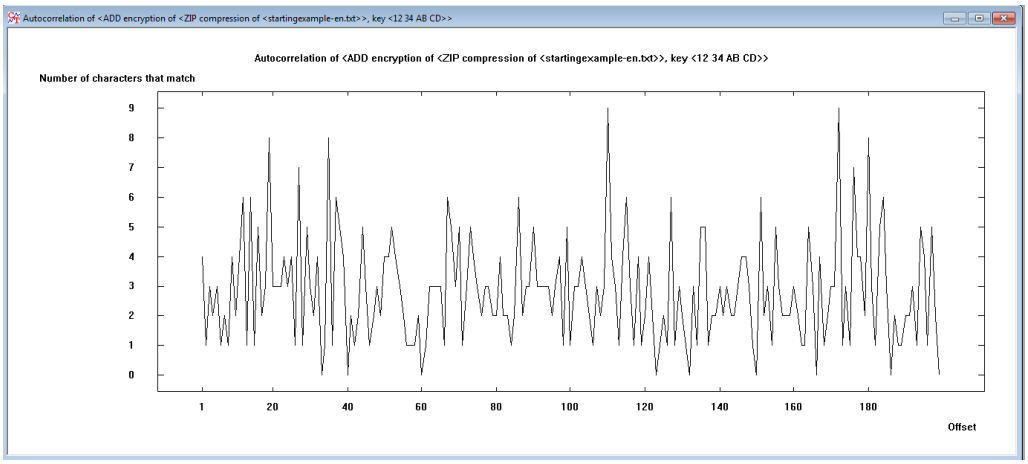
11. Make the compressed document the active window once again and the encrypt it using the same key **12 34 AB CD**.



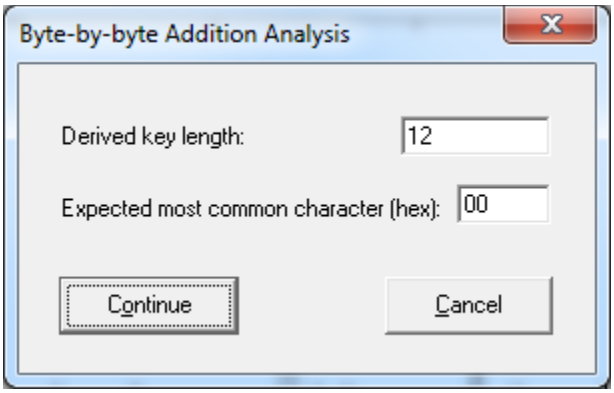
12. Click "**Encrypt**".



13. We invoke the analysis again by choosing from menu "**Analysis\Symmetric\Ciphertext-only\Byte Addition**".



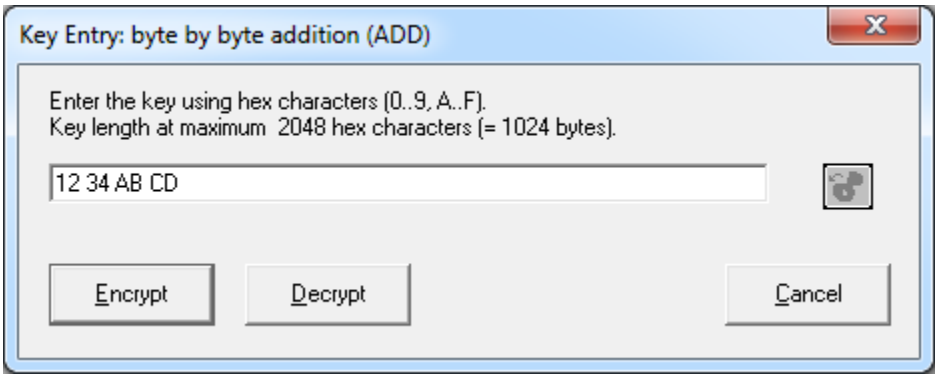
CrypTool returns an incorrect key length of 12.



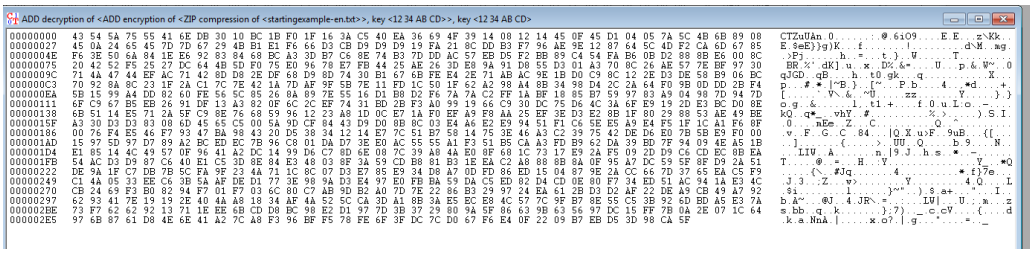
Given this key length, it is not possible to find the correct key either.

14. We will check whether it is possible to arrive at a readable version of the text document from the compressed and then encrypted document. We will provide the key and then unzip.

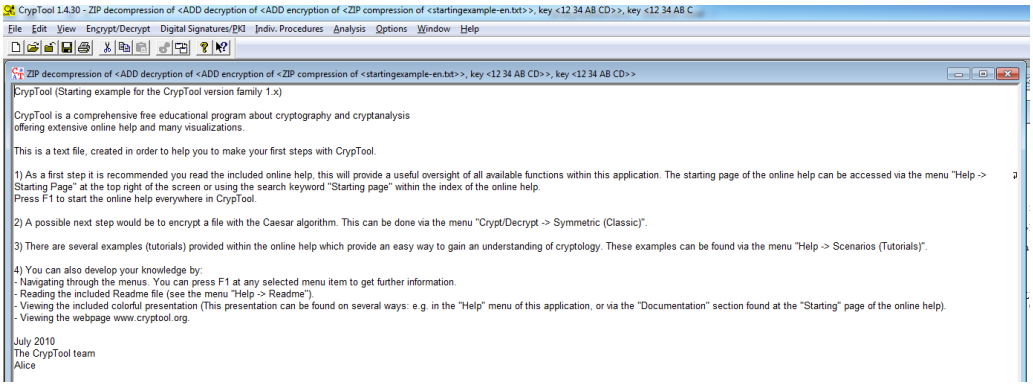
We will make the compressed and encrypted document the active window again. Choose from menu "**Encrypt/Decrypt\Symmetric\Byte Addition**".



15. Enter **12 34 AB CD** as the key and click **Decrypt**.

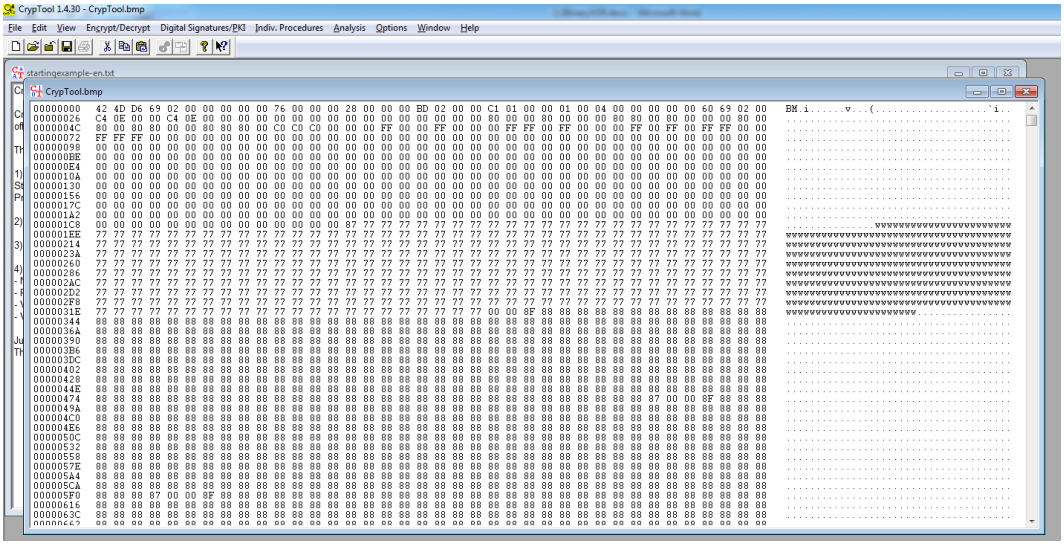


16. Choose from menu "**Indiv. Procedure\Tools\Compress\UnZip**", and the original text is displayed.

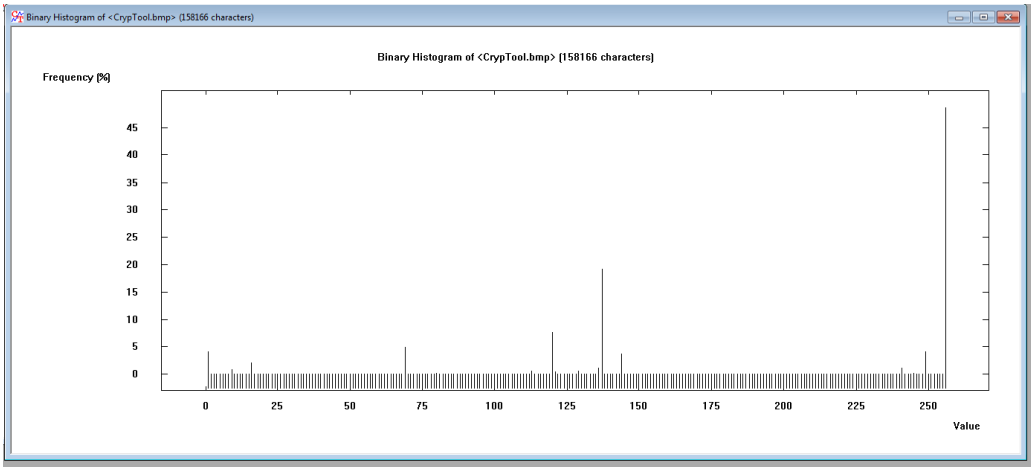


Task 3: Encryption using binary Exclusive-OR (XOR) (practice)

1. Open file CrypTool.bmp from "**C:\Program Files (x86)\CrypTool\examples**".

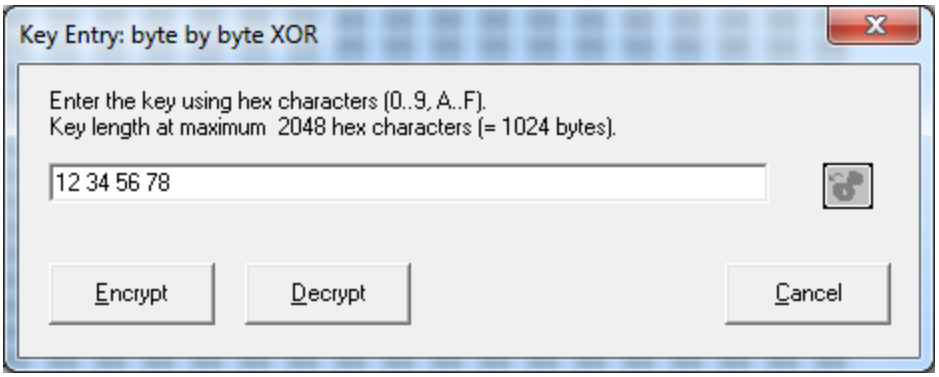


2. Look at the frequency distribution of the characters by clicking "**Analysis\Tools for Analysis \ Histogram**".



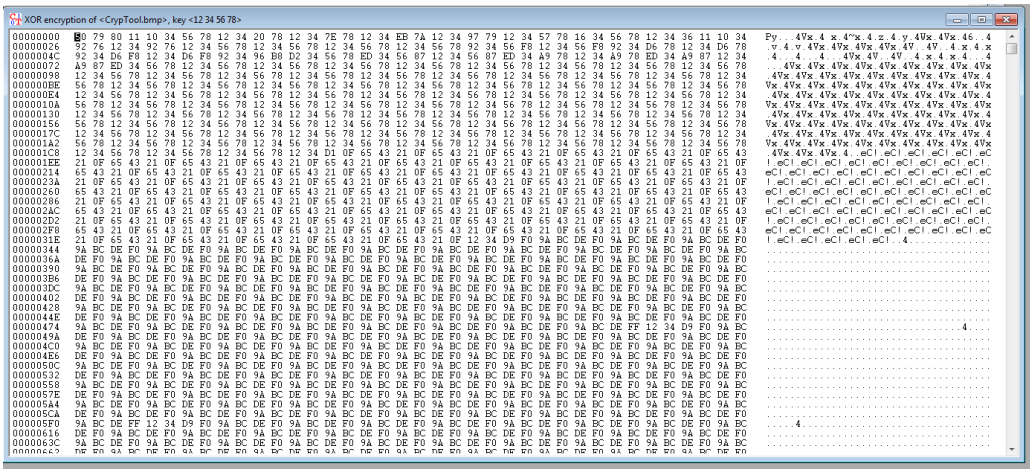
You can see from the histogram that the character which occurs most frequently has the value **255**. In hexadecimal notation this corresponds to **FF**. This information will be used later on during our attack.

3. Click on the window of "**CrypTool.bmp**". And click "**Encrypt\Decrypt/Symmetric/XOR**" from menu.

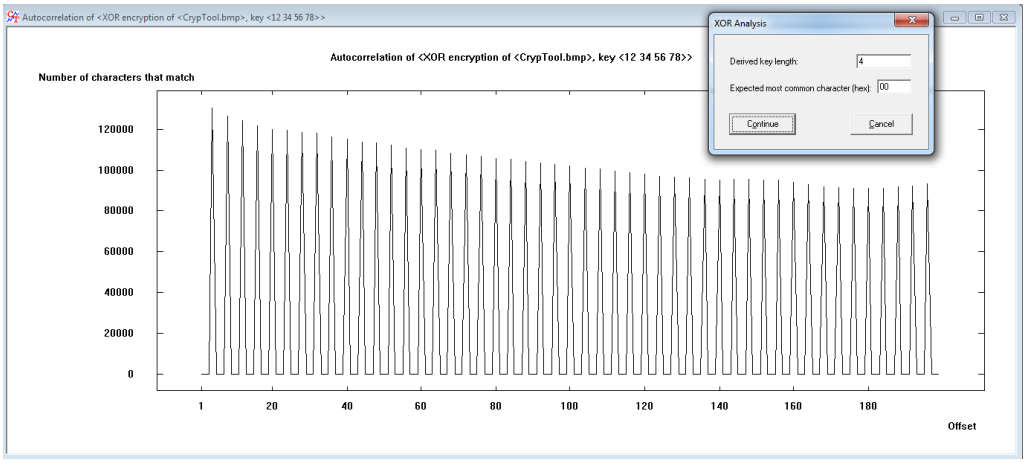


4. Enter **12 34 56 78** as the key.

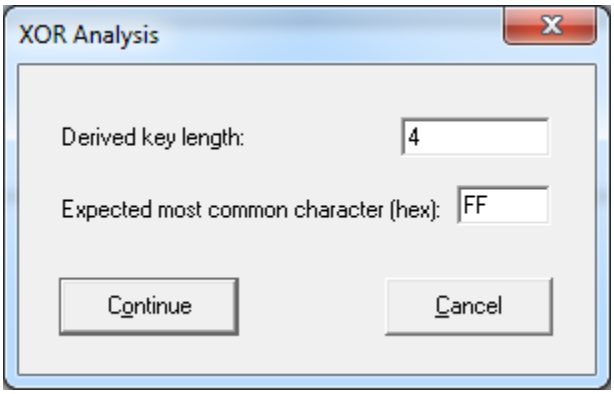
5. Click "**Encrypt**"



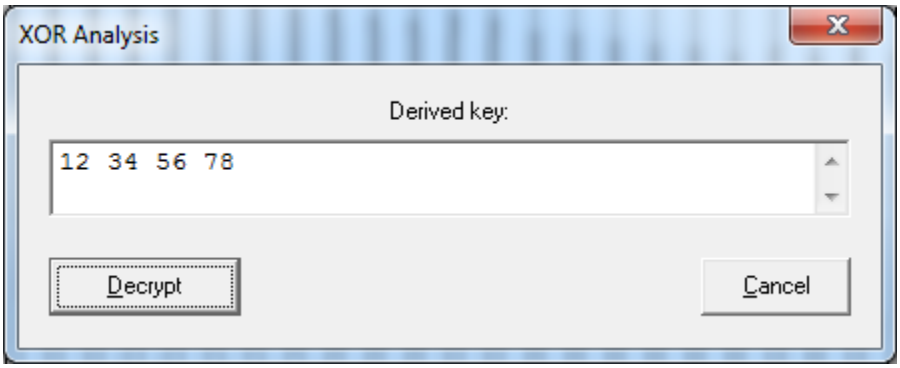
6. We will perform the cipher-text only attack. Select "**Analysis\Symmetric Encryption\Ciphertext-Only\XOR**".



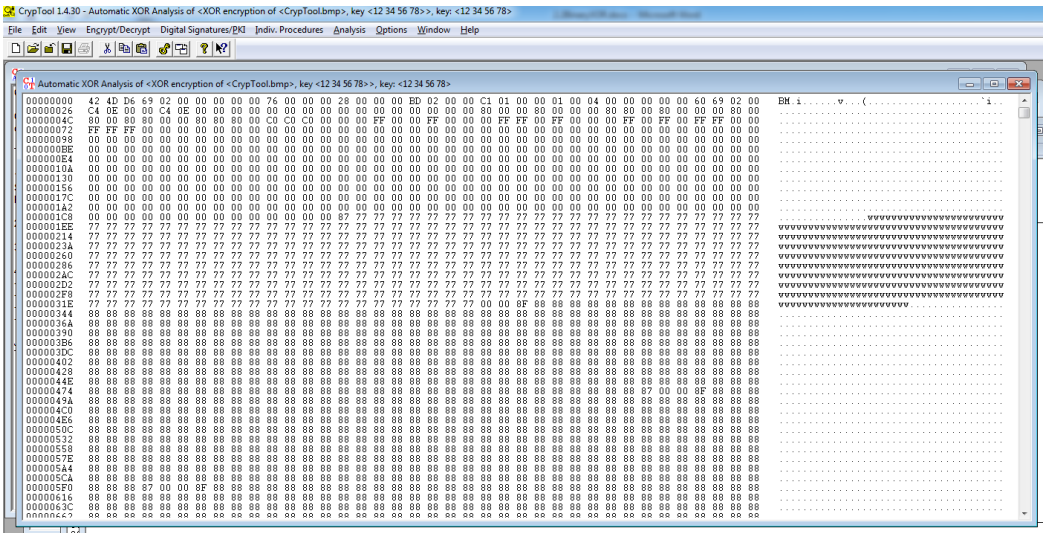
The autocorrelation is calculated and displayed. We are told that the key length is calculated to be 4. As we have seen in step 2, the most commonest character is FF. This we enter in the **Expected most common character field**.



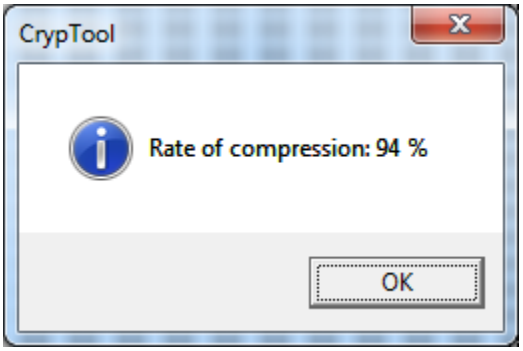
7. Click "**Continue**".



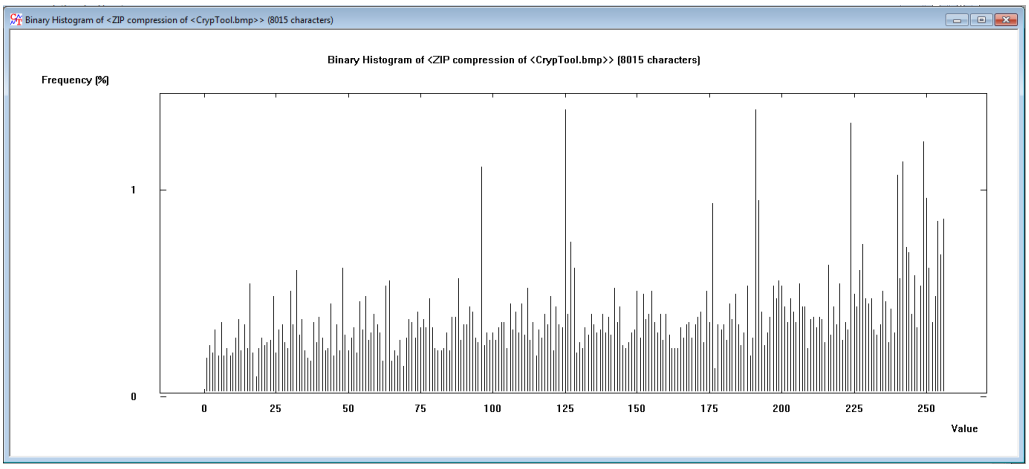
8. Click "**Decrypt**".



9. If we compress the document before encryption. By clicking "**Indiv. Procedure\Tools\Compress\Zip**".



10. Select "**Analysis\Tools for Analysis \ Histogram**", which produces a quite different histogram from the one previously obtained for the uncompressed picture in bitmap format.



11. Encrypt the compressed document by selecting "**Encrypt\Decrypt/Symmetric/XOR**" from menu and use **12 34 56 78** as the key.

12. We will perform the analysis. Select "**Analysis\Symmetric Encryption\Ciphertext-Only\XOR**". CrypTool returns incorrect key length.