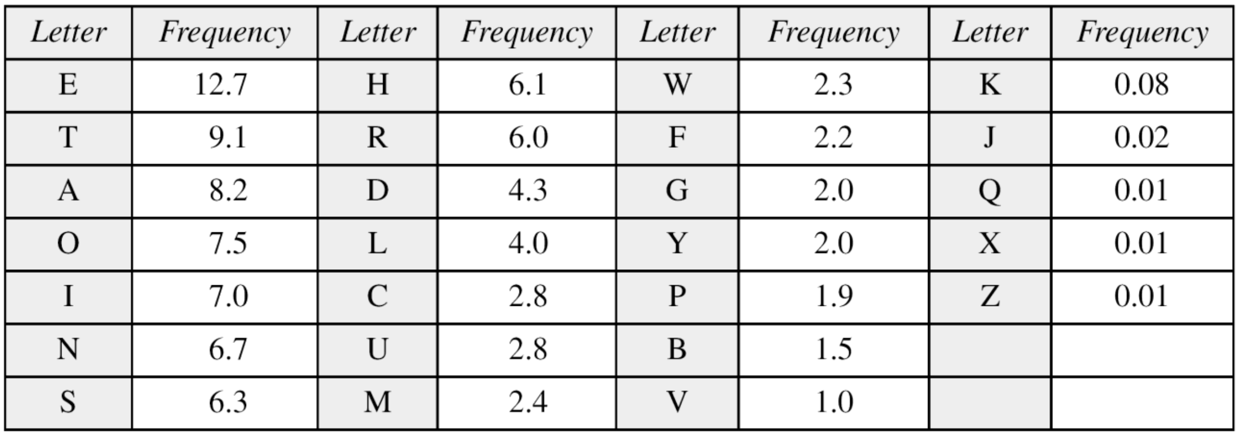
Lab on Frequencey Analysis

Overview

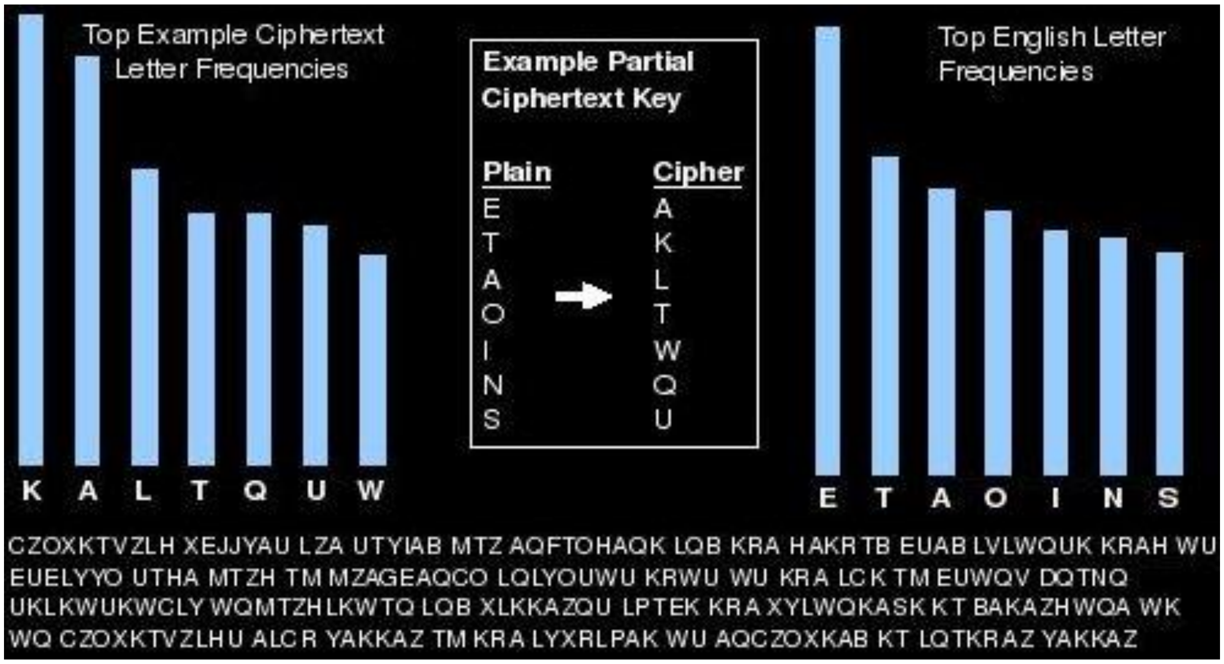
The cryptanalyst can benefit from some inherent characteristics of the plaintext language to launch a statistical attack. For example, we know that the letter E is the most frequently used letter in English text. The cryptanalyst finds the mostly-used character in the ciphertext and assumes that the 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 resonable 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 ciphertext 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.



CZOXKTVZLH XEJJYAU LZA UTYIAB MTZ AQFTOHAQK LQB KRA HAKRTB EUAB LVLWQUK KRAH WU EUELYYO UTHA MTZH TM MZAGEAQCO LQLYOUWU KRWU WU KRA LCK TM EUWQV DQTNQ UKLKWUKWCLY WQMTZHLKWTQ LQB XLKKAZQU LPTEK KRA XYLWQKASK KT BAKAZHWQA WK WQ CZOXKTVZLHU ALCR YAKKAZ TM KRA LYXRLPAK WU AQCZOXKAB KT LQTKRAZ YAKKAZ

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. You will be given an almost fully working C# .NET application (contained in CPSC4600-Lab1.zip). To get this application fully working, you will need to implement the empty methods. After these methods are complete, the program can then be used to complete the remainder of the lab. You do not need to change any of the UI code to get this working, only methods in the Encryption.cs class.

Getting Started

-Open up Visual Studio 2008. (If you do not have a copy for your own computer, it is available through the Microsoft Academic Alliance Program as well as Microsoft's Dreamspark web site)

-Unzip the CPSC-4600-Lab1.zip

-Open up the .sln file with Visual Studio 2008

-The project's contents will be listed on the right-hand side of the IDE.

-MainForm.cs is the UI code that can be left alone (if you would like to tinker with it, you may want to work on a copy)

-**StatisticalAnalysis.cs** contains the methods you will need to implement in order to finish the lab. C# is very much like Java, if you have any questions about the language MSDN is a great resource (http://msdn.microsoft.com/en-us/vcsharp/aa336809.aspx)

Fill In The Code…

Read the descriptions and hints carefully and fill in the missing methods in StatisticalAnalysis.cs.

```
        // Pre-conditions:   a class char[] called Transformation exists
        //                   in the class; the value p is an input parameter
        // Post-conditions:  the contents of the class char[] called
        //                   Transformation has been shifted by p characters
        //                   (make sure it wraps around!)
        //                   HINT: the modulus operator is %
        public void ShiftTransformationArray(int p)

        // Pre-conditions:   a static char[] called Alphabet of length 26
        //                  containing the alphabet (in UPPER CASE!!),
        //                  and the string inputStr is an input parameter
        // Post-conditions:  an int[] of length 26 is calculated, where
        //                  each value in the integer array is the
        //                  number of occurrences (the frequency)
        //                  the corresponding letter occurred in InputStr
        public static int[] DetermineLetterFrequencies(String InputStr)



        // Pre-conditions:  character p is the character to be found arr1
        //                  arr1 is the array to search through
        //                  arr2 is the array to return a character from
        // Post-conditions: Search through arr1 to find the character p, and
        //                  return the corresponding character in arr2
        private char FindInCorrArray(char p, ref char[] arr1, ref char[]
arr2)
```

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

To Turn in on BlackBoard:

A zip file named lastname_firstname_cpsc4600lab1.zip, containing:

- StatisticalAnalysis.cs
- 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