

# Project 1

Cipher Cracker



Samuel Carrasco

## Overview:

Cipher Analyzer is my program that helps facilitate the cracking of the cipher text files. The program begins by determining if the cipher is monoalphabetic or polyalphabetic via frequency analysis and index of coincidence. Once the cipher type is determined there is a sequence of steps taken to crack the cipher text.

## Contents

<b>1</b>	<b>Running the Program</b>	<b>2</b>
<b>2</b>	<b>Cipher Text Input</b>	<b>2</b>
2.1	Input Via User Input	2
2.2	Input Via Text File	3
2.3	Input Chosen	3
<b>3</b>	<b>Cracking the Cipher Text</b>	<b>3</b>
3.1	Determine The Category	3
3.1.1	Frequency Analysis	4
3.1.2	Index of Coincidence	4
3.2	Monoalphabetic Cipher	5
3.2.1	Shift Cipher	5
3.2.2	Substitution Cipher	5
3.2.3	Transposition Cipher	6
3.3	Polyalphabetic Cipher	6
3.3.1	Vigenere Cipher	6
3.3.2	One-Time Pad	7
<b>4</b>	<b>cipher1.txt</b>	<b>7</b>
4.1	Cipher Type	7
4.2	Key	7
4.3	Plain Text	7
<b>5</b>	<b>cipher2.txt</b>	<b>7</b>
5.1	Cipher Type	7
5.2	Key	8
5.3	Plain Text	8
<b>6</b>	<b>cipher3.txt</b>	<b>8</b>
6.1	Cipher Type	8
6.2	Keyword	8
6.3	Plain Text	8
<b>7</b>	<b>cipher4.txt</b>	<b>8</b>
7.1	Cipher Type	8
7.2	Keyword	9
7.3	Plain Text	9
<b>8</b>	<b>Conclusions</b>	<b>9</b>

## 1 Running the Program

- Import the project as a Maven project from your IDE of Choice. (**Import Via Eclipse**)
- Run the executable jar file using the command **java -jar Project1.jar**

## 2 Cipher Text Input

The program begins by having the user input the cipher text via user input or by selecting a text file.  
(Note: The programs makes the assumption that the cipher text is all uppercase English letters and that the cipher text contains **no spaces**.)

### 2.1 Input Via User Input

The user has the ability to input their own cipher text.

```
Welcome to the Input Section.  
1 -- Input your own cipher text.  
2 -- Chose a text file.  
3 -- Leave program  
  
Select an option:  
1  
Input your cipher text below:  
DEEMTYESDFFGAE|
```

Figure 1: User Input Section

## 2.2 Input Via Text File

The user has the ability to browse the resource folder to select a text file.

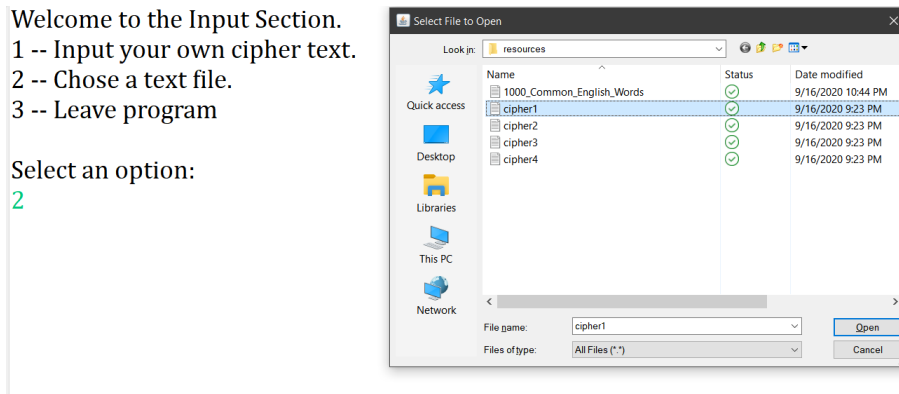


Figure 2: Text File Input Section

## 2.3 Input Chosen

Once the input source has been chosen, the program then reads back the input that was inputted.

```

The following cipher text has been inputted:
KTGPTCTQCWTQXGPTHFRGTYDGZGTDRFQUYTUGQXQUEZEQUTCTUXTAGHFRQFTDGMWRDPSPHDGRATRFDHUTY
Welcome to Cipher Analyzer.
1 -- Run frequency analysis of the cipher text.
2 -- Calculate index of coincidence
3 -- Go to shift cipher tools
4 -- Go to substitution cipher tools
5 -- Go to columnar transposition cipher tools
6 -- Go to Vigenere cipher tools
7 -- Leave program

Select an option:|
  
```

Figure 3: Final Input Section

# 3 Cracking the Cipher Text

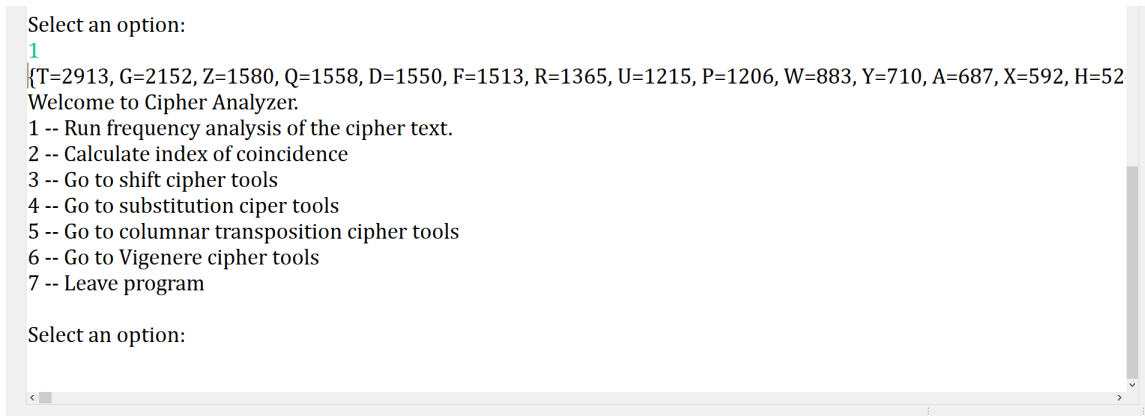
In order to break the cipher text the program has tools to help determine which encryption scheme the cipher text was encoded with. Following these steps helps reduce the time to crack the cipher text.

## 3.1 Determine The Category

Determine whether the cipher text was encrypted using a monoalphabetic cipher or a polyalphabetic cipher. This step is the most crucial because it will guided you to which tools to use next in the program.

### 3.1.1 Frequency Analysis

The first tool the program provides is a sorted frequency analysis of each character in the cipher text. Frequency analysis is a powerful tool to help determine the category by comparing the frequency of each cipher text letter. If the cipher text contains **highly skewed** frequencies of the letters this *hints* that the cipher used is most likely **monoalphabetic** cipher. Likewise, if the cipher text contains a **somewhat uniform** frequencies of the letters this *hints* that the cipher used is most likely **polyalphabetic** cipher.



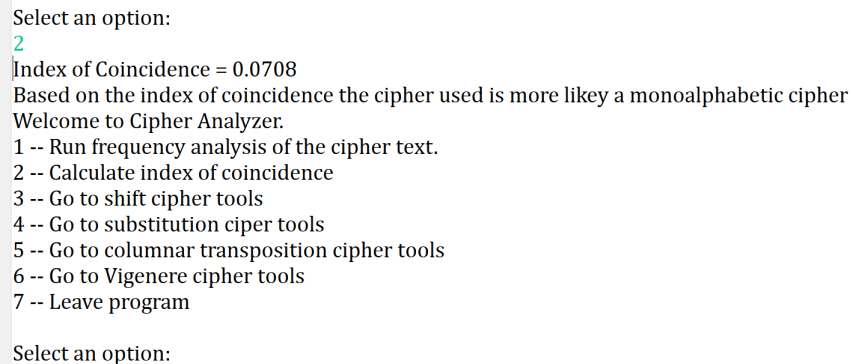
```
Select an option:
1
{T=2913, G=2152, Z=1580, Q=1558, D=1550, F=1513, R=1365, U=1215, P=1206, W=883, Y=710, A=687, X=592, H=52
Welcome to Cipher Analyzer.
1 -- Run frequency analysis of the cipher text.
2 -- Calculate index of coincidence
3 -- Go to shift cipher tools
4 -- Go to substitution cipher tools
5 -- Go to columnar transposition cipher tools
6 -- Go to Vigenere cipher tools
7 -- Leave program

Select an option:
```

Figure 4: Frequency Analysis Section

### 3.1.2 Index of Coincidence

The next tool the program provides is a index of coincidence of the cipher text. The index of coincidence (IC) is a measure of how likely it is to draw two matching letters by randomly selecting two letters from a given text. Having a IC value that is close to .067 reflects that the cipher used is more likely to be a **monoalphabetic** cipher due to nature of the English language. Based on the IC of the cipher text, the program will give a recommendation whether the cipher is a monoalphabetic or polyalphabetic cipher.



```
Select an option:
2
Index of Coincidence = 0.0708
Based on the index of coincidence the cipher used is more likely a monoalphabetic cipher
Welcome to Cipher Analyzer.
1 -- Run frequency analysis of the cipher text.
2 -- Calculate index of coincidence
3 -- Go to shift cipher tools
4 -- Go to substitution cipher tools
5 -- Go to columnar transposition cipher tools
6 -- Go to Vigenere cipher tools
7 -- Leave program

Select an option:
```

Figure 5: Index of Coincidence Section

## 3.2 Monoalphabetic Cipher

The program has tools for the following monoalphabetic ciphers: Shift, Substitution and Transposition. Once the cipher text has been deemed to be a monoalphabetic cipher follow this list in order:

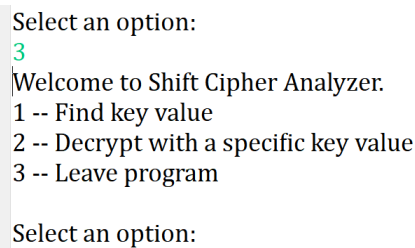
1. Try the shift cipher tools
2. Try the substitution cipher tools
3. Try the transposition cipher tools.

### 3.2.1 Shift Cipher

The shift cipher has the following tools to help crack the cipher text:

- Find the key value
- Decrypt with a key value.

Finding the key feature works by trying all the key values 0-25 on the cipher text and seeing if the plain text contains the most common English words. Once the key value has been found simply use the decrypt feature and see if the plain text is *readable*.



```
Select an option:  
3  
Welcome to Shift Cipher Analyzer.  
1 -- Find key value  
2 -- Decrypt with a specific key value  
3 -- Leave program  
Select an option:
```

Figure 6: Shift Cipher Section

### 3.2.2 Substitution Cipher

The substitution cipher has the following tools to help crack the cipher text:

- Sorted frequency analysis
- Replacement of the cipher text.
- Print the cipher text.

Begin by running a sorted frequency analysis of cipher text letters. Then using the replacement feature and this frequency analyst **table** to start decoding the cipher text. Repeat this process until the cipher text is able to be decoded.

```
Select an option:
4
Welcome to Substitution Cipher Analyzer.
1 -- Run sorted frequency analysis of the cipher text.
2 -- Replace a ciphertext letter to a plaintext letter
3 -- Print out the cipher text
4 -- Reset the cipher text
5 -- Leave program

Select an option:
```

Figure 7: Substitution Cipher Section

### 3.2.3 Transposition Cipher

The transposition cipher has the following tools to help crack the cipher:

- Decrypt using a key value

Begin by enumerating all short keywords and decoding with the decrypt feature. The short keywords can be found [here](#).

```
Select an option:
5
Welcome to Columnar Transposition Cipher Analyzer.
1 -- Decrypt with a specific key value
2 -- Leave program

Select an option:
```

Figure 8: Transposition Cipher Section

## 3.3 Polyalphabetic Cipher

The program has the following polyalphabetic cipher: Vigenere.

### 3.3.1 Vigenere Cipher

The Vigenere cipher has the following tools to help crack the cipher text:

- Find key length
- Find key values
- Decrypt using a key.

Begin by running the find the key length feature. This feature works by performing a shifted index of coincidence (**SIC**) on the cipher text. If the shift SIC is high enough the program will give potential key lengths. Then once the key length is known run the find key values feature. This feature works by doing a statistical analysis of the frequency of the letters. Chose the letter that **maximizes** the value for each key position. Final once the key is known, decrypt using the decrypt feature.

```
Select an option:
6
Welcome to Vigenere Cipher Analyzer:
1 -- Find the key length using a shifted index of coincidence
2 -- Decrypt with a specific key value
3 -- Break the cipher text into groups. Note first find the key length
4 -- Perform a frequency analysis to find the key
5 -- Leave program

Select an option:
```

Figure 9: Vigenere Cipher Section

### 3.3.2 One-Time Pad

If you have exhausted all of the above cipher tools it is highly likely the cipher text is encrypted using a one-time pad. The one-time pad (OTP) is an encryption technique that cannot be cracked without the knowledge of the key.

## 4 cipher1.txt

### 4.1 Cipher Type

Based on the frequency analysis and index of coincidence the cipher is highly likely to be monoalphabetic. Moreover the cipher used was a shift cipher.

Shift Cipher

### 4.2 Key

After running the find key feature it gave 10 and 16 as possible key values. Then running the decrypt feature with key 16 gave a plain text that is readable while the key value of 10 was unreadable.

Key = 16

### 4.3 Plain Text

- See plaintext1.txt in the zipfile.

## 5 cipher2.txt

### 5.1 Cipher Type

Based on the frequency analysis and index of coincidence the cipher is highly likely to be monoalphabetic. Moreover the cipher used was a substitution cipher.

Substitution Cipher



## 5.2 Key

After doing careful substitution and replace here is the substitution table.

a	b	c	d	e	f	g	h	i	j	k	l	m
Z	M	A	Y	T	X	L	P	R	S	B	W	E

n	o	p	q	r	s	t	u	v	w	x	y	z
F	Q	C	I	U	D	G	H	J	K	N	O	V

## 5.3 Plain Text

- see plaintext2.txt

# 6 cipher3.txt

## 6.1 Cipher Type

Based on the frequency analysis and index of coincidence the cipher is highly likely to be polyalphabetic. Moreover the cipher used was a Vigenere cipher.

Vigenere Cipher

## 6.2 Keyword

After running the find key length feature it gave a key length of 6. Then running the find key values feature it gave the keyword of TUCSON.

Keyword = TUCSON

## 6.3 Plain Text

- See plaintext3.txt in the zipfile.

# 7 cipher4.txt

## 7.1 Cipher Type

Based on the frequency analysis and index of coincidence the cipher is highly likely to be polyalphabetic. Moreover the cipher used is suspected to be a one-time pad since the Vigenere tools failed to give a key.

One-Time Pad

## 7.2 Keyword

Key = ?UNKNOWN?

## 7.3 Plain Text

- Plain text not able to be decoded.

## 8 Conclusions

By only having the cipher text requires much more computational power and critical thinking to crack the ciphers. In addition, by having longer cipher texts allows the statistical methods to be more powerful and to give better predictions. Another interesting observation is when you run the shifted index of coincidence for a monoalphabetic cipher the key length is one. Likewise for the Vigenere cipher all multiples of the key length will give a high shifted index of coincidence. Furthermore, I learn how the nature of languages can be exploited to crack a cipher text. Finally the substitution cipher requires the most manual labor due to the process of trial and error; however thankfully the cipher the text was long so that the frequencies reached equilibrium.