Project On Shift, Affine, Quad Cipher

Shift Cipher Project

Two Programs You Need to Write

All of the programs below take a LARGE text from a file.

Two Programs You Need to Write

All of the programs below take a LARGE text from a file.
1) Input a large text of letters. Make all letters small, remove all punctuation, and break the text into blocks of five. For example

Two Programs You Need to Write

All of the programs below take a LARGE text from a file.
1) Input a large text of letters. Make all letters small, remove all punctuation, and break the text into blocks of five. For example
Bill works at a zoo
becomes

Two Programs You Need to Write

All of the programs below take a LARGE text from a file.
1) Input a large text of letters. Make all letters small, remove all punctuation, and break the text into blocks of five. For example
Bill works at a zoo
becomes
billw orksa tazoo

Two Programs You Need to Write

All of the programs below take a LARGE text from a file.
1) Input a large text of letters. Make all letters small, remove all punctuation, and break the text into blocks of five. For example
Bill works at a zoo
becomes
billw orksa tazoo

2) Input a large text of letters, all small letters, in blocks of 5. Output the letters all replaced by numbers: *a* becomes 0, *b* becomes 1, etc. For example

Two Programs You Need to Write

All of the programs below take a LARGE text from a file.
1) Input a large text of letters. Make all letters small, remove all punctuation, and break the text into blocks of five. For example
Bill works at a zoo
becomes
billw orksa tazoo

2) Input a large text of letters, all small letters, in blocks of 5. Output the letters all replaced by numbers: *a* becomes 0, *b* becomes 1, etc. For example
billw orksa tazoo
becomes

Two Programs You Need to Write

All of the programs below take a LARGE text from a file.

1) Input a large text of letters. Make all letters small, remove all punctuation, and break the text into blocks of five. For example

Bill works at a zoo

becomes

billw orksa tazoo

2) Input a large text of letters, all small letters, in blocks of 5. Output the letters all replaced by numbers: *a* becomes 0, *b* becomes 1, etc. For example

billw orksa tazoo

becomes

1-8-11-11-22 14-17-10-18-0 19-0-25-14-14

To avoid confusing 21 with 2 1 I used dashes above. You can use a different mechanism.

Three More Programs You Need to Write

Three More Programs You Need to Write

3) Given a large text of numbers between 0 and 25, in blocks of 5, AND a shift $s \in \{0, \ldots, 25\}$, output the text shifted by s, mod 26. For example,

Three More Programs You Need to Write

3) Given a large text of numbers between 0 and 25, in blocks of 5, AND a shift $s \in \{0, \ldots, 25\}$, output the text shifted by s, mod 26. For example,

1-8-11-11-22 14-17-10-18-0 19-0-25-14-14; 2

becomes

Three More Programs You Need to Write

3) Given a large text of numbers between 0 and 25, in blocks of 5, AND a shift $s \in \{0, \ldots, 25\}$, output the text shifted by s, mod 26. For example,

1-8-11-11-22 14-17-10-18-0 19-0-25-14-14; 2

becomes

3-10-13-13-24 16-19-12-20-2 21-2-1-16-16

Three More Programs You Need to Write

3) Given a large text of numbers between 0 and 25, in blocks of 5, AND a shift $s \in \{0, \ldots, 25\}$, output the text shifted by s, mod 26. For example,

1-8-11-11-22 14-17-10-18-0 19-0-25-14-14; 2

becomes

3-10-13-13-24 16-19-12-20-2 21-2-1-16-16

4) Given a large text of numbers between 0 and 25, in blocks of 5. Output the text in English by using 0 goes to a, 1 goes to b, etc. Put it into blocks of 5. For example

Three More Programs You Need to Write

3) Given a large text of numbers between 0 and 25, in blocks of 5, AND a shift $s \in \{0, \ldots, 25\}$, output the text shifted by s, mod 26. For example,

1-8-11-11-22 14-17-10-18-0 19-0-25-14-14; 2

becomes

3-10-13-13-24 16-19-12-20-2 21-2-1-16-16

4) Given a large text of numbers between 0 and 25, in blocks of 5. Output the text in English by using 0 goes to a, 1 goes to b, etc. Put it into blocks of 5. For example

3-10-13-13-24 16-19-12-20-2 21-2-1-16-16

becomes

Three More Programs You Need to Write

3) Given a large text of numbers between 0 and 25, in blocks of 5, AND a shift $s \in \{0, \ldots, 25\}$, output the text shifted by s, mod 26. For example,

1-8-11-11-22 14-17-10-18-0 19-0-25-14-14; 2

becomes

3-10-13-13-24 16-19-12-20-2 21-2-1-16-16

4) Given a large text of numbers between 0 and 25, in blocks of 5. Output the text in English by using 0 goes to a, 1 goes to b, etc. Put it into blocks of 5. For example

3-10-13-13-24 16-19-12-20-2 21-2-1-16-16

becomes

dknny qtmuc vcbqq

Three More Programs You Need to Write

3) Given a large text of numbers between 0 and 25, in blocks of 5, AND a shift $s \in \{0, \ldots, 25\}$, output the text shifted by s, mod 26. For example,

1-8-11-11-22 14-17-10-18-0 19-0-25-14-14; 2

becomes

3-10-13-13-24 16-19-12-20-2 21-2-1-16-16

4) Given a large text of numbers between 0 and 25, in blocks of 5. Output the text in English by using 0 goes to a, 1 goes to b, etc. Put it into blocks of 5. For example

3-10-13-13-24 16-19-12-20-2 21-2-1-16-16

becomes

dknny qtmuc vcbqq

5) Input a text of English and a shift $s \in \{0, \ldots, 25\}$. Output the result of shifting the text by s. This will just be combining the programs above.

Cracking the Shift Cipher

Is the Shift Cipher Secure?

There are only 26 possible keys, and 26 is small, so crackable.

Is the Shift Cipher Secure?

There are only 26 possible keys, and 26 is small, so crackable.

That is correct but incomplete.

Is the Shift Cipher Secure?

There are only 26 possible keys, and 26 is small, so crackable.

That is correct but incomplete.

Here is the algorithm that reasoning leads to

Is the Shift Cipher Secure?

There are only 26 possible keys, and 26 is small, so crackable.

That is correct but incomplete.

Here is the algorithm that reasoning leads to

1. Input T a text.

Is the Shift Cipher Secure?

There are only 26 possible keys, and 26 is small, so crackable.

That is correct but incomplete.

Here is the algorithm that reasoning leads to

1. Input T a text.
2. For $s = 0$ to 25 generate T_s (T shifted by s)

Is the Shift Cipher Secure?

There are only 26 possible keys, and 26 is small, so crackable.

That is correct but incomplete.

Here is the algorithm that reasoning leads to

1. Input T a text.
2. For $s = 0$ to 25 generate T_s (T shifted by s)
3. **Look** at each T_s. One will **look like** English.

Is the Shift Cipher Secure?

There are only 26 possible keys, and 26 is small, so crackable.

That is correct but incomplete.

Here is the algorithm that reasoning leads to

1. Input T a text.
2. For $s = 0$ to 25 generate T_s (T shifted by s)
3. **Look** at each T_s. One will **look like** English.

For the last step we need a **program** that can tell if a text **looks like English**

Freq Vectors

Let T be a long text. Length N. May or may not be coded.

Let N_a be the number of $a's$ in T.
Let N_b be the number of $b's$ in T.
$\vdots$

Freq Vectors

Let T be a long text. Length N. May or may not be coded.

Let N_a be the number of $a's$ in T.
Let N_b be the number of $b's$ in T.
$\vdots$

The **Freq Vector of T** is

$$\vec{f_T} = \left(\frac{N_a}{N}, \frac{N_b}{N}, \cdots, \frac{N_z}{N} \right)$$

Options for Is-English Program

Let $\vec{f_E}$ be Freq Vector for English.
Let $\vec{f_T}$ be Freq Vector for T.

Options for Is-English Program

Let $\vec{f_E}$ be Freq Vector for English.
Let $\vec{f_T}$ be Freq Vector for T.

How to tell if $\vec{f_T}$ is **close to** $\vec{f_E}$?

Options for Is-English Program

Let $\vec{f_E}$ be Freq Vector for English.
Let $\vec{f_T}$ be Freq Vector for T.

How to tell if $\vec{f_T}$ is **close to** $\vec{f_E}$?

- $\sum_{i=0}^{25} |f_{E,i} - f_{T,i}|$ If small then IS-ENGLISH. Need to find out how small.

Options for Is-English Program

Let $\vec{f_E}$ be Freq Vector for English.
Let $\vec{f_T}$ be Freq Vector for T.

How to tell if $\vec{f_T}$ is **close to** $\vec{f_E}$?

- $\sum_{i=0}^{25} |f_{E,i} - f_{T,i}|$ If small then IS-ENGLISH. Need to find out how small.
- $\sum_{i=0}^{25} (f_{E,i} - f_{T,i})^2$ If small then IS-ENGLISH. Need to find out how small.

Options for Is-English Program

Let $\vec{f_E}$ be Freq Vector for English.
Let $\vec{f_T}$ be Freq Vector for T.

How to tell if $\vec{f_T}$ is **close to** $\vec{f_E}$?

- $\sum_{i=0}^{25} |f_{E,i} - f_{T,i}|$ If small then IS-ENGLISH. Need to find out how small.
- $\sum_{i=0}^{25} (f_{E,i} - f_{T,i})^2$ If small then IS-ENGLISH. Need to find out how small.
- $\sum_{i=1}^{25} f_{E,i} f_{T,i}$ If large then IS-ENGLISH. Need to find out how large. This one is actually used.

How to Find Parameters

Let $f(T) = \sum_{i=0}^{25} |f_{E,i} - f_{T,i}|$
We want numbers α, β such that

- If T is English then $f(T) \geq \alpha$.
- If T is English SHIFTED by a non-zero number then $f(T) \leq \beta$.
- $\alpha - \beta$ is large.

How to find α, β?
Next slide.

Program to Find α, β, Method 1

Let T be a large English Text.
Let $f(T) = \sum_{i=0}^{25} |f_{E,i} - f_{T,i}|$

Program to Find α, β, Method 1

Let T be a large English Text.
Let $f(T) = \sum_{i=0}^{25} |f_{E,i} - f_{T,i}|$
1) Compute $\alpha = f(T)$.

Program to Find α, β, Method 1

Let T be a large English Text.
Let $f(T) = \sum_{i=0}^{25} |f_{E,i} - f_{T,i}|$

1) Compute $\alpha = f(T)$.

2) For $i = 1$ to 25
   Compute T_i, which is T shifted by i.
   Compute $\beta_i = f(T_i)$.

Program to Find α, β, Method 1

Let T be a large English Text.
Let $f(T) = \sum_{i=0}^{25} |f_{E,i} - f_{T,i}|$

1) Compute $\alpha = f(T)$.

2) For $i = 1$ to 25
   Compute T_i, which is T shifted by i.
   Compute $\beta_i = f(T_i)$.

3) Compute $\beta = \min\{\beta_1, \ldots, \beta_{25}\}$.

Program to Find α, β, Method 1

Let T be a large English Text.
Let $f(T) = \sum_{i=0}^{25} |f_{E,i} - f_{T,i}|$

1) Compute $\alpha = f(T)$.

2) For $i = 1$ to 25
   Compute T_i, which is T shifted by i.
   Compute $\beta_i = f(T_i)$.

3) Compute $\beta = \min\{\beta_1, \ldots, \beta_{25}\}$.

SO, are these the α, β you should use? You should do this for a few more texts and let α be the max of the α's, and β be the min of the β's.

Program to Find α, β, Method 2

Do the same with
$\sum_{i=0}^{25}(f_{E,i} - f_{T,i})^2$

Program to Find α, β, Method 3

Do the same with
$\sum_{i=1}^{25} f_{E,i} f_{T,i}$
BUT note here we want minimize α and maximize β.

Variants On a Theme

We removed ALL punctuation.
We ignored numbers and math symbols.

Variants On a Theme

We removed ALL punctuation.
We ignored numbers and math symbols.

You can try this on Math texts so include $\{0, \ldots, 9, +, \times\}$ and other math symbols.

Variants On a Theme

We removed ALL punctuation.
We ignored numbers and math symbols.

You can try this on Math texts so include $\{0, \ldots, 9, +, \times\}$ and other math symbols.
You can try this without removing punctuation.

Affine Cipher Project

Affine Cipher

Recall that Shift Cipher was $f(x) = x + s \pmod{26}$.
For all s this function is 1-1 which is needed for encryption.

Affine Cipher

Recall that Shift Cipher was $f(x) = x + s \pmod{26}$.
For all s this function is 1-1 which is needed for encryption.

We now try a more complicated encryption.
$f(x) = ax + b \pmod{26}$.
NOT all a, b work. Need a to be relatively prime to 26 (or whatever the size of the alphabet is.)

Affine Cipher

Recall that Shift Cipher was $f(x) = x + s \pmod{26}$.
For all s this function is 1-1 which is needed for encryption.

We now try a more complicated encryption.
$f(x) = ax + b \pmod{26}$.
NOT all a, b work. Need a to be relatively prime to 26 (or whatever the size of the alphabet is.)

For each program you wrote for the SHIFT cipher, write a similar one for AFFINE.

Quadratic Cipher Project

Quadratic Cipher

Recall that Shift Cipher was $f(x) = x + s \pmod{26}$.
For all s this function is 1-1 which is needed for encryption.

Quadratic Cipher

Recall that Shift Cipher was $f(x) = x + s \pmod{26}$.
For all s this function is 1-1 which is needed for encryption.

Recall that Affine cipher was $f(x) = ax + b \pmod{26}$.
For all a, b where a is relatively prime to 26 the function is 1-1 which is needed for encryption.

Quadratic Cipher

Recall that Shift Cipher was $f(x) = x + s \pmod{26}$.
For all s this function is 1-1 which is needed for encryption.

Recall that Affine cipher was $f(x) = ax + b \pmod{26}$.
For all a, b where a is relatively prime to 26 the function is 1-1 which is needed for encryption.

We now try a more complicated encryption.
$f(x) = ax^2 + bx + c \pmod{26}$.
NOT all a, b, c work. There is no easy test. So write a program that will, given a, b, c, determine if $f(x) = ax^2 + bx + c$ is 1-1 and onto.

Quadratic Cipher

Recall that Shift Cipher was $f(x) = x + s \pmod{26}$.
For all s this function is 1-1 which is needed for encryption.

Recall that Affine cipher was $f(x) = ax + b \pmod{26}$.
For all a, b where a is relatively prime to 26 the function is 1-1 which is needed for encryption.

We now try a more complicated encryption.
$f(x) = ax^2 + bx + c \pmod{26}$.
NOT all a, b, c work. There is no easy test. So write a program that will, given a, b, c, determine if $f(x) = ax^2 + bx + c$ is 1-1 and onto.

For each program you wrote for the SHIFT cipher, write a similar one for AFFINE.