Substitution, Even Queen Mary Version Was Broken Via Frequency Analysis and Similar Techniques.

We Want to Threat Frequency Analysis!

Possible 2 As in Plain $\Rightarrow$ Diff Cipher
2 Cipher Text Same Case From 2 Diff Plain.

Plain: GO KNIGHTS

Key: UC F UCFUCF

Diff Positions Are Shifted by Different Values.

Relatively Secure From 1500s $\Rightarrow$ 1800s.

```c
void encrypt(char* plain, char* key)
{
    int klen = strlen(key);
    int len = strlen(plain);

    for (int i = 0; i < len; i++)
    {
        int val = (plain[i] + (i % klen)) % 26;
        printf("0%c", key - 'a' + val);
    }
}
```
**How to break**

**Step 1: figure out keyword length**

**Kasiski Test:** find repeated trigrams or longer in ciphertext.

\[
\begin{align*}
\text{gap}_1 &= 71 \\
\text{gap}_2 &= 197 \\
\text{gap}_3 &= 350 \\
\end{align*}
\]

More likely same word lined up exactly at same key position.

\[ \text{gcd}(71, 197, 350) \]

\[ \text{gcd}(71, 197) = 126 \\
\text{gcd}(197, 350) = 153 \\
\text{gcd}(153, 126) = 27 \\
\]

\[ 153 = 1 \times 126 + 27 \\
126 = 4 \times 27 + 18 \\
27 = 1 \times 18 + 9 \\
18 = 2 \times 9 \\
\]
Other way to determine keyword length

Index of Coincidence

Problem: bag of candy
20 M&Ms
30 Snickers
5 reeses
45 Skittles

randomly select 2 bags of candy (w/o replacement)
What is the probability I get 2 of the same bag?

\[
\frac{20 \times 19}{100 \times 99} + \frac{30 \times 29}{100 \times 99} + \frac{5 \times 4}{100 \times 99} + \frac{45 \times 44}{100 \times 99}
\]

frequencies \( f_1, f_2, \ldots \), \( f_k \)

\[
\sum_{i=1}^{k} f_i = n \quad (n \text{ total items})
\]

\[
IC \text{ (Set)} = \sum_{i=1}^{k} \frac{f_i(f_i-1)}{n(n-1)}
\]

\[
\frac{19}{5 \times 99} + \frac{87}{10 \times 99} + \frac{1}{5 \times 99} + \frac{9}{20 \times 99} + \frac{49}{99} + \frac{22}{9}
\]

\[
\frac{38 + 87 + 2 + 198}{10 \times 99} = \frac{65}{2 \times 99} = \frac{65}{198}
\]
If letters all had same freq, then

\[ \text{LOC (English)} = \frac{1}{26} \sim 0.037 \text{ish} \]

but because freq are diff

\[ \text{LOC (English)} \sim 0.0676 \]

\[ \frac{98 \times 57}{100 \times 99} + 0 \]

To determine keyword length,
pretend key word was length = 2, 3, 4, 5, ...
for each guess, k, put the ciphertext letters in k bins.

\[ k = 5 \]

bin 1: \( C_1, C_6, C_{11}, C_{16} \) \( \sim \) LOC (bin1)
bin 2: \( C_2, C_7, C_{12}, C_{17} \) \( \sim \) LOC (bin2)
bin 3: \( C_3, C_8, C_{13}, C_{18} \) \( \sim \) LOC (bin3)
bin 4: \( C_4, C_9, C_{14}, C_{19} \) \( \sim \) LOC (bin4)
bin 5: \( C_5, C_{10}, C_{15}, C_{20} \) \( \sim \) LOC (bin5)

If guess is correct each bin will have a high LOC.
If not, they'll not be consistently high.
What to do when I know the length of the keyword?

5 letter language X:

bin 1

bin 2

bin 3

See which bar graph looks "most like" English.

Mutual Index of Coincidence Test

2 bags of candy 5 mm 20 snick 30 shtr 15 rce
15 mm 30 snick 15 shtr 5 rce

What's the probability if I grab 1 bag of candy from each set that I get the same bag?
\[
\text{MIC} = \frac{5}{70} \times \frac{15}{65} + \frac{20}{70} \times \frac{30}{65} + \frac{30}{70} \times \frac{15}{65} + \frac{15}{70} \times \frac{5}{65}
\]

\[
= 25 \left( 1 \times 3 + 4 \times 6 + 6 \times 3 + 3 \times 1 \right)
\]

\[
= \frac{48}{14 \times 13} = \frac{24}{7 \times 13} = \frac{24}{91}
\]