Playfair Cipher

The Playfair cipher was created by Charles Wheatstone and was one of the more popular diagram substitution ciphers in the 19th century. Once Vigenere was cracked, the idea was that less information could be gathered if two or more letters at a time were encrypted and mapped to a different set of two or more letters at a time. In general, a cipher that encrypts more than one letter at a time is called a polyalphabetic cipher. A general "substitution" chart for all $26^2 = 676$ possible digraphs would create a possible astounding number of 676! keys. Also, storing and using such a chart would be quite laborious. Thus, the Playfair cipher doesn't allow all of these possible keys, and that compromise is to make it very easy to implement by hand.

Here is how it's done:

1) Pick a key word.

2) Remove all repeated letters from it.

3) Fill in a 5 x5 grid from the top left going in order of rows, with your keyword.

4) Fill in the rest of the letters that are NOT in the keyword, in order, in the grid. Since there are only 25 grid spots, I and J share the same spot in the grid.

Here is an example with the keyword "PROBLEMS":

P	R	O	B	L
E	M	S	A	C
D	F	G	H	I/J
K	N	Q	T	U
V	W	X	Y	Z

Note that, if we picked a keyword of "TENNESSEE", it would be reduced to "TENS" in step 2.

In this cipher, we will encipher letters pairs at a time. Technically, the way the rules work, the rules only allow for pairs of letters that are not the same to be encrypted. We'll end up with a special rule for when this occurs in the plaintext.

Consider the following plaintext:

```
SHE WENT TO THE STORE
```

When we pair up the letters they get grouped as follows:

```
SH EW EN TT OT HE ST OR E
```

As previously mentioned, we are not allowed to encipher any double letters. So, in this case, we will insert an Q into the plaintext. (If Q is a double letter, then insert another infrequent letter, say

X.) So, in general, the fix to the double letter problem is to go through the plaintext, from left to right forming digraphs. Any time a repeated letter occurs, insert a Q in between. If the letter itself is a Q (hopefully not), then insert an X. Finally, if the last letter is by itself, also add the padding letter (Q).

Here is what we get for our example after adding padding:

```
SH EW EN TQ TO TH ES TO RE
```

To encipher pairs of letters, adhere to the following rules:

1) If the two letters are on the same row of the chart, like "ES", then replace each letter by the letter to the right. (If necessary, wrap around to the left end of the row. So "ES" encrypts to "MA".

2) If the two letters are on the same column of the chart, like, "TH", then replace each letter by the letter below it. (If necessary, wrap around to the top end of the column.) So "TH" encrypts to "YT".

3) If two letters are on a different row and column, like, "SH", then replace each letter by another letter on its same row, but in the column of the other letter. So "SH" encrypts to "AG".

Using these rules, here is the encryption of the plaintext above:

```
Plaintext : SH EW EN TQ TO TH ES TO RE
Ciphertext: AG MV MK UT QB YT MA QB PM
```

For decryption, if two ciphertext letters are on the same row or column, replace them with the two letters to the left or above, respectively. Otherwise, for each letter choose the letter on the same row and the other letter's column for decryption. (So this is the original operation, it is the reverse of itself.)

To cryptanalyze Playfair, we first might want to try to determine if a ciphertext is using Playfair. Here are some clues that it is:

1) There must be an even number of characters in the cipher text.

2) The rare consonants (j,k,q,x,z) will appear more frequently in the plaintext.

3) When divided into digraphs, no repeated letters will appear.

4) The frequency distribution of digraphs will approximate that of plaintext.

Here are some other unique characteristics of the Playfair cipher:

1) No single letter ever encrypts to itself.

2) Two reversed digraphs in the plaintext will always be represented by reverse digraphs in the ciphertext.

3) Every single letter from the plaintext can be enciphered by one of only five other letters – the one directly below it in the Playfair square or the other four in its row.

To perform a known-plaintext attack on the Playfair cipher, you try different positions of the known-plaintext to match with the ciphertext, and cross-check results with the rules above.

For example, if you tried to match the following:

```
Plaintext : asample
Ciphertext: pkkmkme
```

You can rule this out because this matches m to m and e to e, which is impossible in Playfair.

When you get a matching piece of plain and cipher text, you can start putting together possible placements of the key.

For example, consider the following matching:

```
Plaintext : asample
Ciphertext: ewdwqnb
```

From this you can make the following deductions:

a, s, e, and w are all on the same row, column or "box."
a, m, d and w are also on the same, row, column or "box."

It is quite likely that d and e are in the same row as a, and that w is in the same column as a.

In this manner you can attempt to start filling out the Playfair grid. Knowing that the keyword is in the beginning and the rest of the letters will roughly appear in alphabetical order can also help. From here, trial and error can eventually yield a solution.

Here is one more example:

Imagine that you know the first 8 letters of the ciphertext decrypt to “PLAYFAIR” and that we have the following “matching” plain-cipher text information:

PL -> QK
AY -> FV
FA -> GB
IR -> LE

Looking at the first set of letters, it is likely that they are in a box:

```
K  L
P  Q
```

The reason for this is that it would be bizarre for these letters to be on the same row, because they are spaced out quite a bit (M, N, and O are missing) – though that could be plausible for all three of those to be in the keyword, or at least two of them. If the arrangement above is correct, then the two columns and rows are likely consecutive.

It is also likely that the second set of letters forms a box as well, with V and Y on the last row:

```
A  F
V  Y
```

It is unlikely that they are all in the same column because V and Y are too close together, and nearly impossible unless they are all in the keyword that the letters are all on the same row.

The third set of letters strongly suggests that the letters are all on the same row, something like this:

```
A B   F G
```

The guess is that the middle letter in the row is missing and the other letters are in the keyword. Now, this gives us the following:

```
A B   F G
    K L
    P Q
V     Y
```

The placement of K, L, P and Q is a guess. It could be shifted to the left or the right in the square. It is quite safe to guess that the last letters of the alphabet fill out the bottom row since there's no extra room:

```
A B   F G
    K L
    P Q
V W X Y Z
```

Since I goes to L, it's likely that I is on the same row as L and that R is in the same column as L. This means there is only one place for R, It also strongly insinuates one place for I, since I/J go together:

```
      R
A B   F G
  I K L
    P Q
V W X Y Z
```

We also see that M, N, O fill out perfectly in between L and P, so none of these letters are in the key word:

```
      R
A B   F G
  I K L M
N O P Q
V W X Y Z
```

Two of S, T and U are in the keyword. Since S and T are very frequent, let's guess that U is NOT in the keyword, and H also falls into place:

```
      R
A B   F G
H I K L M
N O P Q U
V W X Y Z
```

Finally, we have that S, and T are in the keyword along with two of the following letters: C, D, E. A bit of trial and error (based on the cipher text) will lead to the following arrangement:

```
S E C R T
A B D F G
H I K L M
N O P Q U
V W X Y Z
```

which uses the keyword "secret."