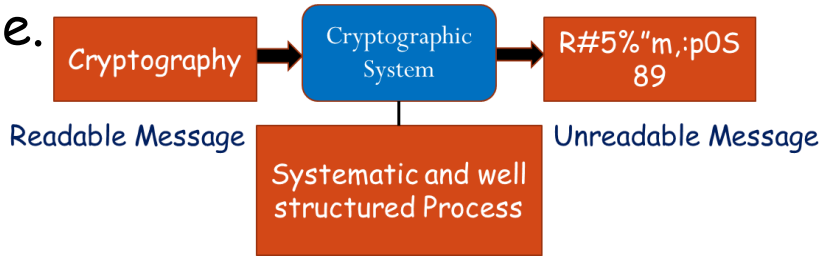
# Cryptography
# Classical Cipher

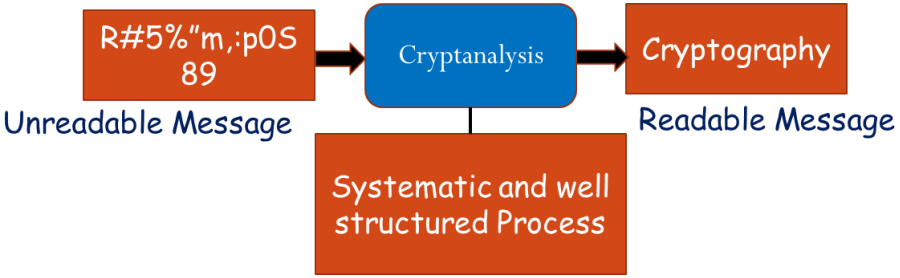
Module-1

# Introduction

- **Cryptography**
  - The art of achieving security by encoding messages to make them non-readable.




- **Cryptanalysis**
  - The technique of decoding messages from a non readable format back to a readable format without knowing how they were initially converted from readable format to non-readable format.




- **Cryptology**

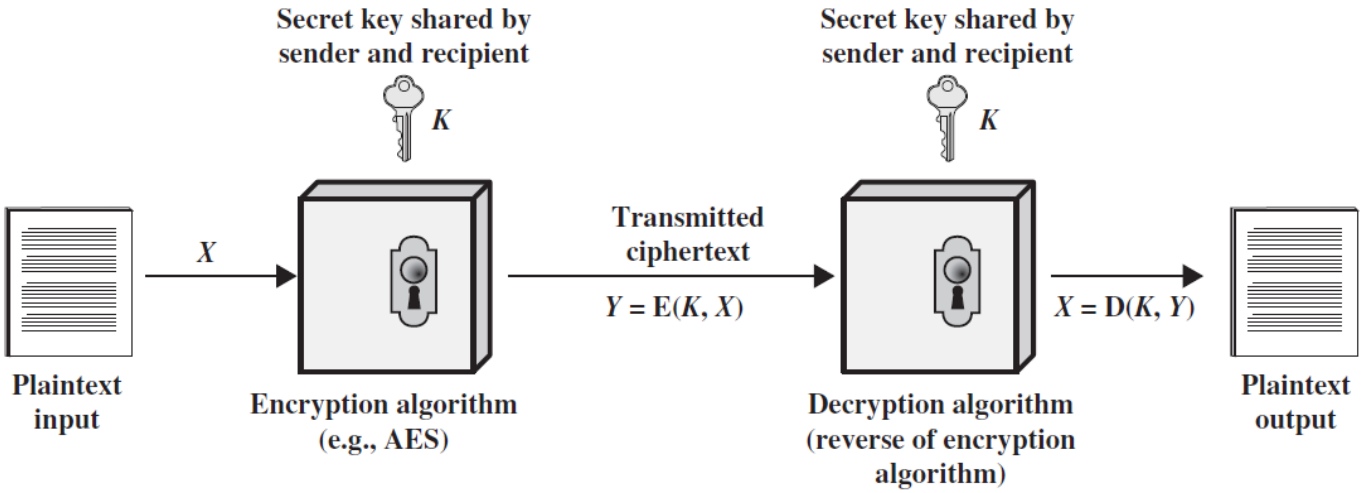
**Cryptology = Cryptography + Cryptanalysis**

# Symmetric Cipher Model

- **Plain text**
  - This is the original intelligible message or data that is fed into the algorithm as input.
  - Also Known as **cleartext**
- **Cipher Text**
  - This is the scrambled message produced as output.
  - It depends on the plaintext and the secret key.
  - For a given message, two different keys will produce two different ciphertexts.
  - The ciphertext is an apparently random stream of data and, as it stands, is unintelligible.
  - Also Known as **secrettext**
- **Encryption algorithm**
  - Performs various substitutions and transformations on the plaintext.
- **Secret key**
  - The secret key is also input to the encryption algorithm.
  - The key is a value independent of the plaintext and of the algorithm.
  - The algorithm will produce a different output depending on the specific key being used at the time.
  - The exact substitutions and transformations performed by the algorithm depend on the key.

# Symmetric Cipher Model

- **Decryption algorithm**
  - This is essentially the encryption algorithm run in reverse.
  - It takes the ciphertext and the secret key and produces the original plaintext.

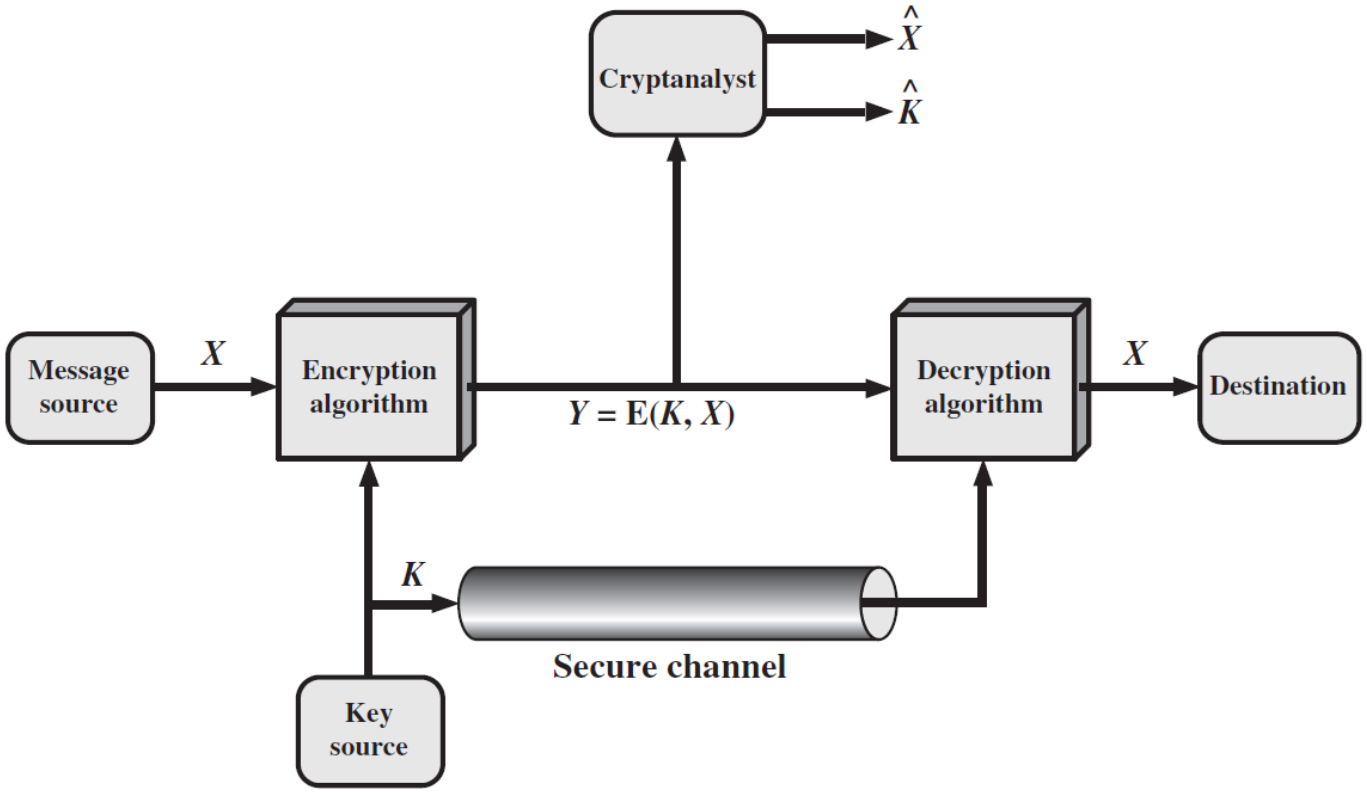



- **There are two requirements for secure use of conventional encryption:**
  1. We need a strong encryption algorithm.
  2. Sender and receiver must have obtained copies of the secret key in a secure fashion and must keep the key secure.

# Introduction

- **Cryptographic system**
  - It is a five tuple (P, C, K, E, D), where the following conditions are satisfied
    1. P is a finite set of possible plaintexts
    2. C is a finite set of possible ciphertexts
    3. K, the keyspace, is a finite set of possible keys
    4. For each $k \varepsilon K$, there is an encryption rule $e_k \varepsilon E$ and a corresponding decryption rule $d_k \varepsilon D$. Each $e_k$: $P \rightarrow C$ and $d_k$: $C \rightarrow P$ are functions such that $d_k(e_k(x)) = x$ for every plaintext element $x \varepsilon P$.

# Model of Symmetric Cryptosystem

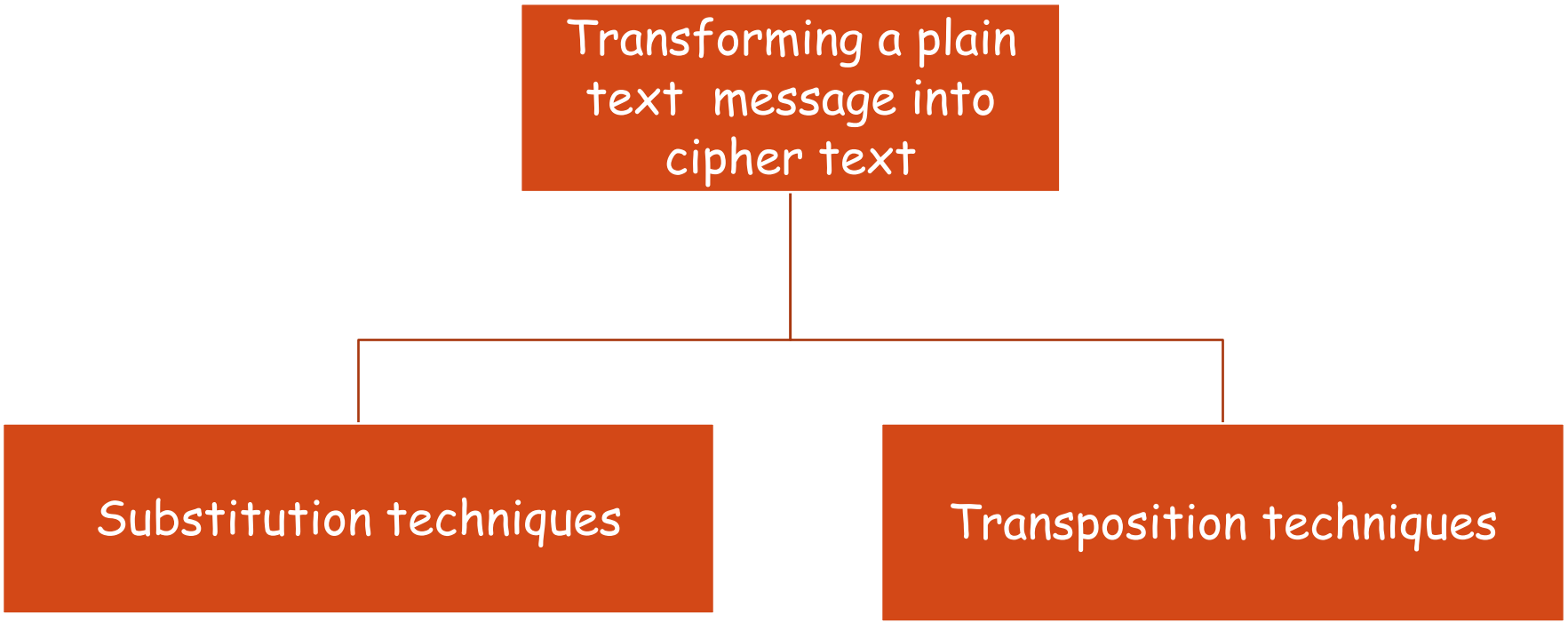



Original Message string $x = x_1x_2...x_n$ for, $n>=1$ where, $x_i \varepsilon P$, $1 \le i \le n$

$y = e_k(x_i)$, $1 \le i \le n$ and resulting cipher text string $y = y_1y_2...y_n$

$y = e_k(x_1) = e_k(x_2)$ where $x_1 \neq x_2$

# Techniques for transforming Plain text to Cipher text



# Shift Cipher or Caesar Cipher

- Modular Arithmetic
  - Suppose a and b are integers, and m is a positive integer. Then we write $a \equiv b \bmod m$ if m divides b-a. The phrase $a \equiv b \bmod m$ is called a congruence, and it is read as "a is congruent to b modulo m". The integer **m** is called the modulus.
  - If we replace **a** by **a mod m**, a is reduced modulo m
  - If $a = q_1 m + r_1$ and $b = q_2 m + r_2$ and $r_1 = r_2$, then we can say that $a \equiv b \bmod m$.
  - Arithmetic modulo m: $Z_m$ is the set {0,….,m-1}, equipped with two operations, + and *. Addition and multiplication in $Z_m$ work exactly like real addition and multiplication, except that the results are reduced modulo m.

# Shift Cipher or Caesar Cipher

- Let $P = C = K = Z_{26}$ for $0 \leq K \leq 25$, define

  $e_K(x) = (x+K) \bmod 26$

And

$d_K(y) = (y-K) \bmod 26$ where $x, y \in Z_{26}$

- The Caesar cipher involves replacing each letter of the alphabet with the letter standing three places further down the alphabet (i.e. K=3).
- Purportedly used by Julius Caesar

**plain:** meet me after the toga party

**cipher:** PHHW PH DIWHU WKH WRJD SDUWB

$C = E(3, p) = (p + 3) \bmod 26$

# Example

| $A$ | $B$ | $C$ | $D$ | $E$ | $F$ | $G$ | $H$ | $I$ | $J$ | $K$ | $L$ | $M$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

| $N$ | $O$ | $P$ | $Q$ | $R$ | $S$ | $T$ | $U$ | $V$ | $W$ | $X$ | $Y$ | $Z$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |

`wewillmeetatmidnight.`

We first convert the plaintext to a sequence of integers using the specified correspondence, obtaining the following:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 22 | 4 | 22 | 8 | 11 | 11 | 12 | 4 | 4 | 19 |
| 0 | 19 | 12 | 8 | 3 | 13 | 8 | 6 | 7 | 19 |

Next, we add 11 to each value, reducing each sum modulo 26:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 7 | 15 | 7 | 19 | 22 | 22 | 23 | 15 | 15 | 4 |
| 11 | 4 | 23 | 19 | 14 | 24 | 19 | 17 | 18 | 4 |

Finally, we convert the sequence of integers to alphabetic characters, obtaining the ciphertext:

`HPHTWWXPPELEXTOYTRSE.`

# Cryptanalysis of Caesar Cipher

- Not very much secure
- Cryptanalyzed by the obvious method of exhaustive key search
- Since there are only 26 possible keys, it is easy to try every possible decryption rule until a meaningful plaintext string is obtained
- Three important characteristics of this problem enabled us to use a brute force cryptanalysis:
  - The encryption and decryption algorithms are known.
  - There are only 25 keys to try.
  - The language of the plaintext is known and easily recognizable

# Cryptanalysis of Caesar Cipher (Cont..)

- **Brute-Force Attack**
  - An attack on a cipher text message, wherein the attacker attempts to use all possible permutations and combinations
- A cryptanalyst attempting a brute-force attack tries to derives the original plain text message from a given Cipher text message

| KEY | PHHW PH DIWHU WKH WRJD SDUWB |
|---|---|
| 1 | oggv og chvgt vjg vqic rctva |
| 2 | nffu nf bgufs uif uphb qbsuz |
| 3 | meet me after the toga party |
| 4 | ldds ld zesdq sgd snfz ozqsx |
| 5 | kccr kc ydrcp rfc rmey nyprw |
| 6 | jbbq jb xcqbo qeb qldx mxoqv |
| 7 | iaap ia wbpan pda pkcw lwnpu |
| 8 | hzzo hz vaozm ocz ojbv kvmot |
| 9 | gyyn gy uznyl nby niau julns |
| 10 | fxxm fx tymxk max mhzt itkmr |
| 11 | ewwl ew sxlwj lzw lgys hsjlq |
| 12 | dvvk dv rwkvi kyv kfxr grikp |
| 13 | cuuj cu qvjuh jxu jewq fqhjo |
| 14 | btti bt puitg iwt idvp epgin |
| 15 | assh as othsf hvs hcuo dofhm |
| 16 | zrrg zr nsgre gur gbtn cnegl |
| 17 | yqqf yq mrfqd ftq fasm bmdfk |
| 18 | xppe xp lqepc esp ezrl alcej |
| 19 | wood wo kpdob dro dyqk zkbdi |
| 20 | vnnc vn jocna cqn cxpj yjach |
| 21 | ummb um inbmz bpm bwoi xizbg |
| 22 | tlla tl hmaly aol avnh whyaf |
| 23 | skkz sk glzkx znk zumg vgxze |
| 24 | rjjy rj fkyjw ymj ytlf ufwyd |
| 25 | qiix qi ejxiv xli xske tevxc |

# Substitution Cipher or Mono-alphabetic Cipher

- A single cipher alphabet (mapping from plain alphabet to cipher alphabet) is used per message
- Let P = C = Z26

  K consists of all possible permutations of the 26 symbols 0,1,…,25

  For each permutation $\pi \varepsilon K$, define

  $$e_\pi (x) = \pi (x) \text{ and define}$$

  $$d_\pi (y) = \pi^{-1}(y)$$

  - Where $\pi^{-1}$ is the inverse permutation to $\pi$.

- Permutation
  - a finite set of elements S is an ordered sequence of all the elements of S, with each element appearing exactly once.
  - **Example**, if S = {a, b, c}, there are six permutations of S:

    abc, acb, bac, bca, cab, cba

- There are 26! or greater than $4 * 10^{26}$ possible keys.

# Example Problem

| $a$ | $b$ | $c$ | $d$ | $e$ | $f$ | $g$ | $h$ | $i$ | $j$ | $k$ | $l$ | $m$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $X$ | $N$ | $Y$ | $A$ | $H$ | $P$ | $O$ | $G$ | $Z$ | $Q$ | $W$ | $B$ | $T$ |

| $n$ | $o$ | $p$ | $q$ | $r$ | $s$ | $t$ | $u$ | $v$ | $w$ | $x$ | $y$ | $z$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S$ | $F$ | $L$ | $R$ | $C$ | $V$ | $M$ | $U$ | $E$ | $K$ | $J$ | $D$ | $I$ |

$e_\pi$ (a) = X, $e_\pi$ (b) = N and so on…

- **Decryption Function**
  - Inverse permutation
    - Writing the first line first, and then sorting in alphabetic order

| $A$ | $B$ | $C$ | $D$ | $E$ | $F$ | $G$ | $H$ | $I$ | $J$ | $K$ | $L$ | $M$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $d$ | $l$ | $r$ | $y$ | $v$ | $o$ | $h$ | $e$ | $z$ | $x$ | $w$ | $p$ | $t$ |

| $N$ | $O$ | $P$ | $Q$ | $R$ | $S$ | $T$ | $U$ | $V$ | $W$ | $X$ | $Y$ | $Z$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $b$ | $g$ | $f$ | $j$ | $q$ | $n$ | $m$ | $u$ | $s$ | $k$ | $a$ | $c$ | $i$ |

Hence, $d_\pi$ (A) = d, $d_\pi$ (B) = l and so on…

# Cryptanalysis of Substitution Cipher (Frequency Analysis)

- If the cryptanalyst knows the nature of the plaintext (e.g., non compressed English text)
- Then the analyst can exploit the regularities of the language.
- The cipher text to be solved is

  UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
  VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
  EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ

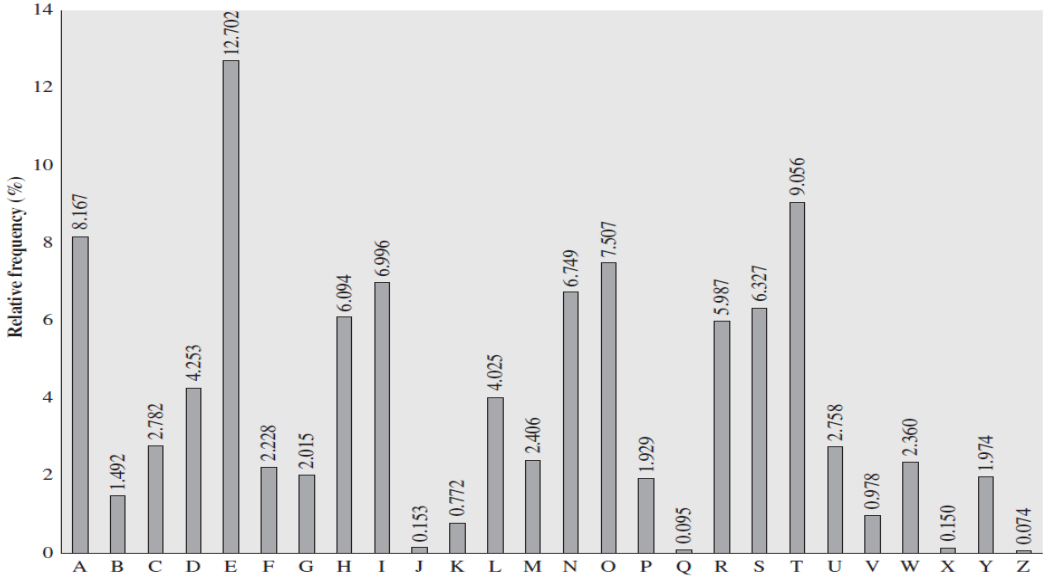
- As a first step, the relative frequency of the letters can be determined and compared to a standard frequency distribution for English
- If the message were long enough, this technique alone might be sufficient

# Cryptanalysis of Substitution Cipher (Frequency Analysis)

- The relative frequencies of the letters in the Cipher text (in percentages) are as follows:

| | | | | |
|---|---|---|---|---|
| P 13.33 | H 5.83 | F 3.33 | B 1.67 | C 0.00 |
| Z 11.67 | D 5.00 | W 3.33 | G 1.67 | K 0.00 |
| S 8.33 | E 5.00 | Q 2.50 | Y 1.67 | L 0.00 |
| U 8.33 | V 4.17 | T 2.50 | I 0.83 | N 0.00 |
| O 7.50 | X 4.17 | A 1.67 | J 0.83 | R 0.00 |
| M 6.67 | | | | |

- Comparing this breakdown with the figure of relative frequency of letters in English text, it seems likely that cipher letters P and Z are the equivalents of plain letters e and t, but it is not certain which is which.



# Cryptanalysis of Substitution Cipher (Frequency Analysis)

- The letters S, U, O, M, and H are all of relatively high frequency and probably correspond to plain letters from the set {a, h, i, n, o, r, s}.
- The letters with the lowest frequencies (namely, A, B, G, Y, I, J) are likely included in the set {b, j, k, q, v, x, z}.
- There are a number of ways to proceed at this point.
- We could make some tentative assignments and start to fill in the plaintext to see if it looks like a reasonable "skeleton" of a message.
- A more systematic approach is to look for other regularities.
- For example, certain words may be known to be in the text.
- Or we could look for repeating sequences of cipher letters and try to deduce their plaintext equivalents.

# Cryptanalysis of Substitution Cipher (Frequency Analysis)

- So far, then, we have

```
UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
 t a        e  e te  a that e e a       a
VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
   e t   ta t ha e ee  a e  th    t  a
EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ
 e  e e tat  e   the   t
```

- Only four letters have been identified, but already we have quite a bit of the message.
- Continued analysis of frequencies plus trial and error should easily yield a solution from this point.
- The complete plaintext, with spaces added between words, follows:

  it was disclosed yesterday that several informal but direct contacts have been made with political representatives of the viet cong in moscow

# Affine Cipher

- Combination of additive and multiplicative ciphers with a pair of keys
- First key is used with the multiplicative cipher and second key is used with the additive cipher
- The encryption and decryption are

$$e_K(x) = (P*K_1+K_2) \bmod 26$$

and

$$d_K(y) = ((y-K_2)*K_1^{-1}) \bmod 26$$

# Example Problem

- Encrypt the message "hello" with the key pair (7,2)

| | | |
|---|---|---|
| P: h → 07 | Encryption: (07 × 7 + 2) mod 26 | C: 25 → Z |
| P: e → 04 | Encryption: (04 × 7 + 2) mod 26 | C: 04 → E |
| P: l → 11 | Encryption: (11 × 7 + 2) mod 26 | C: 01 → B |
| P: l → 11 | Encryption: (11 × 7 + 2) mod 26 | C: 01 → B |
| P: o → 14 | Encryption: (14 × 7 + 2) mod 26 | C: 22 → W |

- Decrypt the message "ZEBBW" with the key pair (7,2)

| | | |
|---|---|---|
| C: Z → 25 | Decryption: $((25 - 2) \times 7^{-1}) \bmod 26$ | P: 07 → h |
| C: E → 04 | Decryption: $((04 - 2) \times 7^{-1}) \bmod 26$ | P: 04 → e |
| C: B → 01 | Decryption: $((01 - 2) \times 7^{-1}) \bmod 26$ | P: 11 → l |
| C: B → 01 | Decryption: $((01 - 2) \times 7^{-1}) \bmod 26$ | P: 11 → l |
| C: W → 22 | Decryption: $((22 - 2) \times 7^{-1}) \bmod 26$ | P: 14 → o |

# Extended Euclidean Algorithm For Calculating Multiplicative inverse

1. r1 ← n; r2←b;
2. t1 ← 0; t2←1;
3. while (r2>0)
4. {
5. q ← r1/r2;
6. r ← r1 - q * r2;
7. r1 ← r2; r2 ← r;
8. t ← t1 - q * t2;
9. t1 ← t2; t2←t;
10. }
11. if (r1 = = 1) then b-1 ← t1

# Cryptanalysis of Affine Cipher

- Assume that attacker intercept the following Ciphertext
  PWUFFOGWCHFDWIWEJOUUNJORSMDWRHVCMWJUPVCCG
- By using the frequency analysis:

Plaintext: et Ciphertext: WC

Plaintext: et Ciphertext: WF

$e \rightarrow W$ $4 \rightarrow 22$ $(4*k_1)+k_2 \bmod 26=22$

$t \rightarrow C$ $19 \rightarrow 2$ $(19*k_1)+k_2 \bmod 26=2$

$$\begin{pmatrix} 4 & 1 \\ 19 & 1 \end{pmatrix} \begin{pmatrix} K_1 \\ K_2 \end{pmatrix} = \begin{pmatrix} 22 \\ 2 \end{pmatrix}$$

$A^{-1} = 1/\det[\text{Adj } A]$

$$\begin{pmatrix} K_1 \\ K_2 \end{pmatrix} = \begin{pmatrix} 22 \\ 2 \end{pmatrix} \begin{pmatrix} 4 & 1 \\ 19 & 1 \end{pmatrix}^{-1}$$

$$\begin{pmatrix} 4 & 1 \\ 19 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} -1/15 & 1/15 \\ 19/15 & -4/15 \end{pmatrix} = \begin{pmatrix} -15^{-1} & 15^{-1} \\ 19*15^{-1} & -4*15^{-1} \end{pmatrix} = \begin{pmatrix} -7+26 & 7 \\ 3 & -2+26 \end{pmatrix} = \begin{pmatrix} 19 & 7 \\ 3 & 24 \end{pmatrix}$$

$$\begin{pmatrix} K_1 \\ K_2 \end{pmatrix} = \begin{pmatrix} 22 \\ 2 \end{pmatrix} \begin{pmatrix} 19 & 7 \\ 3 & 24 \end{pmatrix} = \begin{pmatrix} 19*22 + 7*2 \\ 3*22 + 24*2 \end{pmatrix} = \begin{pmatrix} 432 \\ 114 \end{pmatrix} = \begin{pmatrix} 16 \\ 10 \end{pmatrix}$$

So, $K_1 = 16$ and $K_2 = 10$

However, this answer is not acceptable because $K_1 = 16$ cannot be the first part of the key

Its value, 16, does not have a multiplicative inverse in $Z_{26}^*$ i.e. $\gcd(K_1,26) = 2 > 1$

Now tries the result of the second set of data

| | | |
|---|---|---|
| e→W | 4→22 | $(4*k_1)+k_2 \bmod 26=22$ |
| t →F | 19→5 | $(19*k_1)+k_2 \bmod 26=5$ |

$$\begin{pmatrix} 14 & 1 \\ 19 & 1 \end{pmatrix} \begin{pmatrix} K_1 \\ K_2 \end{pmatrix} = \begin{pmatrix} 22 \\ 5 \end{pmatrix}$$

$$\begin{pmatrix} K_1 \\ K_2 \end{pmatrix} = \begin{pmatrix} 22 \\ 5 \end{pmatrix} \begin{pmatrix} 14 & 1 \\ 19 & 1 \end{pmatrix}^{-1}$$

Similarly we find out $K_1 = 11$ and $K_2 = 4$

However, this answer is acceptable because $K_1 = 11$ can be the first part of the key

Its value, 11, does have a multiplicative inverse in $Z_{26}^*$ i.e. $\gcd(K_1,26) = 1$

$d_K(y) = ((y-K_2)*K_1^{-1}) \bmod 26$
$(15-4)*19 \bmod 26 = 1 =$ **b**
$(22-4)*19 \bmod 26 = 4 =$ **e**
$(20-4)*19 \bmod 26 = 18 =$ **s**
$(5-4)*19 \bmod 26 = 19 =$ **t**

In the same manner we find all the character of plaintext, so, the plaintext is

**best time of the year is spring when flowers bloom**

# Polyalphabetic Cipher: Vigenere Cipher

- Designed by Blaise de Vigenere
- Associate each key K with an alphabet string of length m, called a keyword
- Encrypts m alphabetic characters at time
- Key stream does not depend on the plaintext characters
- It depends only on the position of the character in the plaintext
- The first letter of the key is added to the first letter of the plaintext, mod 26,
- The second letters are added, and so on through the first m letters of the plaintext.
- For the next m letters of the plaintext, the key letters are repeated
- This process continues until all of the plaintext sequence is encrypted

# Polyalphabetic Cipher: Vigenere Cipher

- Let m be a positive integer
- Define $P = C = K = (Z_{26})^m$, for a key $K = (k_1, k_2, \ldots, k_m)$
- We define,
  - **Encryption equation as:**

$$C_i = (p_i + k_i \bmod m) \bmod 26$$

  - **Decryption equation as:**

$$pi = (Ci - ki \bmod m) \bmod 26$$

- Where all operations are performed in Z26

# Vigenere Cipher: Example

- Suppose m = 9 and the keyword is **deceptive**
- This corresponds to the numerical equivalent K=(3,4,2,4,15,19,8,21,4)
- Suppose the plaintext is the string

  "we are discovered save yourself"
- We convert the plaintext elements to residues modulo 26, write them in groups of nine, and the keyword modulo 26, as follows:

| key | 3 | 4 | 2 | 4 | 15 | 19 | 8 | 21 | 4 | 3 | 4 | 2 | 4 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| plaintext | 22 | 4 | 0 | 17 | 4 | 3 | 8 | 18 | 2 | 14 | 21 | 4 | 17 | 4 |
| ciphertext | 25 | 8 | 2 | 21 | 19 | 22 | 16 | 13 | 6 | 17 | 25 | 6 | 21 | 19 |

| key | 19 | 8 | 21 | 4 | 3 | 4 | 2 | 4 | 15 | 19 | 8 | 21 | 4 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| plaintext | 3 | 18 | 0 | 21 | 4 | 24 | 14 | 20 | 17 | 18 | 4 | 11 | 5 |
| ciphertext | 22 | 0 | 21 | 25 | 7 | 2 | 16 | 24 | 6 | 11 | 12 | 6 | 9 |

# Vigenere Cipher (Cont..)

- The strength of this cipher is that there are multiple ciphertext letters for each plaintext letter, one for each unique letter of the keyword.
- Thus, the letter frequency information is obscured.
- However, not all knowledge of the plaintext structure is lost.

# Cryptanalysis of Vigenere Cipher

- Cryptanalysis here consists of two parts:
  1. Finding the length of the key (m)
     - **Kasiski test** is used for find the length of the key
     - Cryptanalyst searches for repeated text segments, of at least three characters, in the cipher text
     - If two of these segments are found and the distance between them is d
     - If we obtain several search such distances, say $d_1$, $d_2$,…,$d_n$ then we would conjecture that m divides all of the $d_i$'s ($d_i$/m) and gcd ($d_1$, $d_2$,…,$d_n$)/m
     - The Index of Coincidence method is often used to confirm the m value determined by the Kasiski test
       - The Index of Coincidence of $x=x_1x_2…x_n$ which is a string of length n formed by the alphabets A,B,…,Z is defined as the probability that the random elements of x are the same
       - If the frequencies of A,B,…,Z in x are denoted by the $f_0$,…$f_{25}$

$$I_c(\mathbf{x}) = \frac{\sum_{i=0}^{25} \binom{f_i}{2}}{\binom{n}{2}} = \frac{\sum_{i=0}^{25} f_i(f_i-1)}{n(n-1)} \qquad I_c(\mathbf{x}) \approx \sum_{i=0}^{25} {p_i}^2 = 0.065$$

# Cryptanalysis of Vigenere Cipher

- Using the m value of the kasiski test, we arrange the given alphabet string y = $y_1y_2...y_n$ into m substrings as follows

$$\mathbf{y}_1 = y_1 y_{m+1} y_{2m+1} \cdots,$$

$$\mathbf{y}_2 = y_2 y_{m+2} y_{2m+2} \cdots,$$

$$\vdots \quad \vdots \quad \vdots$$

$$\mathbf{y}_m = y_m y_{2m} y_{3m} \cdots.$$

2. Finding the key itself
   - We need the concept of Mutual Index of Coincidence (MI) between two alphabetic strings x and y
     - Suppose $x=x_1x_2...x_n$ and $y=y_1y_2...y_n$ are two alphabetic strings
     - Probability that a random element of x is equal to that y
     - if the probabilities of A,B,...are $f_0,...f_{25}$ and $f'_0,...f'_{25}$ respectively in the x and y, then

$$MI_c(x,y) = \sum_{i=0}^{25} f_i f'_i / nn'$$

# Cryptanalysis of Vigenere Cipher: Example

***Example 1.12*** Ciphertext obtained from a *Vigenère Cipher*

```
CHREEVOAHMAERATBIAXXWTNXBEEOPHBSBQMQEQERBW
RVXUOAKXAOSXXWEAHBWGJMMQMNKGRFVGXWTRZXWIAK
LXFPSKAUTEMNDCMGTSXMXBTUIADNGMGPSRELXNJELX
VRVPRTULHDNQWTWDTYGBPHXTFALJHASVBFXNGLLCHR
ZBWELEKMSJIKNBHWRJGNMGJSGLXFEYPHAGNRBIEQJT
AMRVLCRREMNDGLXRRIMGNSNRWCHRQHAEYEVTAQEBBI
PEEWEVKAKOEWADREMXMTBHHCHRTKDNVRZCHRCLQOHP
WQAIIWXNRMGWOIIFKEE
```

First, let's try the Kasiski test. The ciphertext string $CHR$ occurs in five places in the ciphertext, beginning at positions 1, 166, 236, 276 and 286. The distances from the first occurrence to the other four occurrences are (respectively) 165, 235, 275 and 285. The greatest common divisor of these four integers is 5, so that is very likely the keyword length.

Let's see if computation of indices of coincidence gives the same conclusion. With $m = 1$, the index of coincidence is 0.045. With $m = 2$, the two indices are 0.046 and 0.041. With $m = 3$, we get 0.043, 0.050, 0.047. With $m = 4$, we have indices 0.042, 0.039, 0.045, 0.040. Then, trying $m = 5$, we obtain the values 0.063, 0.068, 0.069, 0.061 and 0.072. This also provides strong evidence that the keyword length is five. ∎

# Cryptanalysis of Vigenere Cipher: Example

| $i$ | value of $M_g(\mathbf{y}_i)$ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | .035 | .031 | .036 | .037 | .035 | .039 | .028 | .028 | .048 |
| | .061 | .039 | .032 | .040 | .038 | .038 | .045 | .036 | .030 |
| | .042 | .043 | .036 | .033 | .049 | .043 | .042 | .036 | |
| 2 | .069 | .044 | .032 | .035 | .044 | .034 | .036 | .033 | .029 |
| | .031 | .042 | .045 | .040 | .045 | .046 | .042 | .037 | .032 |
| | .034 | .037 | .032 | .034 | .043 | .032 | .026 | .047 | |
| 3 | .048 | .029 | .042 | .043 | .044 | .034 | .038 | .035 | .032 |
| | .049 | .035 | .031 | .035 | .066 | .035 | .038 | .036 | .045 |
| | .027 | .035 | .034 | .034 | .036 | .035 | .046 | .040 | |
| 4 | .045 | .032 | .033 | .038 | .060 | .034 | .034 | .034 | .050 |
| | .033 | .033 | .043 | .040 | .033 | .029 | .036 | .040 | .044 |
| | .037 | .050 | .034 | .034 | .039 | .044 | .038 | .035 | |
| 5 | .034 | .031 | .035 | .044 | .047 | .037 | .043 | .038 | .042 |
| | .037 | .033 | .032 | .036 | .037 | .036 | .045 | .032 | .029 |
| | .044 | .072 | .037 | .027 | .031 | .048 | .036 | .037 | |

# Cryptanalysis of Vigenere Cipher: Example

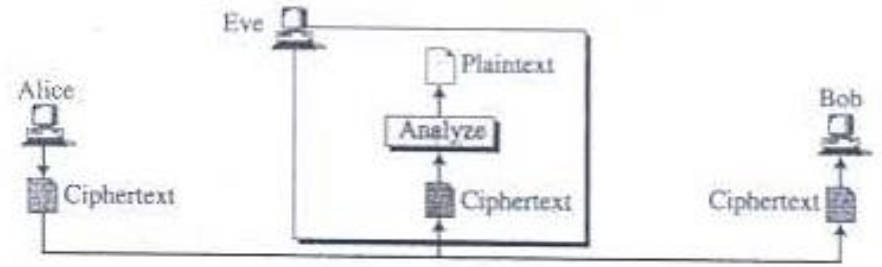
***Example 1.12*** ***(Cont.)*** We have hypothesized that the keyword length is 5. We now compute the values $M_g$ as described above, for $1 \leq i \leq 5$. These values are tabulated in Table 1.4. For each $i$, we look for a value of $M_g$ that is close to 0.065. These $g$'s determine the shifts $k_1, \ldots, k_5$.

From the data in Table 1.4, we see that the key is likely to be $K = (9, 0, 13, 4, 19)$, and hence the keyword likely is $JANET$. This is correct, and the complete decryption of the ciphertext is the following:

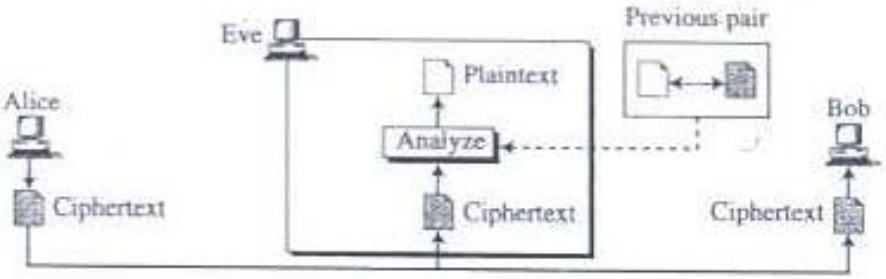
The almond tree was in tentative blossom. The days were longer, often ending with magnificent evenings of corrugated pink skies. The hunting season was over, with hounds and guns put away for six months. The vineyards were busy again as the well-organized farmers treated their vines and the more lackadaisical neighbors hurried to do the pruning they should have done in November.[4]

# Cryptanalysis Attack

- Ciphertext Only Attack:
  - Attacker has access to only ciphertext
  - Attacker try to find the corresponding key and can intercept the ciphertext
  - Example: Brute Force Attack, Statistical Attack and Pattern Attack

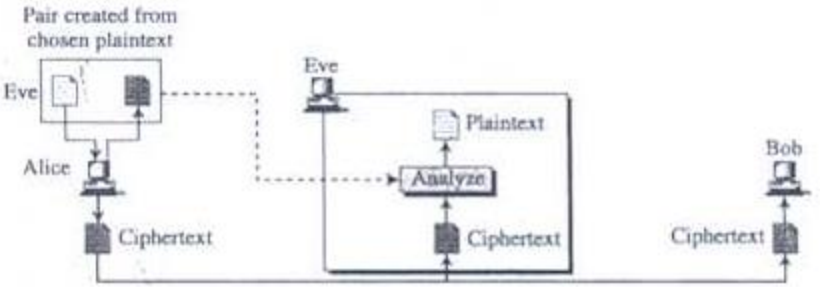



- Known Plaintext Attack:
  - Attacker has access to some Plaintext/Ciphertext pairs in addition to intercepted ciphertext that attacker wants to break
  - The Plaintext/Ciphertext pairs have been collected earlier

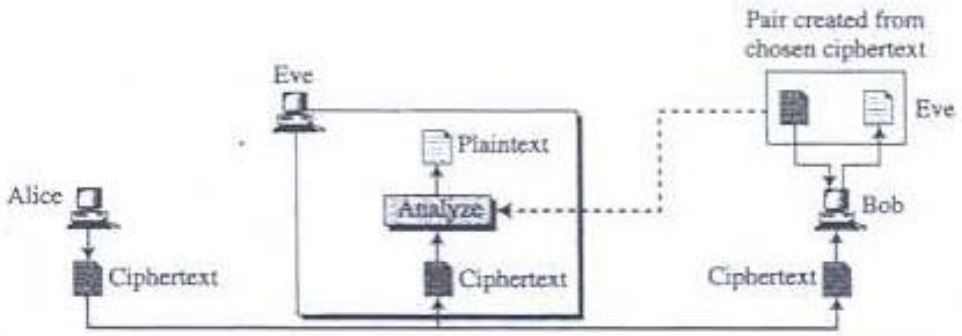


# Cryptanalysis Attack

- Chosen Plaintext Attack:
  - Similar to the known plaintext attack
  - But the Plaintext/Ciphertext pairs have been chosen by the attacker itself that result in obtaining more information about the key




- Chosen Ciphertext Attack:
  - Similar to the chosen plaintext attack
  - Attacker chooses some ciphertext and decrypts it to form a ciphertext/plaintext pair



# Cryptanalysis Attack

| Type of Attack | Known to Cryptanalyst |
|---|---|
| Ciphertext Only | • Encryption algorithm<br>• Ciphertext |
| Known Plaintext | • Encryption algorithm<br>• Ciphertext<br>• One or more plaintext–ciphertext pairs formed with the secret key |
| Chosen Plaintext | • Encryption algorithm<br>• Ciphertext<br>• Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key |
| Chosen Ciphertext | • Encryption algorithm<br>• Ciphertext<br>• Ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key |
| Chosen Text | • Encryption algorithm<br>• Ciphertext<br>• Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key<br>• Ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key |

# Hill Cipher

- Polyalphabetic cryptosystem
- Invented in 1929 by Lester S. Hill
- Takes m successive plaintext letters and substitutes for them m ciphertext letters
- Plaintext is divided into equal size blocks
- Belongs to a category of ciphers called block cipher
- The key, say K, is a square matrix of size m x m in which m is the size of the block

$$\mathbf{K} = \begin{pmatrix} k_{1,1} & k_{1,2} & \dots & k_{1,m} \\ k_{2,1} & k_{2,2} & \dots & k_{2,m} \\ \vdots & \vdots & & \vdots \\ k_{m,1} & k_{m,2} & \dots & k_{m,m} \end{pmatrix}$$

# Hill Cipher (Cont..)

- The substitution is determined by m linear equations in which each character is assigned a numerical value (a = 0, b = 1, c, z = 25)
- For m = 3, the system can be described as

$c1 = (k_{11}p_1 + k_{21}p_2 + k_{31}p_3) \bmod 26$

$c2 = (k_{12}p_1 + k_{22}p_2 + k_{32}p_3) \bmod 26$

$c3 = (k_{13}p_1 + k_{23}p_2 + k_{33}p_3) \bmod 26$

- This can be expressed in terms of row vectors and matrices:

$$(c_1\ c_2\ c_3) = (p_1\ p_2\ p_3)\begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{pmatrix} \bmod 26$$

or

$$\mathbf{C} = \mathbf{PK} \bmod 26$$

# Hill Cipher: Example

Plaintext : "**Code is Ready**"

- **Encryption:** $e_k(x) = xK$

$$\underset{C}{\begin{pmatrix} 14 & 7 & 10 & 13 \\ 8 & 7 & 6 & 11 \\ 11 & 8 & 18 & 18 \end{pmatrix}} = \underset{P}{\begin{pmatrix} 2 & 14 & 3 & 4 \\ 8 & 18 & 17 & 4 \\ 0 & 3 & 24 & 25 \end{pmatrix}} \underset{K}{\begin{pmatrix} 9 & 7 & 11 & 13 \\ 4 & 7 & 5 & 6 \\ 2 & 21 & 14 & 9 \\ 3 & 23 & 21 & 8 \end{pmatrix}}$$

Ciphertext : "**OHKNIHGKLISS**"

- **Decryption:** $d_k(y) = yK^{-1}$

$$\underset{P}{\begin{pmatrix} 2 & 14 & 3 & 4 \\ 8 & 18 & 17 & 4 \\ 0 & 3 & 24 & 25 \end{pmatrix}} = \underset{C}{\begin{pmatrix} 2 & 14 & 3 & 4 \\ 8 & 18 & 17 & 4 \\ 0 & 3 & 24 & 25 \end{pmatrix}} \underset{K^{-1}}{\begin{pmatrix} 9 & 7 & 11 & 13 \\ 4 & 7 & 5 & 6 \\ 2 & 21 & 14 & 9 \\ 3 & 23 & 21 & 8 \end{pmatrix}}$$

# Cryptanalysis of Hill Cipher

- Hill cipher is strong against a ciphertext-only attack, it is easily broken with a known plaintext attack
- For an m * m Hill cipher, suppose we have m plaintext-ciphertext pairs, each of length m
- We label the pairs $P_j = (p_{1j}\ p_{1j} \ldots p_{mj})$ and $C_j = (c_{1j}c_{1j} \ldots c_{mj})$ such that $C_j = P_jK$ for $1 \leq j \leq m$ and for some unknown key matrix K
- Now define two m * m matrices $X = (p_{ij})$ and $Y = (c_{ij})$
- Then we can form the matrix equation $Y = XK$
- If X has an inverse, then we can determine $K = X^{-1}Y$
- If X is not invertible, then a new version of X can be formed with additional plaintext-ciphertext pairs until an invertible X is obtained.

# Cryptanalysis of Hill Cipher (Cont..)

- Suppose that the plaintext "**hillcipher**" is encrypted using a 2 * 2 Hill cipher to yield the ciphertext "**HCRZSSXNSP**"
- Thus, we know that (7 8)K mod 26 = (7 2);
- (11 11)K mod 26 = (17 25); and so on
- Using the first two plaintext–ciphertext pairs, we have

$$\begin{pmatrix} 7 & 2 \\ 17 & 25 \end{pmatrix} = \begin{pmatrix} 7 & 8 \\ 11 & 11 \end{pmatrix} \mathbf{K} \bmod 26$$

- The inverse of X can be computed:

$$\begin{pmatrix} 7 & 8 \\ 11 & 11 \end{pmatrix}^{-1} = \begin{pmatrix} 25 & 22 \\ 1 & 23 \end{pmatrix}$$

- So

$$\mathbf{K} = \begin{pmatrix} 25 & 22 \\ 1 & 23 \end{pmatrix} \begin{pmatrix} 7 & 2 \\ 17 & 25 \end{pmatrix} = \begin{pmatrix} 549 & 600 \\ 398 & 577 \end{pmatrix} \bmod 26 = \begin{pmatrix} 3 & 2 \\ 8 & 5 \end{pmatrix}$$

- This result is verified by testing the remaining plaintext-ciphertext pairs.

# Transposition Cipher: Permutation Cipher

- A very different kind of mapping is achieved by performing some sort of permutation on the plaintext letters
- Keep the plaintext characters unchanged, but to alter their positions by rearranging them using a permutation
- It follows that, for every $x \varepsilon X$, there is a unique element $x' \varepsilon X$ such that $\pi(x') = x$
- This allow us to define the inverse permutation $\pi^{-1}: X \rightarrow X$ by the rule

$$\pi^{-1}(x) = x' \qquad \text{iff} \qquad \pi(x') = x$$

- Then $\pi^{-1}$ is also a permutation of X
- **Algorithm**
  - Let m be a positive integer
  - Let $P = C = (Z_{26})^m$ and let K consist of all permutations of {1,…,m}
  - For a key $\pi$, we define

and

$$e_\pi(x_1, \ldots, x_m) = (x_{\pi(1)}, \ldots, x_{\pi(m)})$$

$$d_\pi(y_1, \ldots, y_m) = (y_{\pi^{-1}(1)}, \ldots, y_{\pi^{-1}(m)}),$$

- Where $\pi^{-1}$ is the inverse permutation to $\pi$

# Permutation Cipher: Example

- Suppose m = 6 and the key is the following permutation π:

| $x$ | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| $\pi(x)$ | 3 | 5 | 1 | 6 | 4 | 2 |

- The inverse permutation $\pi^{-1}$ can be constructed by interchanging the two rows, and rearranging the columns so that the first row is in increasing order

| $x$ | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| $\pi^{-1}(x)$ | 3 | 6 | 1 | 5 | 2 | 4 |

- Now suppose, the plaintext is

  `shesellsseashellsbytheseashore`

- First partition the plaintext into groups of six letters:

  `shesel` | `lsseas` | `hellsb` | `ythese` | `ashore`

- Now each group of six letters is rearranged according to the permutation π, yielding the following:

  `EESLSH` | `SALSES` | `LSHBLE` | `HSYEET` | `HRAEOS`

# Permutation Cipher: Example (Cont..)

- So the ciphertext is:

  EESLSHSALSESLSHBLEHSYEETHRAEOS

- The ciphertext can be decrypted in a similar fashion, using tne inverse permutation $\pi^{-1}$
- Permutation cipher is a special case of the Hill Cipher
- Given a permutation $\pi$ of the set {1,…,m}, we can define an associated m x m permutation matrix $K_\pi = (K_{i,j})$ according the formula

$$k_{i,j} = \begin{cases} 1 & \text{if } i = \pi(j) \\ 0 & \text{otherwise.} \end{cases}$$

- A permutation matrix is a matrix in which every row and column contains exactly one "1" and all other values are "0".
- A permutation matrix can be obtained from an identity matrix by permuting rows and columns.

$$K_\pi = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

# Cryptanalysis of Permutation Cipher

- Vulnerable to several kinds of ciphertext-only attack
- **Statistical Attack**
  - Does not change the frequency of letters in the ciphertext, it only reorders the letters
  - The attack that can be applied is single letter frequency analysis
    - Useful if the length of the ciphertext is long enough
  - Do not preserve the frequency of diagrams and trigrams
- **Brute-Force attack**
  - Attacker can try all possible keys to decrypt the message
  - The number of keys can be huge (1!+2!+…,L!), where L is the length of the ciphertext
  - Better approach is to guess the number of columns

# Cryptanalysis of Permutation Cipher

Suppose that Eve has intercepted the ciphertext message "EEMYNTAACTTKONSHITZG". The message length $L = 20$ means the number of columns can be 1, 2, 4, 5, 10, or 20. Eve ignores the first value because it means only one column and no permutation.

a. If the number of columns is 2, the only two permutations are (1, 2) and (2, 1). The first one means there would be no permutation. Eve tries the second one. Eve divides the ciphertext into two-character units: "EE MY NT AA CT TK ON SH IT ZG". She then tries to permute each of these getting "ee ym nt aa tc kt no hs ti gz", which does not make sense.

b. If the number of columns is 4, there are $4! = 24$ permutations. The first one (1 2 3 4) means there would be no permutation. Eve needs to try the rest. After trying all 23 possibilities, Eve finds no plaintext that makes sense.

c. If the number of columns is 5, there are $5! = 120$ permutations. The first one (1 2 3 4 5) means there would be no permutation. Eve needs to try the rest. The permutation (2 5 1 3 4) yields a plaintext "enemyattackstonightz" that makes sense after removing the bogus letter z and adding spaces.

# Cryptanalysis of Permutation Cipher

## Pattern Attack

- The ciphertext created from a keyed transposition cipher has some repeated patterns

**3 8 13 18 1 6 11 16 4 9 14 19 5 10 15 20 2 7 12 17**

The 1st character in the ciphertext comes from the 3rd character in the plaintext. The 2nd character in the ciphertext comes from the 8th character in the plaintext. The 20th character in the ciphertext comes from the 17th character in the plaintext, and so on. There is a pattern in the above list. We have five groups: (3, 8, 13, 18), (1, 6, 11, 16), (4, 9, 14, 19), (5, 10, 15, 20), and (2, 7, 12, 17). In all groups, the difference between the two adjacent numbers is 5. This regularity can be used by the cryptanalyst to break the cipher. If Eve knows or can guess the number of columns (which is 5 in this case), she can organize the ciphertext in groups of four characters. Permuting the groups can provide the clue to finding the plaintext.

# Stream Cipher

- One that encrypts a digital data stream one bit or one byte at a time
- Examples of classical stream ciphers are the Autokeyed Vigenère cipher and the Vernam cipher
- In the ideal case, a one-time pad version of the Vernam cipher would be used, in which the keystream ($k_i$) is as long as the plaintext bit stream ($p_i$)
- The basic idea is to generate a keystream $K = k_1, k_2 \ldots$ and use it to encrypt a plaintext string $X = x_1 x_2 \ldots$ according to the rule

$$Y = y_1 y_2 \ldots = e_{k1}(x1) e_{k2}(x2) \ldots$$

- **These are of two types**
  - **Synchronous**
    - Vernam Cipher: One Pad Cipher and Linear Feedback Shift Cipher
  - **Asynchronous**
    - Vigenere Cipher: Autokey Cipher

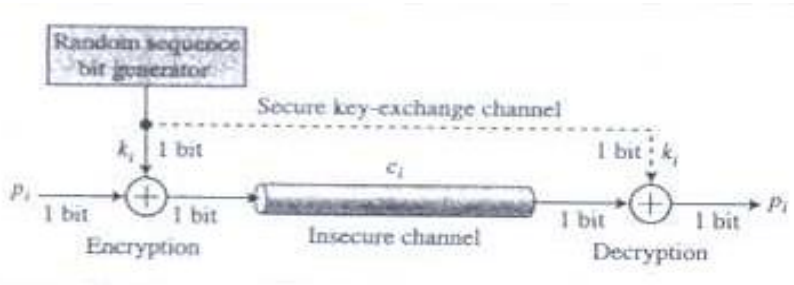
# Synchronous Stream Cipher

- **Synchronous Stream Cipher**
  - The simplest type of stream cipher
  - The key stream is constructed from the key, independent of the plaintext string, using some specified algorithm
  - It is a tuple (P,C,K,L,E,D), together with a function g, such that the following conditions are satisfied:

1. $\mathcal{P}$ is a finite set of possible *plaintexts*
2. $\mathcal{C}$ is a finite set of possible *ciphertexts*
3. $\mathcal{K}$, the *keyspace*, is a finite set of possible *keys*
4. $\mathcal{L}$ is a finite set called the *keystream alphabet*
5. $g$ is the *keystream generator*. $g$ takes a key $K$ as input, and generates an infinite string $z_1 z_2 \cdots$ called the *keystream*, where $z_i \in \mathcal{L}$ for all $i \geq 1$.
6. For each $z \in \mathcal{L}$, there is an *encryption rule* $e_z \in \mathcal{E}$ and a corresponding *decryption rule* $d_z \in \mathcal{D}$. $e_z : \mathcal{P} \rightarrow \mathcal{C}$ and $d_z : \mathcal{C} \rightarrow \mathcal{P}$ are functions such that $d_z(e_z(x)) = x$ for every plaintext element $x \in \mathcal{P}$.

# Vernam Cipher: One Time Pad Cipher

- A one time pad cipher uses a key stream of non repeating character that is randomly chosen for each encipherment
- Once an input cipher text for transposition is used, it is never used again for any other message
- The length of input cipher text is equal to the length of the original plain text
- The encryption and decryption algorithms each use a single EX-OR operation
- In this cipher the EX-OR operation is used one bit at a time




- Get your plaintext and convert it to binary
- Generates a key that is totally random and, in binary, is at least as long as the plaintext
- Produce the ciphertext by applying bitwise XOR on the plaintext and the key

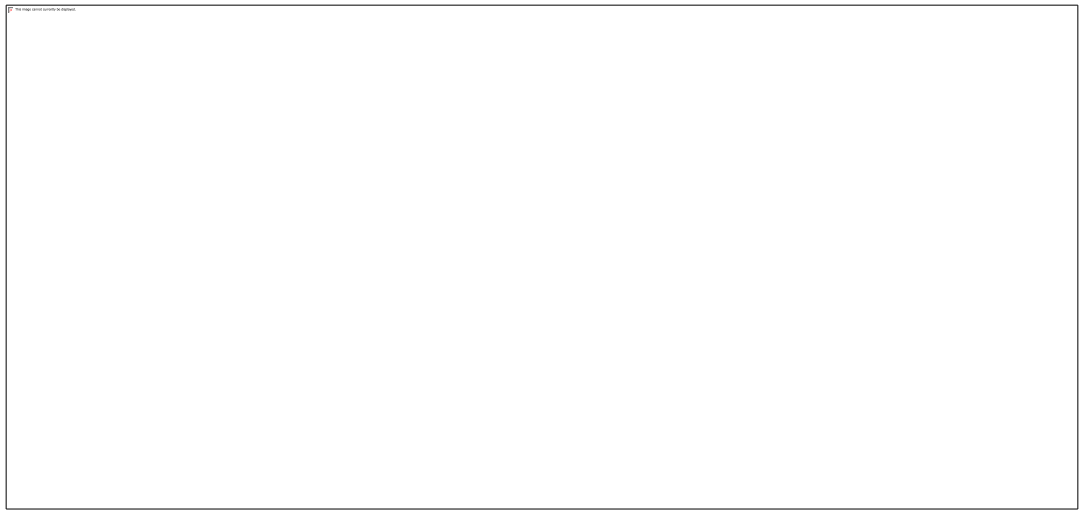
# One Time Pad Cipher: Example

- **Plaintext : Hi!** 1001000 1101001 0100001
- **Key: 0l;@** XOR 0110000 1101100 0111011
- **Ciphertext: x□ □ @** 1111000 0000101 0011010
- **Algorithm**
  1. Treat each plaintext alphabet as a number in an increasing sequence i.e. A=0,B=1,….,Z=25
  2. Do the same for each character of the input ciphertext
  3. Add each number corresponding to the plaintext alphabet to the corresponding input ciphertext alphabet number
  4. If the sum thus produced is greater than 26, subtract 26 from it
  5. Translate each number of the sum back to the corresponding alphabet. This gives the output ciphertext

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1. Plain text | H<br>7 | O<br>14 | W<br>22 | A<br>0 | R<br>17 | E<br>4 | Y<br>24 | O<br>14 | U<br>20 |
| + | | | | | | | | | |
| 2. One-time pad | 13<br>N | 2<br>C | 1<br>B | 19<br>T | 25<br>Z | 16<br>Q | 0<br>A | 17<br>R | 23<br>X |
| 3. Initial Total | 20 | 16 | 23 | 19 | 42 | 20 | 24 | [illegible] | [illegible] |
| 4. Subtract 26, if > 25 | 20 | 16 | 23 | 19 | 16 | 20 | 24 | 5 | 17 |
| 5. Cipher text | U | Q | X | T | Q | U | Y | F | R |

# Linear Feedback Shift Register Cipher

- Made of a shift register and a feedback function



- Shift register is a sequence of m cells, $b_0$ to $b_{m-1}$,
- Each cell holds a single bit
- Cells are initialized to an m-bit word, called the initial value or the seed
- Whenever an output bit is needed every bit is shifted one cell to the right
- **Linear Feedback Shift Register**
- $b_m$ is a linear function of $b_0$ to $b_{m-1}$

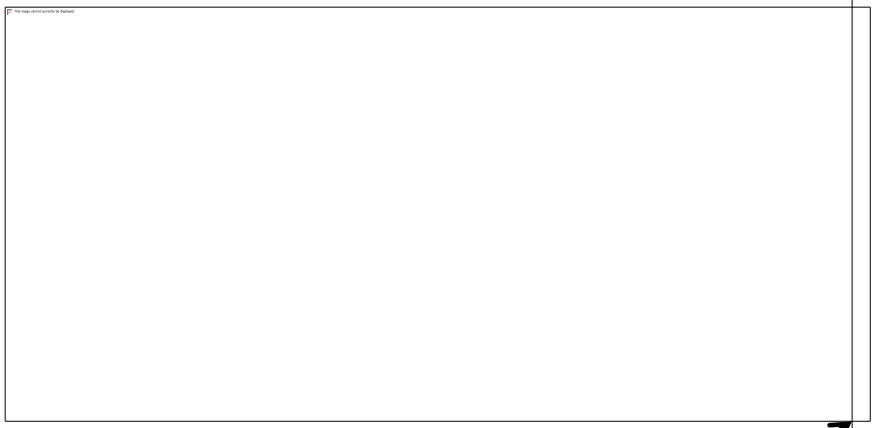
$$b_m = c_{m-1}\ b_m + \ldots\ldots + c_2\ b_2 + c_1\ b_1 + c_0\ b_0 (c_0 \neq 0)$$

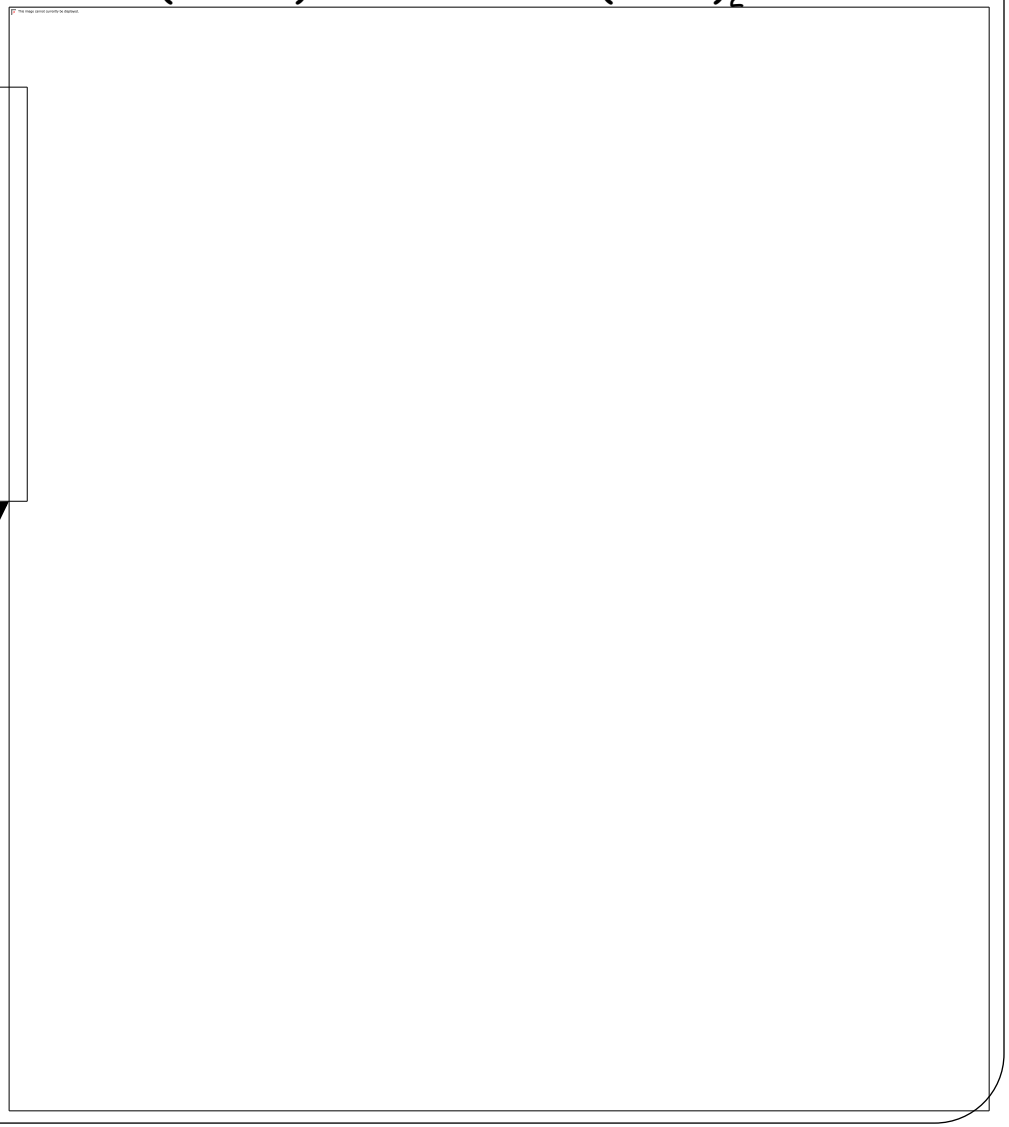
- The addition operation is also the EX-OR operations

$$b_m = c_{m-1}\ b_m \oplus \ldots\ldots \oplus c_2\ b_2 \oplus c_1\ b_1 \oplus c_0\ b_0 (c_0 \neq 0)$$

# Linear Feedback Shift Register Cipher

- Create a linear feedback shift register with 4 cells in which $b_4 = b_1 \oplus b_0$.
- show the value of output for 20 transactions (shifts) if the seed is $(0001)_2$
- Use of LFSR in encryption





Shows the values of the key stream

# Linear Feedback Shift Register Cipher

- The key stream generated from a LFSR is a pseudorandom sequence in which the sequence is repeated after N bits
- The maximum period of an LFSR is $2^m-1$

# Cryptanalysis of LFSR Cipher

- Since all operations in this crypto system are linear, the cryptosystem is vulnerable to a known plaintext attack
- Suppose attacker has a plaintext string $x_1x_2...x_n$ and the corresponding ciphertext string $y_1y_2...y_n$, then attacker can compute the keystream bits as

$$k_i = (x_i + y_i) \bmod 2,\ 1 \leq i \leq n$$

- Suppose attacker also knows the value of m, then attacker needs only to compute$c_0....c_{m-1}$ in order to be able to reconstruct the entire keystream

# Cryptanalysis of LFSR Cipher: Example

Suppose Oscar obtains the ciphertext string

$$101101011110010$$

corresponding to the plaintext string

$$011001111111000.$$

Then he can compute the keystream bits:

$$110100100001010.$$

Suppose also that Oscar knows that the keystream was generated using a 5-stage LFSR. Then he would solve the following matrix equation, which is obtained from the first 10 keystream bits:

$$(0,1,0,0,0) = (c_0,c_1,c_2,c_3,c_4)\begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}.$$

It can be verified that

$$\begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}^{-1} = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{pmatrix},$$

by checking that the product of the two matrices, computed modulo 2, is the identity matrix. This yields

$$\begin{aligned}(c_0,c_1,c_2,c_3,c_4) &= (0,1,0,0,0)\begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{pmatrix} \\ &= (1,0,0,1,0).\end{aligned}$$

# Asynchronous Stream Cipher: Vigenere Cipher: Autokey Cipher

- The key is a stream of subkeys
- Each subkey is used to encrypt the corresponding character in the plaintext
- The name of the cipher implies that the subkeys are automatically created from the plaintext cipher character during the encryption process
- Let $P = C = K = L = Z_{26}$

  Let $z_1 = K$, and define $z_i = x_{i-1}$ for all $i \geq 2$

  For $0 \leq z \leq 25$, define

  $e_z(x) = (x+z) \bmod 26$ and

  $d_z(y) = (y-z) \bmod 26$, where $(x,y) \in Z_{26}$

# Autokey Cipher: Example

- Suppose the key is **K = 12**
- Plaintext: **Attack is today**
- **Encryption:**

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Plaintext:** | **a** | **t** | **t** | **a** | **c** | **k** | **i** | **s** | **T** | **o** | **d** | **a** | **y** |
| **P's Value:** | 0 | 19 | 19 | 0 | 2 | 10 | 8 | 18 | 19 | 14 | 3 | 0 | 24 |
| **Key Stream** | 12 | 0 | 19 | 19 | 0 | 2 | 10 | 8 | 18 | 19 | 14 | 3 | 0 |
| **C's Values** | 12 | 19 | 12 | 19 | 2 | 12 | 18 | 0 | 11 | 7 | 17 | 3 | 24 |
| **Ciphertext** | **M** | **T** | **M** | **T** | **C** | **M** | **S** | **A** | **L** | **H** | **R** | **D** | **Y** |

- **Decryption:**

$d_{12}(12) = (12-12) \bmod 26 = 0 = a$

$d_{0}(19) = (19-0) \bmod 26 = 19 = t$

$d_{19}(12) = (12-19) \bmod 26 = 19 = t$

and so on…