# AN TOÀN & BẢO MẬT THÔNG TIN

Giảng viên: Ths Phạm Thanh Bình

Bộ môn Kỹ thuật máy tính & mạng

*http://dhthuyloi.blogspot.com*

# Chương 2:

## *MẬT MÃ ĐỐI XỨNG*

- Những vấn đề cơ bản của mật mã
- Các kỹ thuật mã hoá cổ điển
- Chuẩn mã hoá dữ liệu DES
- Chuẩn mã hoá cải tiến AES

# Bài 2.1 - Những vấn đề cơ bản của mật mã

- Mật mã là công cụ cơ bản để bảo vệ sự bí mật và toàn vẹn của dữ liệu
- Mật mã có thể biến đổi dữ liệu từ dạng ban đầu sang một dạng khác khó đọc hơn, nhằm bảo vệ sự bí mật của dữ liệu.
- Một số kỹ thuật mật mã có thể giúp chứng minh được nguồn gốc và sự toàn vẹn của dữ liệu
- Mật mã cũng được ứng dụng trong các giao thức mạng, nhằm bảo vệ dữ liệu khi lưu thông trên mạng

# *The Imitation Game*

Film received eight nominations at the 87th Oscar

## *Bài tập 1:*

*- Lập trình nhập một chuỗi kí tự từ bàn phím, cộng mỗi phần tử của chuỗi với 3. Hiện chuỗi mới ra màn hình.*

*- Lập trình khôi phục lại chuỗi ban đầu.*

## *Bài tập 2:*

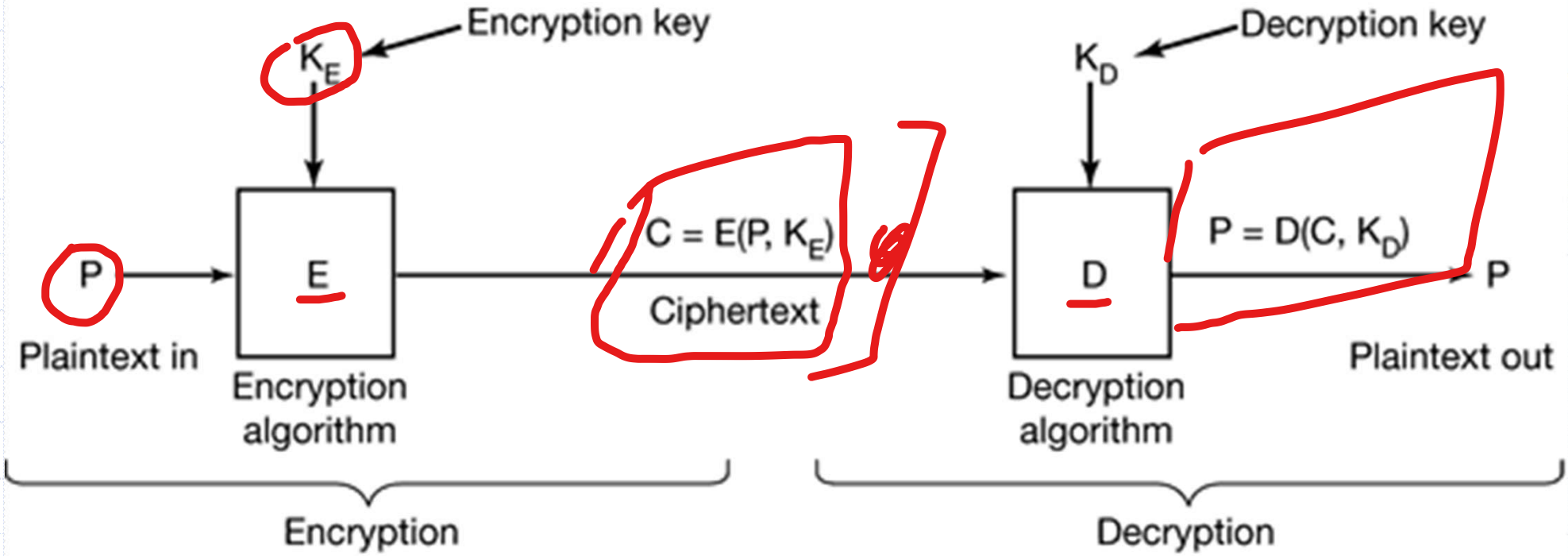
*- Lập trình nhập một chuỗi kí tự từ bàn phím, cộng mỗi phần tử của chuỗi với K (K là một giá trị nhập từ bàn phím). Hiện chuỗi mới ra màn hình.*

*- Lập trình khôi phục lại chuỗi ban đầu.*

***Muốn khôi phục được chuỗi ban đầu thì cần phải biết K.***

- *Trong kĩ thuật mật mã, K được gọi là "Khoá" (Key) - dùng để mã hoá và giải mã.*
- *Chuỗi ban đầu được gọi là "bản rõ" (Plain text)*
- *Chuỗi sau khi mã hoá được gọi là "bản mã" (Cipher text)*
- *Cách thức biến bản rõ thành bản mã được gọi là "thuật toán mã hoá".*
- *Cách thức biến bản mã thành bản rõ được gọi là "thuật toán giải mã".*

# Quá trình mã hoá và giải mã



# Trong đó:

- $P$ là bản rõ
- $K_E$ là khoá mã hoá *(Encryption Key)*
- $C$ là bản mã
- $E$ là thuật toán mã hoá (hay hàm mã hoá)
- $C = E(P, K_E)$ là công thức định nghĩa sự mã hoá.

*Bản mã được tạo ra nhờ áp dụng thuật toán mã hoá $E$, tác động vào bản rõ $P$, với khoá mã hoá $K_E$ làm tham số.*

- $K_D$ là khoá giải mã *(Decryption Key)*
- $D$ là thuật toán giải mã (hay hàm giải mã)
- $P = D(C, K_D)$ là công thức định nghĩa sự giải mã.

  *Thuật toán $D$ sẽ tác động lên bản mã $C$ với $K_D$ là tham số, nó sẽ biến bản mã $C$ trở về dạng ban đầu là bản rõ $P$.*

- Nếu $K_D = K_E$ (tức là khoá giải mã và khoá mã hoá giống nhau), ta gọi đây là "*mật mã đối xứng*". Khoá này phải được giữ bí mật.
- Ngược lại, nếu $K_D \neq K_E$, đây là "*mật mã bất đối xứng*".

# Bài 2.2 - Các kỹ thuật mã hoá cổ điển

- Kỹ thuật thay thế
- Kỹ thuật chuyển dịch - hoán vị

# Kỹ thuật thay thế

- Thuật toán mã hoá sẽ thay thế mỗi kí tự trong bản rõ bằng một kí tự khác
- Một số mật mã tiêu biểu:
  - Mật mã CAESAR
  - Mật mã Affine
  - Mật mã Monoalphabetic
  - Mật mã Polyalphabetic…

# Mật mã CAESAR

- Mỗi kí tự trong bảng chữ cái được thay thế bởi một kí tự khác cùng bảng, cách sau nó ba vị trí
- Đây là một trong những mật mã ra đời sớm nhất, do Julius Caesar phát minh.

# Quy tắc thay thế:

- Plaintext:

a b c d e f g h i j k l m n o p q r s t u v w x y z

- Ciphertext:

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

# Ví dụ:

- Plaintext:

  `meet me after the toga party`

- Ciphertext:

  `PHHW PH DIWHU WKH WRJD SDUWB`

- Để tiện cho việc tính toán, ta sẽ gán mỗi ký tự với một số nguyên tương ứng:

| 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 |

- Như vậy, mọi phép toán mã hoá và giải mã đều được thực hiện trong một tập hợp 26 số nguyên từ 0 đến 25 (gọi là "Vành 26" hay $Z_{26}$).
- Công thức mã hoá:

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

- Công thức giải mã:

$$P = D(C, 3) = (C - 3) \bmod 26$$

# Ví dụ:

***Mã hoá:***

- Với $P$ = ‘A’ = 0 thì $C = (0 + 3) \bmod 26$
  - ⇨ $C = 3 \bmod 26 = 3$ = ‘D’
- Với $P$ = ‘X’ = 23 thì $C = (23+3) \bmod 26$
  - ⇨ $C = 26 \bmod 26 = 0$ = ‘A’

## *Giải mã:*

- Với $C$ = 'A' = 0 thì $P = (0 - 3) \bmod 26$

  ⇨ $P = -3 \bmod 26 = -3$

- Chú ý: Đối với a thuộc $Z_{26}$ thì **a + 26 = a**

  ⇨ $P = -3 + 26 = 23$ = 'X'

- Tổng quát hoá mật mã Caesar bằng cách thay thế một kí tự ban đầu bởi kí tự đứng sau nó $K$ vị trí.
- Công thức mã hoá:

$$C = E(P, K) = (P + K) \bmod 26$$

- Công thức giải mã:

$$P = D(C, K) = (C - K) \bmod 26$$

# Ví dụ: với $K$=6

***Mã hoá:***

- Với $P$ = 'X' = 23 thì $C = (23+6) \bmod 26$

  ⇨ $C = 29 \bmod 26 = 3$ = 'D'

***Giải mã:***

- Với $C$ = 'D' = 3 thì $P = (3 - 6) \bmod 26$

  ⇨ $P = -3 + 26 = 23$ = 'X'

# Ưu nhược điểm:

- Mật mã Caesar đơn giản, dễ thực hiện
- Độ an toàn không cao, dễ bị bẻ khoá bởi tấn công Brute-force do số lượng khoá quá ít (chỉ có 25 khoá)

*(Brute-force là hình thức tấn công bằng cách thử tất cả các khả năng của khoá để tìm ra khoá đúng)*

## *Bài tập 1:*

*- Lập trình nhập một chuỗi kí tự từ bàn phím, mã hoá chuỗi bằng thuật toán CAESAR tổng quát với khoá K nhập từ bàn phím. Hiện chuỗi mới ra màn hình.*

*- Lập trình giải mã để khôi phục lại chuỗi ban đầu.*

## *Bài tập 2:*

*Lập trình bẻ khoá mật mã Caesar bằng phương pháp Brute-force.*

*- Đầu vào chương trình là chuỗi kí tự cipher text thu được từ Bài tập 1.*

*- Hãy xác định khoá K đã sử dụng và nội dung của plain text ban đầu.*

# Mật mã Affine

- Kí tự $P$ ban đầu được thay thế bởi kí tự $C$ theo công thức:

$$C = E(P, \{a,b\}) = (aP + b) \bmod 26$$

- Công thức giải mã:

$$P = D(C, \{a,b\}) = a^{-1}(C - b) \bmod 26$$

Trong đó khoá $K$ chính là cặp 2 số nguyên $\{a,b\}$ *thuộc* $Z_{26}$

- *Nếu* $a = 1$ *thì mật mã Affine sẽ trở thành mật mã Caesar tổng quát*

# Điều kiện:

- Để có thể giải mã được thì a phải nguyên tố với 26, tức là ước số chung lớn nhất của a và 26 bằng 1: ƯSCLN(a, 26) = 1
- Vì chỉ khi ƯSCLN(a, 26) = 1 thì mới tồn tại $a^{-1} \in Z_{26}$ để tính P
- $a^{-1}$ là một số thuộc $Z_{26}$ thoả mãn:

$$a.a^{-1} = a^{-1}.a = 1 \text{ (trong } Z_{26})$$

# Bài tập:

- Xác định các giá trị có thể có của a trong $Z_{26}$, và tính $a^{-1}$ tương ứng

# Đáp số:

- Các giá trị của a:

  1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, 25

- Các $a^{-1}$ tương ứng:

  1, 9, 21, 15, 3, 19, 7, 23, 11, 5, 17, 25

## *Bài tập 1:*

*- Lập trình nhập một chuỗi kí tự từ bàn phím, mã hoá chuỗi bằng thuật toán Affine với cặp số {a,b} nhập từ bàn phím. Hiện chuỗi mới ra màn hình.*

## *Bài tập 2:*

*Lập trình giải mã Affine để khôi phục lại chuỗi ban đầu:*

*- Đầu vào chương trình là cặp số* $\{a,b\}$ *và chuỗi kí tự cipher text từ Bài tập 1.*

*- Đầu ra chương trình là chuỗi kí tự plain text*

# Bẻ khoá mật mã Affine

- Có tất cả bao nhiêu cặp số nguyên $\{a,b\}$ trong $Z_{26}$ có thể dùng làm khoá của mật mã Affine?

- Có 12 giá trị của a
- Có 26 giá trị của b
- Tổng cộng có 12 x 26 = 312 cặp số {a,b}

# Nhận xét:

- Mật mã Affine có độ phức tạp lớn hơn mật mã Caesar tổng quát, số lượng khoá cũng nhiều hơn
- Độ an toàn chưa cao, dễ bị phá bởi tấn công Brute-force do số lượng khoá chưa nhiều (chỉ có 312 khoá)

## *Bài tập 3:*

*Lập trình bẻ khoá mật mã Affine bằng phương pháp Brute-force:*

*- Đầu vào chương trình là chuỗi kí tự cipher text thu được từ Bài tập 1.*

*- Hãy xác định cặp số {a,b} đã sử dụng và nội dung của plain text ban đầu.*

# Mật mã Monoalphabetic

- Mỗi kí tự trong bảng chữ cái được thay thế bởi một kí tự bất kì khác cùng bảng
- Trong ví dụ sau, các kí tự *A* được thay bằng *Q*, *B* được thay bằng *W*, *C* được thay bằng *E*…

- Bản rõ:

  ABCDEFGHIJKLMNOPQRSTUVWXYZ
- Bản mã:

  QWERTYUIOPASDFGHJKLZXCVBNM

## *Bài tập 1:*

*- Lập trình nhập một chuỗi kí tự từ bàn phím, mã hoá chuỗi bằng thuật toán Monoalphabetic với khoá K nhập từ bàn phím (Khoá K là một chuỗi gồm 26 chữ cái có trật tự bất kì). Hiện chuỗi mới ra màn hình.*

- ***Bài tập 2:***

  *Lập trình giải mã Monoalphabetic để khôi phục lại chuỗi ban đầu:*

  *- Đầu vào chương trình là khoá K và chuỗi kí tự cipher text từ Bài tập 1.*

  *- Đầu ra chương trình là chuỗi kí tự plain text*

# Bẻ khoá mật mã Monoalphabetic

- Có tất cả bao nhiêu chuỗi bất kì gồm 26 chữ cái có thể dùng làm khoá của mật mã Monoalphabetic?

- Số lượng hoán vị của chuỗi dài 26 chữ cái là:

  $26! \approx 4.10^{26}$ (trên $4.10^{26}$ khoá!)

- Giả sử một máy tính cá nhân tốc độ 4 GHz (thực hiện 4 tỷ phép tính/1 giây) có thể kiểm tra được 1 khoá trong 1 phép tính.
- Để kiểm tra hết $4.10^{26}$ khoá sẽ cần tới:

$$\frac{4.10^{26}}{4.10^{9}.3600.24.365} \approx 3 \text{ tỉ năm!}$$

# Nhận xét:

- Mật mã Monoalphabetic có số lượng khoá rất lớn, khó bẻ khoá bằng phương pháp Brute -force
- Tuy nhiên vẫn có thể bẻ khoá mật mã này dựa trên các thống kê về các đặc điểm tự nhiên của ngôn ngữ

# Ví dụ:

- Hãy giải mã thông điệp ***tiếng Anh*** dưới đây (được mã hoá bởi phương pháp Monoalphabetic):

YIFQFMZRWQFYVECFMDZPCVMRZWNMD
ZVEJBTXCDDUMJNDIFEFMDZCDMQZKCE
YFCJMYRNCWJCSZREXCHZUNMXZNZUCD
RJXYYSMRTMEYIFZWDYVZVYFZUMRZCR
WNZDJJXZWGCHSMRNMDHNCMEQCHZ
JMXJZWIEJYUCFWDJNZDIR

# Bảng thống kê xác suất xuất hiện của các kí tự trong tiếng Anh

| Kí tự | Xác suất | Kí tự | Xác suất | Kí tự | Xác suất |
|---|---|---|---|---|---|
| A | 0.082 | J | 0.002 | S | 0.063 |
| B | 0.015 | K | 0.008 | T | 0.091 |
| C | 0.028 | L | 0.040 | U | 0.028 |
| D | 0.043 | M | 0.024 | V | 0.010 |
| E | 0.127 | N | 0.067 | W | 0.023 |
| F | 0.022 | O | 0.075 | X | 0.001 |
| G | 0.020 | P | 0.019 | Y | 0.020 |
| H | 0.061 | Q | 0.001 | Z | 0.001 |
| I | 0.070 | R | 0.060 | | |

*Có thể chia 26 chữ cái thành 5 nhóm sau:*

- E có xác suất cao nhất: 0.127
- T, A, O, I, N, S, H, R có xác suất từ 0.060 đến 0.090
- D và L có xác suất khoảng 0.04
- C, U, M, W, F, G, Y, P, B có xác suất từ 0.015 đến 0.028.
- V, K, J, X, Q, Z có xác suất dưới 0.01

- *Ba mươi cặp kí tự tiếng Anh có xác suất xuất hiện cao nhất (từ cao xuống thấp) là:*

TH, HE, IN, ER, AN, RE, ED, ON, ES, ST, EN, AT, TO, NT, HA, ND, OU, EA, NG, AS, OR, TI, IS, ET, IT, AR, TE, SE, HI, OF

- *Mười hai bộ 3 kí tự tiếng Anh có xác suất xuất hiện cao nhất (từ cao xuống thấp) là:*

THE, ING, AND, HER, ERE, ENT, THA, NTH, WAS, ETH, FOR, DTH

# Tần suất xuất hiện các chữ cái trong thông điệp đã cho:

| Chữ cái | Tần suất | Chữ cái | Tần suất | Chữ cái | Tần suất |
|---|---|---|---|---|---|
| A | 0 | J | 11 | S | 3 |
| B | 1 | K | 1 | T | 2 |
| C | 15 | L | 0 | U | 5 |
| D | 13 | M | 16 | V | 5 |
| E | 7 | N | 9 | W | 8 |
| F | 11 | O | 0 | X | 6 |
| G | 1 | P | 1 | Y | 10 |
| H | 4 | Q | 4 | Z | 20 |
| I | 5 | R | 10 | | |

# Phân tích:

- Z xuất hiện nhiều nhất: có thể Z là **e**
- C, D, F, J, M, R, Y xuất hiện trên 10 lần: có thể ứng với **t, a, o, i, n, s, h, r**

*Xét các cặp kí tự có thể chứa e, chúng có dạng Z- hoặc -Z:*

- DZ, ZW xuất hiện 4 lần
- NZ, ZV xuất hiện 3 lần
- RZ, HZ, XZ, FZ, ZR, ZV, ZC, ZD, ZJ xuất hiện 2 lần

- Vì ZW xuất hiện 4 lần, WZ không xuất hiện, và W xuất hiện ít, nên có thể W là **d**, và ZW là **ed**
- Vì DZ xuất hiện 4 lần, ZD xuất hiện 2 lần, nên D có thể là một trong 3 kí tự **r**, **s**, **t**

- Vì bộ ba ZRW, RZW xuất hiện ở đầu bản mã, và cặp RW còn xuất hiện sau đó, và R xuất hiện khá nhiều trong bản mã, nên có thể R là **n**
- Như vậy ZRW có thể là **end**, RZW là **ned**, RW là **nd**

- Vì cặp NZ hay xuất hiện, còn cặp ZN thì không, nên có thể N là **h**, NZ là **he**
- Vì M xuất hiện nhiều thứ 2 sau Z, bộ ba RNM có dạng nh-, nên M có thể là một nguyên âm (**a, o,** hoặc **i**)

- Bộ ba NMD xuất hiện 2 lần
- M có thể là **a, o, i**
- D có thể là **r**, **s**, **t**

⇒ Vậy NMD có thể là his hoặc has

(với dự đoán D là s, M là i hoặc a)

- Cụm NCMF có dạng h-i- hoặc h-a- gợi ý F là một phụ âm (**t, n, r** )
- Khá hợp lý nếu NCMF là **hair**, hay M là **i** và F là **r**, suy ra C là **a**?
- Như vậy cụm HNCMF có thể là **chair**, hay H là **c**

- Lúc trước ta dự đoán: C, D, F, J, M, R, Y xuất hiện trên 10 lần: có thể ứng với **t, a, o, i, n, s, h, r**
- Vậy **o** có thể là Y hoặc J.
- Sau khi thay thế vào văn bản ta thấy Y là **o** hợp lý hơn, còn lại J là **t**:

**o-r-r iend-ro--ari se-a -inedh i se--t- --ass -i t**
YIFQFMZRWQFYVECFMDZPCVMRZWNMDZVEJBTXCDDUMJ
**h s-r-r iseas i–e-a-oration hadta-en--ace-h i -e**
NDIFEFMDZCDMQZKCEYFCJMYRNCWJCSZREXCHZUNMXZ
**he-asnt-oo- in-i -o-redso-e -ore-ineandhesett**
NZUCDRJXYYSMRTMEYIFZWDYVZVYFZUMRZCRWNZDZJJ
**-ed-ac- inh ischa ir-aceti-ted-- to-ardsthes-n**
XZWGCHSMRNMDHNCMFQCHZJMXJZWIEJYUCFWDJNZDIR

- Tiếp tục suy luận ta thu được bản rõ:

*"Our friend from Paris examined his empty glass with surprise, as if evaporation had taken place while he wasn't looking. I poured some more wine and he settled back in his chair, face tilted up towards the sun"*

# Bài tập:

- Lập trình nhập một chuỗi kí tự từ bàn phím. Đếm xem kí tự đầu tiên của chuỗi xuất hiện bao nhiêu lần trong chuỗi?

# Bài tập:

- Lập trình tính tần suất xuất hiện của các kí tự trong một đoạn văn bản cho trước.

# Bài tập:

- Hãy giải mã thông điệp ***tiếng Anh*** dưới đây (được mã hoá bởi phương pháp Monoalphabetic):

UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFP
ESXUDBMETSXAIZVUEPHZHMDZSHZOW
SFPAPPDTSVPQUZWYMXUZUHSXEPYEPOP
DZSZUFPOMBZWPFUPZHMDJUDTMOHMQ

# Gợi ý: Tần suất xuất hiện các chữ cái trong thông điệp đã cho (%):

| | | | | |
|---|---|---|---|---|
| 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 | | | | |

# Nhận xét

- Mật mã Monoalphabetic dễ bẻ vì không che giấu được tần xuất suất hiện của các kí tự trong văn bản
- Để khắc phục có thể áp dụng mã hoá đa kí tự, hoặc sử dụng nhiều bảng mã thay thế (Polyalphabetic)…

# *Hết Phần 2_1*