1. Classical Cipher Techniques

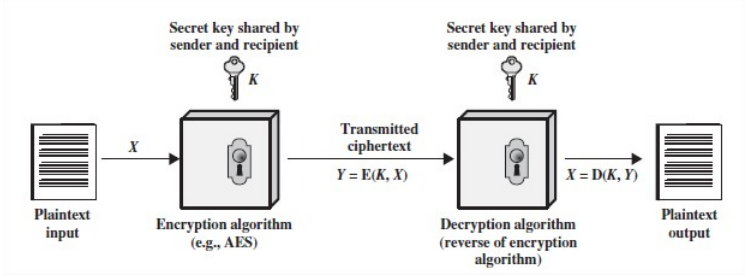
1.1 TỔNG QUAN

Trong **Mật mã học**, xét về mặt thời gian người ta thường chia thành 2 dạng mã hóa chính là mã hóa cổ điển và mã hóa hiện đại. Mã hóa cổ điển đã xuất hiện từ lâu trong lịch sử loài người nhưng ngày nay ít còn được sử dụng phổ biến bởi những hạn chế của nó, chủ yếu sử dụng trong các trò chơi tìm khóa hay những ứng dụng không yêu cầu bảo mật cao. Mã hóa cổ điển thường có 2 loại chính:

1. Phương pháp mã hóa **thay thế**: từng kí tự (hay nhóm kí tự) trong plaintext (bản rõ) được thay thế bằng một kí tự (hay một nhóm kí tự) khác để tạo ra ciphertext (bản mã). Bên nhận chỉ cần thay thế ngược lại trên bản mã để có được bản rõ ban đầu.
2. Phương pháp mã hóa **hoán vị**: các kí tự trong bản rõ vẫn được giữ nguyên, chúng chỉ được sắp xếp lại vị trí để tạo ra bản mã, hay các kí tự trong bản rõ hoàn toàn không bị thay đổi bằng kí tự khác mà chỉ đổi chỗ của chúng để tạo thành bản mã.

Khi nghiên cứu, người ta chia các thuật toán mã hóa thành 2 loại chính:

1. Mã hóa **đối xứng** *(mã hóa khóa bí mật - Symmetric Cipher)*: Sử dụng cùng 1 khóa bởi người gửi (cho việc mã hóa) và người nhận (cho việc giải mã).
2. Mã hóa **bất đối xứng** *(mã hóa khóa công khai - Asymmetric Cipher)*: Sử dụng 2 khóa khác nhau, 1 khóa công khai (để mã hóa thông điệp và công khai cho người khác biết) và một khóa riêng (để giải mã thông điệp chỉ có người sở hữu biết).



Hình 1.1: Mô hình mã hóa đối xứng cơ bản

Hầu hết các loại mã hóa cổ điển đều thuộc loại mã hóa đối xứng. *Bài thực hành này sẽ tập trung vào việc phân tích và tìm hiểu một số thuật toán mã hóa cổ điển nổi bật để sinh viên có cái nhìn tổng quan về Mật mã học.*

1.1.1 Kiến thức nền tảng

Để thực hành tốt, sinh viên cần nắm được các kiến thức cơ bản về:
- Các khái niệm cơ bản liên quan đến mật mã, mã hóa.
- Các thuật toán mã hóa cổ điển cơ bản: Ceasar, Monoalphabetic, Playfair, Hill, ...
- Cách sử dụng CrypTool và kiến thức lập trình cơ bản.

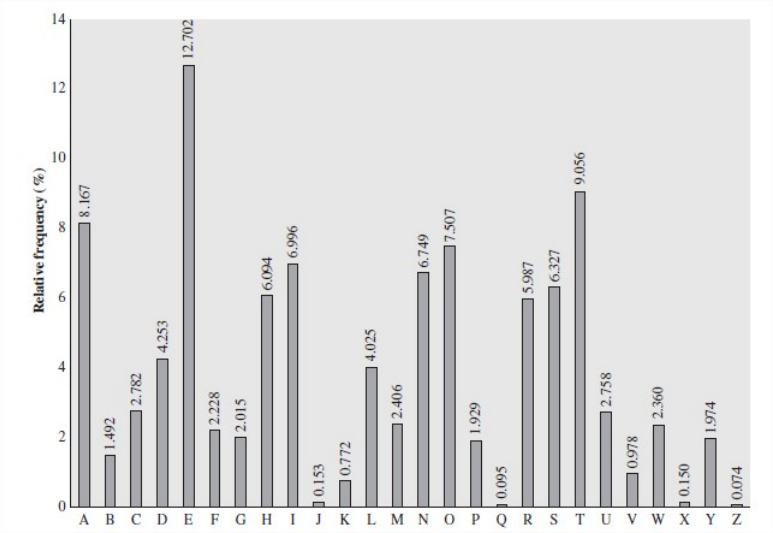
a. Mã hóa Caesar

Giới thiệu 1.1.1 Nguyên tắc của mã hóa Caesar là văn bản mã được tạo ra bằng cách thay thế mỗi chữ cái trong văn bản với một chữ cái cách nó một đoạn cho trước trong bảng chữ cái.

■ **Ví dụ 1.1** Thực hiện phép dịch chuyển 3 ký tự
ABCDEFGHIJKLMNOPQRSTUVWXYZ *(Bảng chữ cái bình thường)*
DEFGHIJKLMNOPQRSTUVWXYZABC *(Bảng mã dịch chuyển với k=3)*
Lúc này, mã hóa từ *MAT MA HOC ANTN* ta sẽ được *PDW PD KRF DQWQ*. ■

b. Mã hóa Monoalphabetic (đơn bảng)

Giới thiệu 1.1.2 Đây là dạng tổng quát hóa của Caesar, không dịch chuyển k vị trí trong bảng chữ cái mà thay bằng một hoán vị bất kỳ của 26 chữ cái. Mỗi hoán vị xem là 1 khóa. Việc phá mã có thể dựa vào dự đoán trên tần suất xuất hiện của các ký tự trong bảng chữ cái Tiếng Anh.



Hình 1.2: Tần suất xuất hiện của các chữ cái Tiếng Anh

c. Mã hóa Polyalphabetic (đa bảng) - mã hóa Vigenère

Giới thiệu 1.1.3 Với loại mã hóa đa bảng, Vigenère là đại diện tiêu biểu. Mỗi dòng của bảng Vigenère là một mã hóa đơn bảng. Để mã hóa cần sử dụng khóa có chiều dài bằng plaintext bằng cách lặp đi lặp lại khóa đó. Lần lượt đối chiếu cột của từng ký tự trong plaintext với từng dòng ký tự tương ứng trong khóa (hay ngược lại) để tìm ra ciphertext.

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

Hình 1.3: Bảng mã Vigenère

■ **Ví dụ 1.2** Key là LEMON, plaintext là ATTACKATDAWN. Sử dụng mã hóa Vigenère, ta được kết quả như sau:
Plaintext: ATTACKATDAWN
Key: LEMONLEMONLE
Ciphertext: LXFOPVEFRNHR ■

d. Mã hóa Playfair

Giới thiệu 1.1.4 Xây dựng ma trận 5x5 từ khóa bằng cách đặt khóa vào đầu ma trận, sau đó điền các chữ cái còn thiếu vào phần còn lại. Kí tự I xem như ký tự J và các ký tự trong ma trận không trùng nhau. Nguyên tắc mã hóa Playfair như sau:[a]

- Mã hóa từng cặp 2 ký tự liên tiếp nhau. Nếu dư 1 ký tự, thêm ký tự "x" vào cuối.
- Nếu 2 ký tự nằm cùng dòng, thay thế bằng 2 ký tự tương ứng bên phải. Ký tự ở cột cuối cùng được thay bằng ký tự ở cột đầu tiên.
- Nếu 2 ký tự nằm cùng cột được thay thế bằng 2 ký tự bên dưới. Ký tự ở hàng cuối cùng được thay thế bằng ký tự ở hàng trên cùng
- Nếu 2 ký tự lập thành hình chữ nhật được thay thế bằng 2 ký tự tương ứng trên cùng dòng ở hai góc còn lại.

[a] Ví dụ về Playfair: `https://en.wikipedia.org/wiki/Playfair_cipher`

Sinh viên tự tìm hiểu các loại mã hóa cổ điển khác.
Có thể tham khảo thêm tại `http://crypto.interactive-maths.com`

1.1.2 Môi trường - Công cụ

1. Máy tính Windows và IDE với ngôn ngữ lập trình tùy ý (C#, Java, Python,...)
2. Phần mềm **CrypTool 1.4** hoặc **CrypTool 2**
   CrypTool là công cụ mã nguồn mở mạnh mẽ trên Windows phục vụ cho việc học tập và nghiên cứu với nhiều thuật toán mã hóa phổ biến. CrypTool được cung cấp miễn phí tại `https://www.cryptool.org` (sử dụng phiên bản mới nhất).

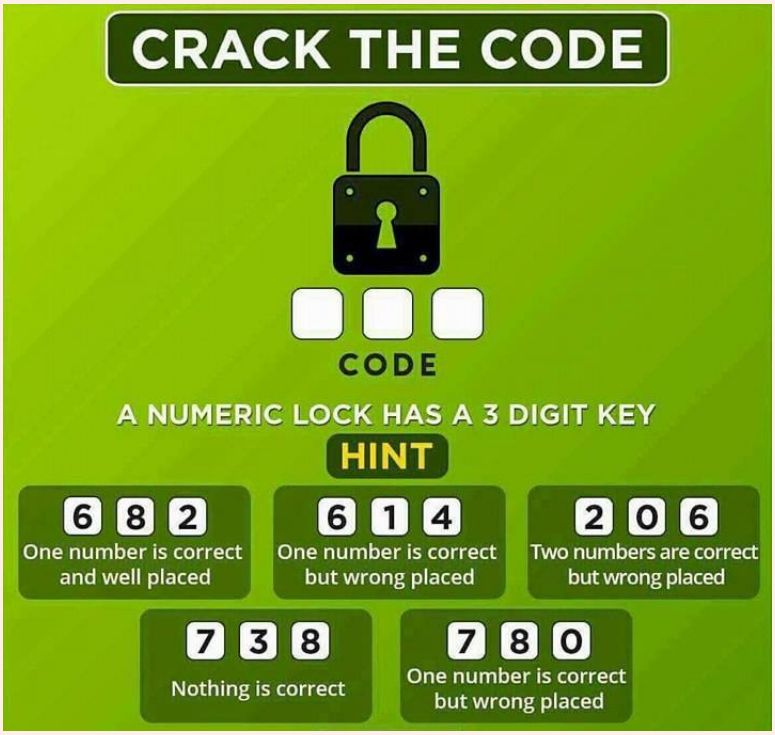
1.2 NỘI DUNG THỰC HÀNH

Lab 1 thực hiện trong 1 buổi thực hành với thời lượng 5 tiết/buổi theo kế hoạch 6 buổi thực hành dành cho lớp ANTT/ANTN.

1.2.1 Mở đầu

Task 1.1 Để mở đầu về mật mã học, sinh viên tiếp cận và đưa ra lời giải cho 2 bài tập sau với việc mã hóa thông tin cơ bản, không sử dụng thuật toán mã hóa sau đây:

1. Hãy tìm 3 chữ số (code) theo các gợi ý dưới đây:



Hình 1.4: Hãy tìm ra 3 chữ số với các dữ kiện đã cho sẵn.

2. Hãy tìm mã hóa tương ứng của các số từ 1 đến 9
   - Mỗi biểu tượng trong số 9 biểu tượng xuất hiện trong bảng dưới đây (△◁▷◯♡♠♢♣●) mã hóa duy nhất một trong các chữ số 1 đến 9.
   - Cột ngoài cùng bên phải là các tổng số ở mỗi hàng
   - Hàng dưới cùng cho các tổng số ở mỗi cột.
   - Một dấu **?** có thể đại diện cho bất kỳ một hoặc hai chữ số và không nhất thiết phải cùng một số trong mỗi trường hợp.

△	△	◁	◯	?
♡	♡	♠	♡	♢♢
?	?	◁	♣	●●
?	♡	♠	♡	●▷
●♡	♢♢	●●	●♢	

Bảng 1.1: Tìm mã hóa của các số từ 1 đến 9

1.2.2 Mã hóa Caesar

Task 1.2 Hãy viết 1 ứng dụng có thể mã hóa và giải mã sử dụng mã hóa Ceasar. Ngôn ngữ lập trình tự chọn và ứng dụng có các chức năng sau:
- Nhập plaintext để mã hóa hoặc ciphertext để giải mã dựa vào khóa k tương ứng.
- Hỗ trợ brute-force (vét cạn) tất cả trường hợp để tìm plaintext khi chỉ cung cấp ciphertext.
Kiểm tra mã hóa, giải mã Caesar với một đoạn plaintext khoảng 100 ký tự bằng ứng dụng vừa xây dựng. Sau đó kiểm tra việc mã hóa, giải mã Caesar với cùng plaintext trên CrypTool và trình bày cách kiểm tra. ■

Gợi ý 1.1. *Có thể mỗi ký tự tương ứng với 1 số từ 0 đến 25. Sử dụng công thức mã hóa cơ bản của Caesar*

$$C = E(k,p) = (p+k)\ mod\ 26 \tag{1.1}$$

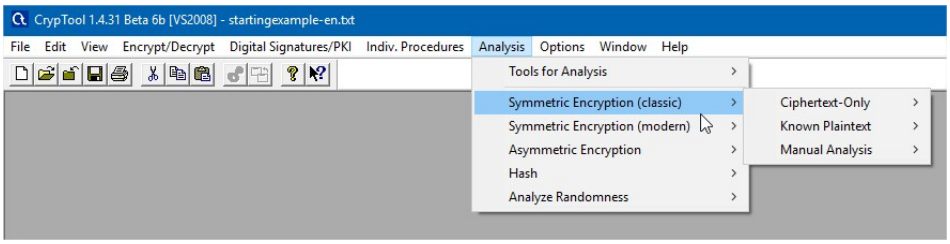
Công thức giải mã:

$$p = D(k,C) = (C-k)\ mod\ 26 \tag{1.2}$$

với C = Ciphertext, p = plaintext, k là mã dịch chuyển với 1<=k<=25.
Trong CrypTool 1, để thực hiện các mã hóa cổ điển, đầu tiên ta tạo tài liệu mới (Ctrl + N), sau đó vào menu Encrypt/Decrypt » Symmetric (classic) và chọn loại mã hóa tương ứng.

Task 1.3 Cho đoạn ciphertext sau, bằng CrypTool hoặc ứng dụng đã xây dựng hãy tìm plaintext ban đầu và giải thích cách thực hiện. Loại mã hóa này có gì đặc biệt?
Gurer ner gjb xvaqf bs crbcyr va guvf jbeyq: gubfr jub ner ybbxvat sbe n ernfba naq gubfr jub ner svaqvat fhpprff. Gubfr jub ner ybbxvat sbe n ernfba nyjnlf frrxvat gur ernfbaf jul gur jbex vf abg svavfurq. Naq crbcyr jub svaq fhpprff ner nyjnlf ybbxvat sbe ernfbaf jul gur jbex pna or pbzcyrgrq. ■

Gợi ý 1.2. *CrypTool có cung cấp chức năng phân tích mã và so sánh với tần suất hiện các chữ cái trong bảng chữ cái Tiếng Anh cho các loại mã hóa phổ biến tại menu **Analysic**.*



Hình 1.5: Menu Analysic trong CrypTool 1

1.2.3 Mã hóa Monoalphabetic

Task 1.4 Sử dụng CrypTool để tìm plaintext của ciphertext đã được mã hóa thay thế sau:
UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ
Trình bày và giải thích cách thực hiện. ■

Gợi ý 1.3. *Dựa vào tần suất xuất hiện của các ký tự trong ciphertext và tần suất xuất hiện phổ biến của các chữ cái trong bảng chữ cái Tiếng Anh. (Hình 1).*
Trong CrypTool, để dò tìm plaintext được mã hóa bằng cách thay thế các ký tự trong bảng chữ cái, có thể vào menu ***Analysic » Symmetric Encryption (Classic) » Substitution...***

Task 1.5 Sử dụng CrypTool để xác định plaintext của ciphertext được mã hóa thay thế sau:
GSVFMREVIHRGBLURMULINZGRLMGVXSMLOLTBRHZNVNYVILUE
RVGMZNMZGRLMZOFMREVIHRGBSLXSRNRMSXRGBEMFSXNZMWRH
GSVLMOBFMREVIHRGBLUERVGMZNGSZGFMWVIGZPVHRMULINZGR
LMZMWXLNNFMRXZGRLMGVXSMLOLTBIVHVZIXSZMWULXFHVWRMW
VKGSGIZRMRMTGSVFMREVIHRGBSZHGSVBLFMTVHGNZMZTVNVMG
IVHVZIXSZMWGVZXSRMTHGZUULUZMBEMFSXNNVNYVIYIRMTRMT
TIVZGVMGSFHRZHNZOLMTDRGSWBMZNRXZMWXIVZGREVZWEZMGZTVH
Trình bày cách thực hiện. Loại mã hóa này là gì và có gì đặc biệt? ■

Mở rộng 1.1 Giải mã thông điệp **Gold Bug** và giải thích cách thực hiện:
53‡‡†305))6*;4826)4‡.)4‡);806*;48†8¶60))85;;]8*;:‡*8†83(88)5*†;46(;88*96
***?;8)*‡(;485);5*†2:*‡(;4956*2(5*—4)8¶8*;4069285);)6†8)4‡‡;1(‡9;48081;8:8‡**
1;48†85;4)485†528806*81(‡9;48;(88;4 (‡?34;48)4‡;161;:188;‡?;
Biết rằng:
– Thông điệp được mã hóa bằng tiếng Anh và không có dấu chấm câu, khoảng trắng được mã hóa.
– Mỗi ký hiệu tương ứng với một chữ cái trong bảng chữ cái tiếng Anh

1.2.4 Mã hóa Polyalphabetic - Vigenère

Task 1.6 Xây dựng ứng dụng mã hóa và giải mã Vigenère (tự chọn ngôn ngữ lập trình) Kiểm tra mã hóa, giải mã Vigenère với một đoạn plaintext khoảng 50 ký tự với 1 khóa từ 10-20 ký tự, sau đó kiểm tra lại bằng CrypTool. Giải thích cách hoạt động của ứng dụng để tìm ciphertext. ■

Gợi ý 1.4. *Sử dụng công thức mã hóa của Vigenère:*
Công thức mã hóa của ký tự thứ i:

$$C_i = (p_i + k_{i \bmod m}) \ mod \ 26 \tag{1.3}$$

Công thức giải mã của ký tự thứ i:

$$p_i = (C_i - k_{i \bmod m}) \ mod \ 26 \tag{1.4}$$

với C = Ciphertext, p = plaintext, k là khóa, m là số ký tự của khóa.

1.2.5 Mã hóa Playfair

Mở rộng 1.2 Xây dựng ứng dụng mã hóa và giải mã Playfair (tự chọn ngôn ngữ lập trình). Ứng dụng có thể đáp ứng:
- Nhập từ khóa, xuất bảng (ma trận 5x5) các chữ cái kèm từ khóa dùng để mã hóa tương ứng
- Kiểm tra mã hóa, giải mã Playfair với một đoạn plaintext khoảng từ 100 ký tự, sau đó kiểm tra lại bằng CrypTool. Giải thích cách hoạt động của ứng dụng để tìm ciphertext.

Task 1.7 Sử dụng ma trận Playfair 1.5 bên dưới để mã hóa thông điệp
Must see you over Cadogan West. Coming at once
Ghi chú: Thông điệp trích từ *Sherlock Holmes, The Adventure of the Bruce-Partington Plans* ■

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

Bảng 1.2: Ma trận Playfair 1.5

Task 1.8 Khi tàu Mỹ PT-109 do đại úy hải quân John F. Kennedy bị đánh chìm bởi một tàu khu trục Nhật vào tháng 8/1943, một thông điệp đã nhận được tại một đài vô tuyến điện của người Úc bằng mã hóa Playfair như sau:
KXJEY UREBE ZWEHE WRYTU HEYFS
KREHE GOYFI WTTTU OLKSY CAJPO
BOTEI ZONTX BYBNT GONEY CUZWR
GDSON SXBOU YWRHE BAAHY USEDQ
Khóa sử dụng là *royal new zealand navy*. Hãy giải mã thông điệp và trình bày cách thực hiện. ■

1.2.6 Các mã hóa khác

Task 1.9 Hãy xác định giá trị *flag* của thông điệp ASCII đã được mã hóa sau và trình bày cách thực hiện, biết trước *flag* là câu chỉ dẫn bằng tiếng Pháp:
4C6520666C6167206465206365206368616C6C656E6765206573743A2
0326163333736343831616535343663643638396435623931323735643332346 5 ■

Gợi ý 1.5. *Sinh viên có thể tham khảo bảng mã ASCII đầy đủ tại* `http://www.ascii-code.com`

Task 1.10 Cho 1 ảnh JPG chứa một thông điệp Flag đã được mã hóa XOR.
Sinh viên download tại `https://goo.gl/NqSNNu`.
Hãy tìm Flag và key đã được sử dụng để mã hóa, trình bày cách thực hiện.
Biết rằng, key để mã hóa có độ dài = 6. ■

Gợi ý 1.6. *Có thể sử dụng CrypTool để phân tích, giải mã XOR và tìm flag.*

Mở rộng 1.3 Trình bày về 1 - 3 loại thuật toán mã hóa cổ điển khác và cho ví dụ minh họa (có thể sử dụng ***CrypTool***, các công cụ mã hóa hoặc tự xây dựng ứng dụng mã hóa/giải mã tương ứng).

1.3 YÊU CẦU - ĐÁNH GIÁ

1.3.1 Yêu cầu

Sinh viên thực hiện đầy đủ các nội dung trong phần thực hành theo nhóm (tối đa **2** sinh viên/nhóm) hoặc cá nhân *(đăng ký từ buổi 1)* và báo cáo kết quả thực hiện như sau:

- File báo cáo .pdf kết quả thực hiện các nội dung thực hành (theo mẫu).
- Đối với các ứng dụng, đặt vào thư mục có tên tương ứng với Task *(source code và file thực thi)*

Báo cáo trình bày trực quan việc thực hành có kèm hình ảnh và chú thích rõ ràng theo cách tiếp cận của bản thân, không sao chép lẫn nhau. Nếu có tham khảo cần ghi nguồn.

R Thời gian hoàn thành Lab 1:
Trong tối đa **10** ngày từ buổi thực hành tại lớp.

Lưu ý 1.3.1 Đặt tất cả báo cáo và thư mục liên quan vào 1 tập tin nén (.zip,.rar) có tên:

[MSSV1]-[Tên SV1]_[MSSV2]-[Tên SV2]_MMH_LabX

Ví dụ: 16520901-Nhut_16521516-Lam_MMH_Lab1. Kích thước tối đa: 10MB.
Nộp báo cáo theo thời gian quy định tại website môn học.

1.3.2 Đánh giá

- Sinh viên hiểu và hoàn thành tốt nội dung thực hành, đúng hạn: 80%
- Sinh viên trình bày cụ thể, rõ ràng kết quả thực hiện: 5-10%
- Khuyến khích sinh viên có tìm hiểu, thực hiện các nội dung mở rộng: 15-20% hoặc cộng trực tiếp vào điểm tổng kết thực hành nếu hoàn thành xuất sắc. *Nội dung mở rộng khuyến nghị thực hiện với sinh viên lớp ANTN.*
- Sao chép bài của sinh viên khác, tham khảo không ghi nguồn: -50% đến -100%.
- Nộp bài trễ: -20% mỗi ngày nộp trễ.

Lưu ý 1.3.2 Báo cáo nộp trễ, sao chép sẽ được xử lý tùy theo mức độ. Các nội dung báo cáo có thể được vấn đáp để đánh giá kết quả tại buổi thực hành tiếp theo, sinh viên vắng thực hành không có lý do thì nhóm sinh viên tương ứng sẽ được trừ tối thiểu 30% điểm bài thực hành đó.

1.4 TÀI LIỆU THAM KHẢO

[1] W. Stallings, *Cryptography and network security: Principles and practice, 6th ed*. Boston, MA, United States: Prentice Hall, 2013. *Chapter 2. Classical Encryption Techniques*
[2] *ASCII code - the extended ASCII table* 2005. [Online]. Available: `www.ascii-code.com`.
[3] Bernhard Esslinger, *Learning and Experiencing Cryptography with CrypTool and SageMath, 12th ed*, CrypTool Project. Available: `https://www.cryptool.org/en/ctp-documentation`.

TÀI LIỆU THỰC HÀNH MẠNG MÁY TÍNH - AN TOÀN THÔNG TIN - UIT
Biên soạn - Tổng hợp: KS. Nguyễn Thanh Hòa

Tài liệu được tổng hợp, tham khảo và biên soạn phục vụ cho việc hướng dẫn thực hành các môn học tại bộ môn An toàn thông tin - Khoa Mạng máy tính và truyền thông.
Tài liệu thực hành này chỉ được sử dụng và cung cấp nội bộ tại website môn học trường ĐH Công nghệ Thông tin `https://courses.uit.edu.vn` cho các sinh viên tham gia các khóa học tương ứng. Tài liệu thực hành này chỉ sử dụng và **lưu hành nội bộ**, phục vụ cho việc học tập, nghiên cứu, không sử dụng và chia sẻ với các mục đích khác khi chưa có sự cho phép của Giảng viên.
Email liên hệ GV khi cần thiết: *hoant@uit.edu.vn*

Biên soạn Tháng 1/2017
Cập nhật lần 1 - Tháng 2/2018

LƯU HÀNH NỘI BỘ