Sécurité des Systèmes d'Information
Exercise sheet 1 : Classical Cryptography

25 Septembre 2019

All answers should be given with detailed justifications.

Exercise 1: Modular arithmetic

1. For $n = 1, ..., 10$ draw :
   - The addition table of $(\mathbb{Z}_n, +)$.
   - The multiplication table of $(\mathbb{Z}_n, \cdot)$.
   - The group of invertible elements $\mathbb{Z}_n^*$
   - Compute the Euler's Totient function $\phi(n)$
2. Compute :
   - $[4 + 5]_7$
   - $[11]_7 - [17]_7$
   - $[11]_7 \cdot [17]_7$
   - $[21 \cdot 27 \cdot 41]_8$
   - $[-44]_3$

Exercise 2: Caesar cipher

The Caesar cipher consists in shifting all caracters of a message by a same number (called the key). For example, a key of 3 replaces A with D, B with E, etc... The permutation is cyclic, when we reach the end of the alphabet, we loop back to the beginning (with the key of 3, X becomes A, Y becomes B, Z becomes C).

1. *Theory*
   - Code the message "BONJOUR" with the key of 4.

- Thinkinf of the alphabet as consisting only of the 26 letters from A to Z, how many different keys do exist ? And by considering all caracters from ASCII 8 bits ?
- Describe an algorithm able to break this encoding.

2. *Programming*
   - Write a Python function caesar(message, key), implementing this encryption for a given key on the standard English alphabet (a,b,c, ...).
   - Write a Python function un_caesar(message, key) that breaks this encoding.

Exercise 3 : Monoalpabetic Substitution

The monoalphabetic substitution consists in assigning to every character a new character. For instance, the key MONALPHBETIQUCDFGJKRSVWXYZ replaces every A with an M, every B with an O, every C with an N, ...
For the sake of simplicity, we often choose a sentence or a text as a key : we take the letters as they appear in the text to build the key, removing the letters that already appeared before and adding the ones missing in the end to finish the key.
We easily see that the key generated before was built from the word "MONOALPHABETIQUE".

1. Theory
   - The message "BGYDKCNGSDMAIBHSJJAFRI" was encoded using the phrase "substitutionmonoalphabetique". Give the complete key, and indicate the original message.
   - Is it hard to break this code ? Describe an algorithm able to break it. (You may be interested to learn about frequency analysis)
2. Programming
   - Write two Python functions :
     create_key(sentence) that creates a substitution key from a given sentence
     monoalphabetic_substitution(message, key) that implements the encryption.
   - *(Bonus)* implement a decoding function for a given message.

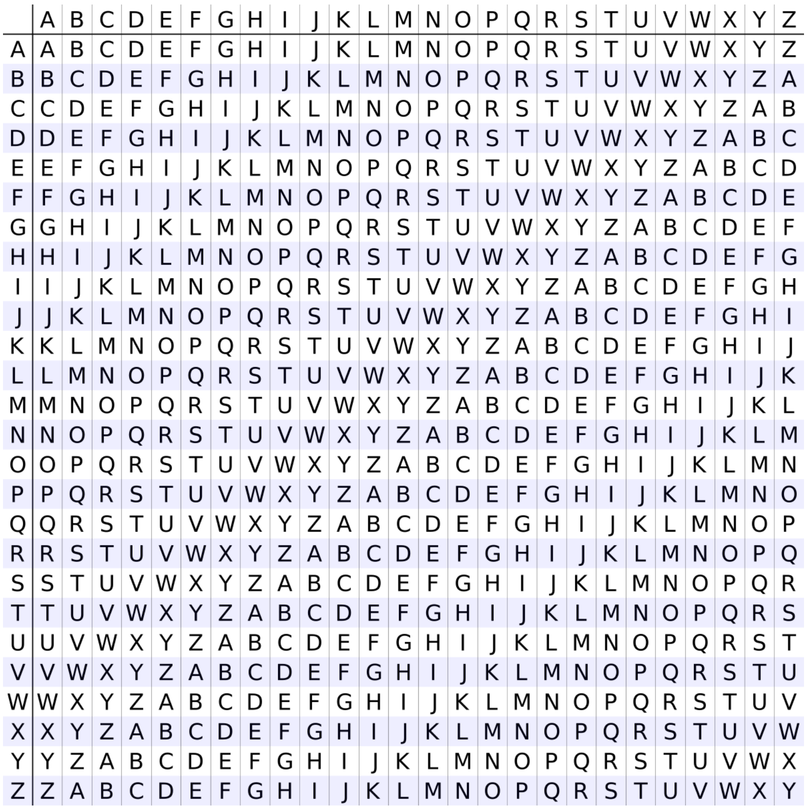
Exercice 4 : Vigenère Algorithm

We now turn to polyalphabetic substitutions, with the Vigenère cipher. First we turn each letter into a number : A is 0, B is 1 and so on up to the character Z which is 25. We then cipher the message by adding to the character value, the corresponding character value at the same position of a chosen key. The addition is made in $\mathbb{Z}_26$. We thus obtain the encrypted character value. The key is a chain of characters that we repeat as much as necessary to reach the same length as the length of the message.

For instance, the message BONJOUR, encrypted with the key BAC gives the message COPKOWS ($B + B = C$, $O + A = O$, $N + C = P$, and we loop back from the beginning of the key, $J + B = K$, ...)

Figure 1 shows the Vigenère table to sum the caracters.

- Encode the message ”JESUISUNPOKEMON” with the key ”ABRA”.
- Deocode the message ”JFJUITLNDSVSSFLR”, knowing the key is again ”ABRA”.
- Knowing that the key is much shorter than the message. Is it hard to break this code ?
- What happens if the key has the same length as the code ?



	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

Figure 1: Vigenère Table