



---

# RP01

## TPX : Lorem ipsum

---

NOM Prénom

NOM Prénom  
(Groupe XX)

11 avril 2021

Ce document synthétise le travail et la réflexion sur le TD de TD01

# Table des matières

<b>1</b>	<b>Une section</b>	<b>1</b>
1.1	Une sous-section . . . . .	1
1.2	Une autre sous-section . . . . .	1
1.2.1	Une sous-sous-section . . . . .	1
1.2.2	Une autre sous-sous-section . . . . .	1
	<b>Une section non numérotée</b>	<b>2</b>
<b>2</b>	<b>Du code</b>	<b>3</b>
2.1	Bouts de codes . . . . .	3
2.1.1	Un plus gros bout de code! . . . . .	3
2.2	Une autre exemple d'inclusion de code . . . . .	4
<b>3</b>	<b>Dit comme ça...</b>	<b>5</b>
3.1	Phénomènes d'induction . . . . .	5
3.1.1	Loi de Lentz . . . . .	5
3.1.2	Théorème de Gauss . . . . .	6
3.2	Vous avez dit potentiel? . . . . .	6
3.3	Des bras et des kets . . . . .	6
3.4	Une matrice . . . . .	6
3.5	Une autre sous-section . . . . .	7

Il ne faut pas respirer de la compote  
ça fait tousser.

---

Kadoc

# 1 Une section

## 1.1 Une sous-section

Une liste non ordonnée :

- Niveau 1 - USB
  - Niveau 2 - Ethernet
    - \* Un élément de niveau 3 - IP
      - Un élément de niveau 4 - TCP
      - Un second élément de niveau 4 - UDP
  - Retour au niveau deux - STP
- Un autre élément de niveau 1 - CSMA/CA

## 1.2 Une autre sous-section

### 1.2.1 Une sous-sous-section

Un excellent professeur proclama un jour :

Il fait trop chaud pour faire du réseau.

A l'**extrême gauche** on a :

Coucou comment ça va ?

Tandis qu'à l'extrême droite on a le <sup>1</sup>RN et aussi cette mise en forme :

Vous ne trouvez pas que petit, on a tous voulu changer la société avant que ce soit elle  
qui nous change ?

### 1.2.2 Une autre sous-sous-section

#### Un paragraphe

Une citation c'est bien, mais bien citer c'est mieux :

---

1. Rassemblement National

*Mais, vous savez, moi je ne crois pas qu'il y ait de bonne ou de mauvaise situation. Moi, si je devais résumer ma vie aujourd'hui avec vous, je dirais que c'est d'abord des rencontres, des gens qui m'ont tendu la main, peut-être à un moment où je ne pouvais pas, où j'étais seul chez moi. Et c'est assez curieux de se dire que les hasards, les rencontres forment une destinée... Parce que quand on a le goût de la chose, quand on a le goût de la chose bien faite, le beau geste, parfois on ne trouve pas l'interlocuteur en face, je dirais, le miroir qui vous aide à avancer. Alors ce n'est pas mon cas, comme je le disais là, puisque moi au contraire, j'ai pu ; et je dis merci à la vie, je lui dis merci, je chante la vie, je danse la vie... Je ne suis qu'amour ! Et finalement, quand beaucoup de gens aujourd'hui me disent : « Mais comment fais-tu pour avoir cette humanité ? » Eh bien je leur réponds très simplement, je leur dis que c'est ce goût de l'amour, ce goût donc qui m'a poussé aujourd'hui à entreprendre une construction mécanique, mais demain, qui sait, peut-être simplement à me mettre au service de la communauté, à faire le don, le don de soi...*

— Otis, *Astérix Mission Cléopâtre*

### Un sous-paragraphe

UN ALLEMAND: [s'esclaffe] Tous les allemands ne sont pas Nazis !

HUBERT BONISSEUR DE LA BATH: Oui, je connais cette théorie

## Une section non numérotée

## 2 Du code

### 2.1 Bouts de codes

Une version humainement lisible d'une fork bombe peut s'écrire ainsi :

```
1 #!/bin/bash
2 fbomb(){
3     fbomb | fbomb &
4 }
5
6 fbomb
```

#### 2.1.1 Un plus gros bout de code !

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 def square_and_multiply(x: int, exponent: int, modulus: int = None, Verbose: bool =
↳ False):
5     """
6     Square and Multiply Algorithm
7
8     x: positive integer
9     exponent: exponent integer
10    modulus: module
11
12    Returns: x**exponent or x**exponent mod modulus when modulus is given
13    """
14    b = bin(exponent).lstrip("0b")
15    r = 1
16    for i in b:
17
18        rBuffer = r
19        r = r ** 2
20
21        if i == "1":
22            r = r * x
23        if modulus:
24            r %= modulus
25
26        if Verbose:
27            print(f"{rBuffer}^2 = {r} mod {modulus}")
28
29    return r
```

## 2.2 Une autre exemple d'inclusion de code

```

1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  import ressources.tools as t
5  import socket
6  import struct
7
8  class TCP:
9      """
10     Represent a TCP packet.
11     Nether case nor order are important.
12     """
13
14     def __init__(self, src_host:str, dst_host:str, src_port:str, dst_port:int,
15         ↪ seq_num:int = 0, ack_num:int = 0, win_size:int = 8192, flags:str =
16         ↪ "001011010", length:int = 612):
17         """
18         Constructor of TCP packet.
19         """
20
21         self.src_host = src_host
22         self.dst_host = dst_host
23         self.src_port = src_port
24         self.dst_port = dst_port
25         self.seq_num = seq_num
26         self.ack_num = ack_num
27         self.win_size = win_size
28         self.flags = int(flags,2) # car 0b001011010 == 90
29         self.length = length
30
31     def build(self) -> bytes:
32         packet = struct.pack(
33             '!HHIIBBHHH',
34             self.src_port, # Source Port
35             self.dst_port, # Destination Port
36             self.seq_num, # Sequence Number
37             self.ack_num, # Acknowledgement Number
38             5 << 4, # Data Offset
39             self.flags, # Flags
40             self.win_size, # Window
41             0, # Checksum (initial value)
42             0 # Urgent pointer
43         )
44
45         pseudo_hdr = struct.pack(
46             '!4s4sHH',
47             socket.inet_aton(self.src_host), # Source Address
48             socket.inet_aton(self.dst_host), # Destination Address
49             socket.IPPROTO_TCP, # PTCL = 6
50             len(packet) + self.length # TCP Length
51         )
52
53         packet += t.payload(self.length) # adding payload

```

```
51 checksum = t.checksum(pseudo_hdr + packet)
52
53 # calculate the checksum and insert it back into the packet
54
55 packet = packet[:16] + struct.pack('H', checksum) + packet[18:]
56
57
58 return packet
```

Code source 1 – Exemple d'un fichier externe labellisé

## 3 Dit comme ça...

### 3.1 Phénomènes d'induction

#### 3.1.1 Loi de Lentz

La **Nature aime la stabilité**. La représentation faite par la Physique d'un système tend toujours à assurer la stabilité en passant d'un état d'équilibre à un autre. Comme par exemple le fait de tordre un bout de métal. On peut croire que rien ne s'est passé mais que nenni ! Il y eu un transfert de chaleur comme **réaction pour restaurer la stabilité**. On comprend plus aisément ce qui va suivre. Quand un courant variable parcourt un circuit, il y a apparition d'un champ qui s'oppose aux variations de courant pour restaurer la stabilité (d'où opposition de phase visible sur oscilloscope).

#### **Théorème 3-1: Loi de Lentz**

La circulation sur un contour fermé du champ électrique agit comme l'opposé de la variation du flux par rapport au temps.

$$\oint_C \vec{E} \cdot d\vec{l} = e = -\frac{d\Phi}{dt}$$

### 3.1.2 Théorème de Gauss

#### Théorème 3-2: Forme globale

Le flux du champ électrique à travers une surface fermée quelconque (que l'on appelle surface de Gauss) est le produit de l'inverse de la perméabilité du vide par la charge algébrique totale.

$$\Phi_E = \frac{1}{\epsilon_0} \iiint_V \rho \, d\tau = \frac{Q_{int}}{\epsilon_0}$$

Forme globale (intégrale) **macroscopique**

Avec  $\rho = \frac{\partial Q}{\partial \tau}$ , la densité volumique de charge.

## 3.2 Vous avez dit potentiel ?

Le potentiel est une grandeur physique qui favorise la naissance d'une force (différence potentiel  $\Rightarrow$  force). On peut comprendre ce concept par la gravitation : Placez un ballon sur un endroit haut d'une pente, une force naîtra et tendra à amener ce ballon vers le bas de la pente. Cette force est née de par la différence de hauteur qui existait. Ici, le potentiel est l'altitude. Et physiquement, on mesure cette différence d'altitude ! (Il va donc de même pour l'électrostatique)

## 3.3 Des bras et des kets

$\langle \varphi | \psi \rangle, \langle \varphi |, | \psi \rangle, | \varphi \rangle \langle \psi |$

Le produit tensoriel de deux qbits donne :

$$|0\rangle \otimes |1\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ 0 \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = |01\rangle \quad (1)$$

## 3.4 Une matrice

	$M$ colonnes	tout plein de bisous
$N$ lignes	$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1M} \\ a_{21} & a_{22} & \cdots & a_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NM} \end{bmatrix}$	$\begin{bmatrix} bisou_1 \\ bisou_2 \\ \vdots \\ bisou_N \end{bmatrix}$



### 3.5 Une autre sous-section

Il est aussi intéressant de bien référencer nos dires. Je veux bien croire que vous êtes très intelligent mais on puise forcément l'eau d'une source. En très bon livre pour apprendre les différentes couches réseaux il y a "Computer Networking A Top-Down Approach" [2]

#### **Coucou**

puis après on a de très bons liens Wikipédia tel que le portail de Cryptologie. [3]

#### **On peut mettre des gifs ?**

Oui mais c'est pas très pratique, faut tout découper dans un fichier séparé etc.

## Références

- [1] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The L<sup>A</sup>T<sub>E</sub>X Companion*. Addison-Wesley, Reading, Massachusetts, 1993.
- [2] James W. Kurose, Keith W. Ross *Computer Networking A Top-Down Approach*. - Pearson, 2021. 7
- [3] Portail de la Cryptologie,  
<https://fr.wikipedia.org/wiki/Portail:Cryptologie>

7

\* \* \*