

# Cycle 1

## Architecture matérielle et initiation à l'algorithmique Informatique

### Cours

#### Chapitre 1 – 01 Architecture matérielle et logicielle et Introduction à l'algorithmique

2 Septembre 2020

##### Savoirs et compétences :

- AA.C01 : Manipuler un OS ou un IDE
- AA.S01 : Se familiariser aux principaux composants d'une machine numérique
- AA.S2 : Se familiariser à la manipulation d'un OS
- AA.S3 : Se familiariser à la manipulation d'un IDE

<b>1</b>	<b>Quelques définitions</b>	<b>2</b>
<b>2</b>	<b>Système d'exploitation – Operating Systems (OS)</b>	<b>2</b>
2.1	Présentation . . . . .	2
2.2	Exemples d'OS . . . . .	2
2.3	Organisation des fichiers . . . . .	3
2.4	Droits d'accès . . . . .	4
2.5	Les programmes . . . . .	4
2.6	Notions de réseau . . . . .	5
<b>3</b>	<b>Applications – Sources et binaires</b>	<b>6</b>
3.1	Application . . . . .	6
3.2	Python . . . . .	6
3.3	À quoi peut servir Python ? . . . . .	6
3.4	Environnement de développement intégré (IDE) . . . . .	6
<b>4</b>	<b>Architecture matérielle d'un ordinateur</b>	<b>8</b>
4.1	Généralités sur le fonctionnement d'un ordinateur . . . . .	8
4.2	Modèle de Von Neumann . . . . .	8
4.3	La carte mère et ses composants . . . . .	9
4.4	Les mémoires . . . . .	10
4.5	Communication avec les périphériques . . . . .	11
4.6	Communication réseau . . . . .	11
4.7	Son et vidéo . . . . .	12



E. Durif – X. Pessoles

M. Rebout

MPSI

## 1 Quelques définitions

**Définition — Informatique.** Le terme informatique désigne littéralement l'automatisation du traitement de l'information par ordinateur. Il s'agit en fait des sciences et techniques de l'information.

**larousse.fr :** science du traitement automatique et rationnel de l'information considérée comme le support des connaissances et des communications.

**fr.wikipedia.org :** l'informatique est un domaine d'activité scientifique, technique et industriel concernant le traitement automatique de l'information par l'exécution de programmes informatiques par des machines : des systèmes embarqués, des ordinateurs, des robots, des automates, etc.

Ces champs d'application peuvent être séparés en deux branches, l'une, de nature théorique, qui concerne la définition de concepts et modèles, et l'autre, de nature pratique, qui s'intéresse aux techniques concrètes de mise en œuvre.

Le terme informatique a été créé en 1962 par Philippe Dreyfus à partir des mots « information » et « automatique ».

**Définition — Algorithmique.** L'algorithmique est l'étude et la production de règles et techniques qui sont impliquées dans la définition et la conception d'algorithmes, c'est-à-dire de processus systématiques de résolution d'un problème permettant de décrire précisément des étapes pour résoudre un problème algorithmique. [fr.wikipedia.org]

Un algorithme est une procédure permettant de résoudre un problème, écrite de façon suffisamment détaillée pour être suivie sans posséder de compétence particulière ni même comprendre le problème que l'on est en train de résoudre.

En informatique un algorithme aura la particularité de permettre de résoudre un type de problème donnée que l'on peut appeler **classe**.

Il possède en entrée des **données** qui représente le problème à résoudre et en revoie en retour un résultat qui correspond à la résolution du problème.

**Définition — Programmation.** Dans le domaine de l'informatique, la programmation, appelée aussi codage, est l'ensemble des activités qui permettent l'écriture des programmes informatiques. C'est une étape importante du développement de logiciels (voire de matériel).

Pour écrire un programme, on utilise un langage de programmation. Un logiciel est un ensemble de programmes (qui peuvent être écrits dans des langages de programmation différents) dédié à la réalisation de certaines tâches par un (ou plusieurs) utilisateurs du logiciel.

La programmation représente donc ici la rédaction du (ou des) code source d'un logiciel. On utilise plutôt le terme développement pour dénoter l'ensemble des activités liées à la création d'un logiciel et des programmes qui le composent. Cela inclut la spécification du logiciel, sa conception, puis son implémentation proprement dite au sens de l'écriture des programmes dans un langage de programmation bien défini et aussi la vérification de sa correction, etc. [fr.wikipedia.org]

### ■ Exemple

- Recettes de cuisine dans la limite de la connaissance d'une certaine technicité.
- Notice d'assemblage d'un meuble à condition de disposer des outils nécessaire.

## 2 Système d'exploitation – Operating Systems (OS)

### 2.1 Présentation

Dans le cadre du modèle de la machine de von Neumann, un **seul programme s'exécute**. Il s'agit donc du système d'exploitation qui est chargé dès le démarrage. Ce programme a pour but de gérer les accès aux processeurs et à la mémoire et aux périphériques, ce qui permet effectivement que l'ordinateur exécute plusieurs programmes à la fois.

Par ailleurs l'OS permet de manière générale de gérer l'organisation des données sur le disque dur ainsi que leur droit d'accès. Il gère aussi les différentes ressources et sert de garde-fou en cas de tentative de mauvaise utilisation des ressources de l'ordinateur.

Pour finir, l'OS permet à un ou plusieurs utilisateurs de s'identifier leur permettant ainsi d'utiliser un seul ordinateur sans nécessairement partager les données et les programmes.

### 2.2 Exemples d'OS

Il existe deux grandes familles d'OS : Windows et UNIX. UNIX est elle-même une grande famille derrière laquelle on peut retrouver beaucoup de distributions : GNU/Linux (Debian, Ubuntu, *etc.*), FreeBSD, MacOS, *etc.* Les différences

pouvant être mises en avant sont les suivantes :

- sous UNIX, il existe des distributions dont les sources sont libres (modifiables par tous) ;
- sous UNIX, l'accès à un shell, permet de réaliser une multitude de tâches en ligne de commande, ce qui est parfois plus efficace qu'avec la souris ;
- sous UNIX, la base de logiciels disponibles est centralisée, ce qui peut faciliter la recherche d'un logiciel et son installation ;
- sous Windows, l'est installations de périphériques est souvent plus aisée.

Les distributions UNIX sont beaucoup plus répandues qu'on peut parfois le penser : un grand nombre de calculateurs et de serveurs sont sous UNIX, les smartphones, tablettes, box internet sont sous UNIX.



Bureau Windows 10



Bureau Ubuntu



Bureau MacOS

## 2.3 Organisation des fichiers

Les données sont organisées par fichiers :

- programmes (et leurs données) :  $\approx 10^5$  fichiers sur mon PC ;
- données :  $\approx 10^6$  fichiers sur mon PC (20 ans de données).

Du point de vue de l'utilisateur les fichiers sont organisés en une structure arborescente :

- chaque fichier est contenu dans un répertoire ;
- ce répertoire, ainsi que d'autres répertoires ou fichiers, peuvent éventuellement être à leur tour contenus dans un autre répertoire ;
- on peut ainsi remonter dans l'arborescence jusqu'à un répertoire appelée répertoire racine.

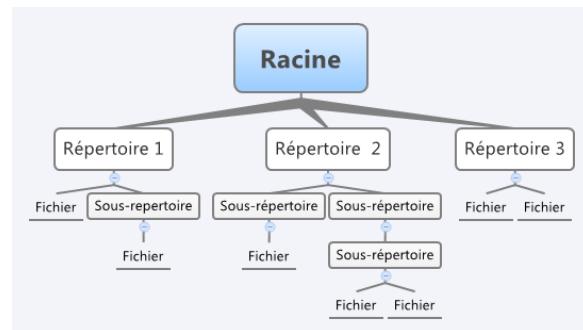


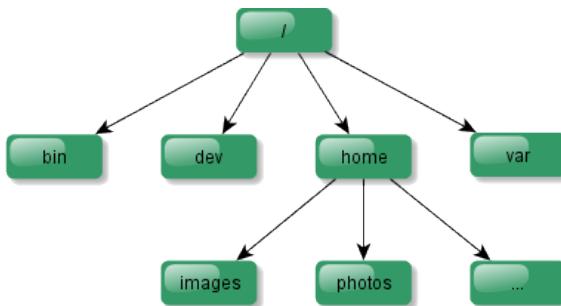
FIGURE 1 – Arborescence des fichiers

**■ Remarque** L'organisation des fichiers est totalement différente entre un environnement Windows et un environnement UNIX. En première approximation, sous Windows, les fichiers sont triés d'une part par applications. D'autre part, les fichiers utilisateurs sont stockés dans le dossier C :\Users.

### 2.3.1 Unix

Sous UNIX, les fichiers sont triés par type de fichiers puis par applications.

- Arborescence unique : unifie l'accès aux différents disques (locaux, distants, disque dur, clé USB, etc.)
- Répertoire racine noté /
- $/a/b$  : fichier ou sous-répertoire  $b$  du sous-répertoire  $a$  de la racine. Notation appelée chemin absolu du fichier ou répertoire  $b$ .
- $a/b$  : fichier ou sous-répertoire  $b$  d'un sous-répertoire  $a$  (d'emplacement non précisé), c'est donc un chemin relatif. Par exemple, si le répertoire courant est /home/adent/ et qu'une commande fait référence à Documents/trombi.pdf, le fichier concerné se trouve au chemin absolu /home/adent/Documents/trombi.pdf !



Organisation des dossiers sous UNIX

Quelques dossiers courants sous UNIX :

- *boot* : fichiers permettant le démarrage de Linux;
- *etc* : fichiers de configuration;
- *home* : répertoires personnels des utilisateurs. On en a déjà parlé un peu avant : c'est dans ce dossier que vous placerez vos fichiers personnels, à la manière du dossier Mes documents de Windows.

### 2.3.2 MS Windows

- Chaque disque a un nom formé d'une lettre suivie de « : » (A:, ..., Z:), souvent source de problème (pas de convention standard).
- On utilise \ au lieu de / pour noter les sous-répertoires.

### 2.3.3 Métadonnées

À chaque fichier/répertoire sont associées des informations, appelées métadonnées et notamment :

- taille du fichier;
- propriétaire;
- droits d'accès (lecture, écriture, exécution) pour le propriétaire, les membres de son groupe, les autres utilisateurs
- date de dernière modification, dernier accès.

Ces métadonnées sont utilisées par le système pour accorder/refuser l'accès, pour savoir ce qui doit être sauvegardé, etc.

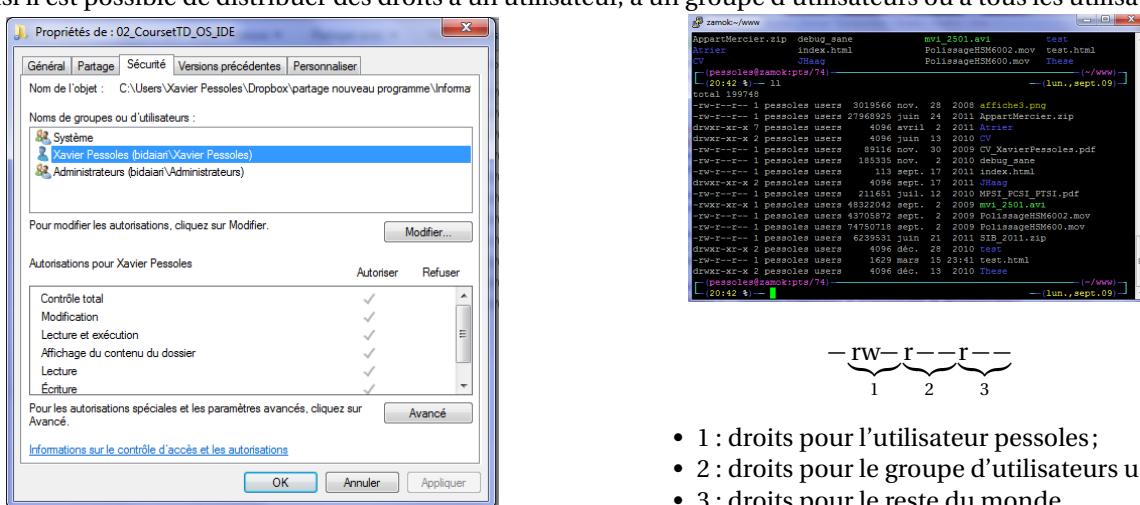
## 2.4 Droits d'accès

Les droits d'accès font partie des propriétés d'un fichier. Sous UNIX, on peut recenser 3 principaux droits pour un fichier :

- le droit en lecture (*r – read*), qui permet à l'utilisateur de lire le contenu du fichier;
- le droit en écriture (*w – write*), qui permet à l'utilisateur d'écrire dans le fichier;
- le droit d'exécution (*x – execute*), qui permet à l'utilisateur d'exécuter le fichier.

Par ailleurs pour un même ordinateur, il peut exister plusieurs utilisateurs. Ces utilisateurs peuvent de plus appartenir à un groupe.

Ainsi il est possible de distribuer des droits à un utilisateur, à un groupe d'utilisateurs ou à tous les utilisateurs.



- 1 : droits pour l'utilisateur pessoles;
- 2 : droits pour le groupe d'utilisateurs user;
- 3 : droits pour le reste du monde.

## 2.5 Les programmes

**Propriété** Les programmes existent essentiellement sous deux formes :

- **les programmes en langage machine** qui contiennent le code des instructions du processeur qu'il faut mettre en mémoire pour exécuter le programme;
- **les scripts** qui sont des programmes sous forme de texte dans un langage humainement compréhensible. Techniquement, le système va lancer un programme appelé interpréteur et lui donner ce texte. L'interpréteur va alors exécuter le programme.

Programmes en langage machine	Scripts (sous Unix)
<ul style="list-style-type: none"> <li>Difficiles à écrire directement, spécifiques à une architecture processeur/système d'exploitation donnée.</li> <li>En général produits par un compilateur, programme permettant de transformer un programme écrit dans un langage évolué en langage machine.</li> </ul>	<ul style="list-style-type: none"> <li>Foultitude de langages disponibles (plus de 200 répertoriés sur wikipedia).</li> <li>Pour réaliser un script : <ul style="list-style-type: none"> <li>écrire le texte du programme dans un fichier texte</li> <li>ajouter en première ligne : #! chemin-vers-l'interpréteur (par exemple #! /bin/sh)</li> <li>Donner les droits d'exécution sur le fichier.</li> </ul> </li> </ul>

## 2.6 Notions de réseau

Les communications Internet se sont développées à partir des années 1990 et ont connu une croissance exponentielle. Le point clé pour permettre de nombreuses communications par réseau internet est d'utiliser des serveurs qui dialoguent entre eux et des clients.

### 2.6.1 Client – Serveur

Deux ordinateurs sont nécessaires pour créer un réseau.

**Définition — Serveur.** Un serveur est un ordinateur ou un logiciel permettant de fournir des services à un ou plusieurs ordinateurs clients : serveur de fichiers, serveur de courrier électronique, serveur web...

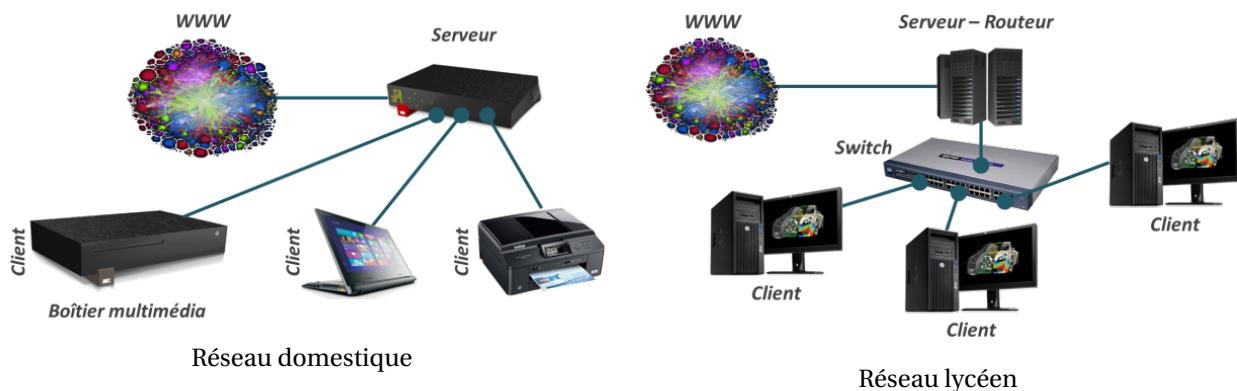
**Définition — Client.** Un client est un ordinateur ou un logiciel qui demande des services à un serveur.

#### ■ Exemple

Service installé sur le serveur	Logiciel installé sur le client utilisant le service
Serveur WEB Apache	Navigateur (Chrome, Firefox, Iceweasel, Safari,...)
Serveur de diffusion streaming VLC serveur	Client lecteur de flux video : VLC Client
Serveur de mail : Postfix	Lecteur de mail : Thunderbird

Pour les utilisateurs, en général, seul le nom du logiciel « client » est connu.

### 2.6.2 Architecture d'un réseau local



Dans le cas d'un réseau domestique, le serveur (box) est en fait relié à un serveur du fournisseur d'accès, lui-même relié à internet. Dans ce cas, le serveur joue le rôle de routeur.

Dans un réseau de lycée (par exemple), le routeur a la particularité de relier un réseau d'ordinateurs à Internet. Il gère aussi la communication entre les ordinateurs du réseau du lycée.

### 3 Applications – Sources et binaires

#### 3.1 Application

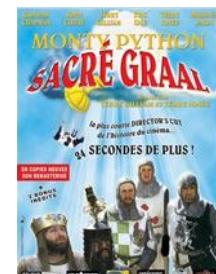
Une application (ou logiciel) est un fichier que l'on peut exécuter afin de satisfaire un besoin (lire un film, ouvre un fichier pdf, naviguer sur internet...). C'est en général un fichier **binnaire**, c'est à dire qu'en général l'utilisateur ne peut pas savoir comment a été codé le logiciel. Il a été produit par l'éditeur de logiciel via un **compilateur**.

Pour savoir comment une application est codée, il est nécessaire d'en connaître le **code source**. Lorsqu'il s'agit d'un logiciel libre, les sources sont disponibles sur Internet. Lorsqu'on travaille chez un éditeur de logiciel, ce code source est aussi accessible. Dans les autres cas, ce code n'est pas accessible. Dans la majorité des cas, l'utilisateur n'a pas besoin de connaître les sources pour utiliser le logiciel.

Parmi les codes compilés les connus on peut nommer le **C** ou le **C++**. Le langage **Python** n'a pas besoin d'être compilé pour être exécuté (en apparence).

#### 3.2 Python

Python est un langage de programmation, dont la première version est sortie en 1991. Créé par Guido van Rossum, il a voyagé du Macintosh de son créateur, qui travaillait à cette époque au Centrum voor Wiskunde en Informatica aux Pays-Bas, jusqu'à se voir associer une organisation à but non lucratif particulièrement dévouée, la Python Software Foundation, créée en 2001. Ce langage a été baptisé ainsi en hommage à la troupe de comiques les « Monty Python ».



#### 3.3 À quoi peut servir Python ?

Python est un langage puissant, relativement facile à apprendre et riche en possibilités. Il est possible d'étendre ses fonctionnalités. On peut aussi appeler des bibliothèques qui aident le développeur à travailler.

Python permet :

- de réaliser de petits programmes très simples, appelés scripts, chargés d'une mission très précise sur votre ordinateur;
- des programmes complets, comme des jeux, des suites bureautiques, des logiciels multimédias, des clients de messagerie...
- des projets très complexes, comme des progiciels (ensemble de plusieurs logiciels pouvant fonctionner ensemble, principalement utilisés dans le monde professionnel).

Voici quelques-unes des fonctionnalités offertes par Python et ses bibliothèques :

- créer des interfaces graphiques;
- faire circuler des informations au travers d'un réseau;
- dialoguer d'une façon avancée avec votre système d'exploitation;
- ...

Python est un langage de programmation interprété, c'est-à-dire comme on l'a spécifié plus tôt, que les instructions que vous lui envoyez sont « transcris » en langage machine au fur et à mesure de leur lecture. D'autres langages (comme le C / C++) sont appelés « langages compilés » car, avant de pouvoir les exécuter, un logiciel spécialisé se charge de transformer le code du programme en langage machine. On appelle cette étape la « compilation ». À chaque modification du code, il faut rappeler une étape de compilation.

Les avantages d'un langage interprété sont la simplicité (on ne passe pas par une étape de compilation avant d'exécuter son programme) et la portabilité (un langage tel que Python est censé fonctionner aussi bien sous Windows que sous Linux ou Mac OS, et on ne devrait avoir à effectuer aucun changement dans le code pour le passer d'un système à l'autre). Cela ne veut pas dire que les langages compilés ne sont pas portables, loin de là ! Mais on doit utiliser des compilateurs différents et, d'un système à l'autre, certaines instructions ne sont pas compatibles, voire se comportent différemment.

En contrepartie, un langage compilé se révélera bien plus rapide qu'un langage interprété (la traduction à la volée de votre programme ralentit l'exécution), bien que cette différence tende à se faire de moins en moins sentir au fil des améliorations. De plus, il faudra installer Python sur le système d'exploitation que vous utilisez pour que l'ordinateur puisse comprendre votre code.

#### 3.4 Environnement de développement intégré (IDE)

Logiciel permettant

- d'**écrire** des programmes;
- de les **exécuter**;
- de chercher l'origine d'un bogue (**débogueur**);

- de consulter la documentation du langage et des bibliothèques utilisées.

La plupart des IDE sont spécialisés à un ou à quelques langages, d'autres sont plus généralistes et ouverts sur grand nombre de langages.

Dans ce cours :



- langage : ;
- IDE : Pyzo ou IDLE (fourni avec Python).

### 3.4.1 Console interactive

On essaie successivement de taper :

```
2 + 3
a = 2
a
a=2
a      =    2
      a = 2
a + a
b + 1
b = 1
```

```
b + 1
b = b + 7
b = b + 5
b
2 ** 5
2 ** 1000
"hello"
"hello" + "world"
"hello" + " world"
```

### 3.4.2 Éditeur

Ouverture d'une nouvelle fenêtre d'édition : menu File/New File

On édite le texte suivant :

```
print("hello world!")
x = 42
```

Exécuter ce code : Run/Run Module (ou F5) Remarquez que python n'a pas écrit 42 :

Lorsqu'il s'exécute, un programme python n'affiche que ce qu'on lui demande explicitement d'afficher!

### 3.4.3 Débogueur

Dans l'éditeur on entre le programme suivant :

```
x = 231 / 77
z = 4
u = x**2 + 3 - z*3
t = z / u
print(t)
```

Exécution (F5 sous Idle ou Ctl+E sous Pyzo) : erreur!

Que se passe t-il?

On va regarder pas à pas en lançant le débogueur :

#### Pyzo

- Menu Shell/ Postmortem : debug from last traceback
- Le programme indique la ligne posant problème puis se positionne en la ligne en question pour tester les variables.

#### Idle

- Menu Debug/Debugger (de la fenêtre d'interaction) : ouvre une fenêtre nommée Debug Control.
- Cocher la case « Source ».
- dans l'éditeur, exécuter le programme (F5). La fenêtre Debug Control montre la ligne qui va être exécutée et la valeur des variables avant cette exécution et la ligne en cours d'exécution est mise en surbrillance dans la fenêtre d'édition.
- Cliquer sur Step pour exécuter cette étape.

### 3.4.4 Utilisation de python sur son téléphone

Différents sites proposent des IDE en ligne. Le site <http://repl.it> est plutôt performant. Il faut créer un compte (gratuit) et ensuite on dispose d'un interface assez ergonomique se rapprochant d'un IDE.

## 4 Architecture matérielle d'un ordinateur

### 4.1 Généralités sur le fonctionnement d'un ordinateur.

**Définition** C'est une machine :

- servant à traiter de l'information;
- programmable;
- universelle.

**Définition — Universelle.**

**Définition informelle** : « capable de calculer toute fonction exprimable par un algorithme ».

**Définition mathématique** : « capable de calculer toute fonction calculable par une machine de Turing ». Une machine de Turing est un dispositif abstrait, que nous ne détaillerons pas ici.

#### ■ Exemple — Exemples/conte-exemples. Exemple ou conte-exemple ?

1. Automobile
2. Thermostat d'ambiance (mécanique)
3. Thermostat d'ambiance (électronique)
4. Téléphone portable (smartphone)
5. PC de bureau
6. Lecteur MP3
7. Box de votre FAI
8. Métier Jacquard



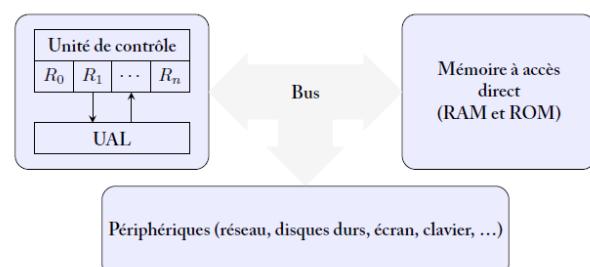
### 4.2 Modèle de Von Neumann

Les ordinateurs actuels sont tous bâtis autour du même modèle architectural théorique, l'architecture de Von Neumann, même si la mise en œuvre pratique est un peu plus complexe.

Le premier ordinateur électronique fut l'ENIAC (dont le but était de calculer des tables indiquant les paramètres de tir d'une batterie d'artillerie en fonction de la distance de l'objectif, du vent, etc.) dont la construction démarra en 1943. Son architecture a été décrite dans un rapport de John Von Neumann en 1945 et est depuis appelée « architecture de Von Neumann ». Depuis près de 70 ans, à quelques variations près, cette architecture sert de base à la plupart des systèmes à microprocesseur actuel.

Elle est composée des éléments suivants :

- d'une mémoire vive;
- d'un processeur qu'on peut conceptuellement décomposer en une unité de contrôle et une unité de calcul arithmétique et logique;
- de dispositifs périphériques, appelés simplement périphériques;
- d'un canal de communication entre la mémoire, le processeur et les périphériques, appelé le bus.

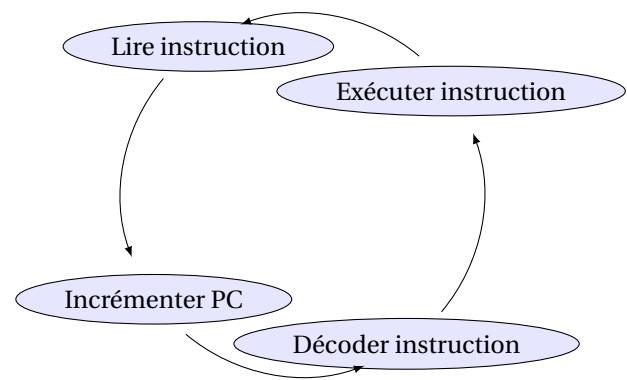


#### Avantages

- Simplicité conceptuelle.
- souplesse (peut manipuler tout ce qui est représentable numériquement, un programme est une donnée).

#### Inconvénients

- Exécution séquentielle.
- Goulet d'étranglement.
- Fragilité (faible robustesse aux erreurs).



Processeur :

- mémoire réduite à quelques mots appelés registres;
- calculs sur les registres par l'Unité Arithmétique et Logique;
- accès à la mémoire vive;
- unité de contrôle.

FIGURE 2 – Déroulement schématique des actions effectuées par l'unité de contrôle

### 4.3 La carte mère et ses composants

#### 4.3.1 La carte mère

La carte mère est un des éléments central de l'ordinateur. Elle assure la liaison avec le boîtier, mais surtout, elle comporte :

- l'interface avec le processeur *via* le socket (le socket de la carte mère est à choisir en fonction du processeur souhaité);
- les connecteurs permettant de brancher la mémoire RAM;
- le BIOS;
- les connecteurs SATA permettant par exemple de relier le disque dur;
- la carte son et souvent la carte graphique;
- les connecteurs USB, audio, video (VGA ou HDMI), RJ45... permettant la connexion des différents périphériques;
- des ports PCI (PCI-Express) permettant de connecter des cartes d'interfaces supplémentaires *etc.*

#### 4.3.2 Le processeur

Le processeur (ou unité centrale de traitement) est le composant exécutant toutes les opérations de base. La fréquence d'un processeur, s'exprime en Ghz. Il s'agit du nombre d'opérations qu'il peut faire par seconde.

En effet, un processeur peut-être multi-coeur, c'est-à-dire qu'il possède plusieurs coeurs physiques fonctionnant de façon simultanée. Un processeur multi-coeur permet de réaliser des opérations simultanées (à condition que le système d'exploitation puisse le gérer). De plus, il existe des processeurs multi-thread permettant d'exécuter plusieurs tâches en parallèles (sur le même processeur).

En conséquence, pour comparer la rapidité de deux processeurs, la fréquence ne suffit pas, le nombre de coeurs et le nombre de threads sont à prendre en compte.



#### 4.3.3 Notion de bus

En informatique, le terme bus désigne un ensemble de fils support de l'information et organe de communication entre différents composants. Il existe deux grands types de bus :

- le bus série : il comporte plusieurs fils, dont la masse (référence de potentiel), le fil de données, le fil d'horloge. Les données sont transmises en série;
- le bus parallèle : il comporte un fil de masse, un fil d'horloge, et  $n$  fils de données pour un bus  $n$  bits; les données sont transmises en parallèle.

Par ailleurs, des bus sont intégrés dans des puces. Par exemple, dans les microprocesseurs, les différents constituants communiquent entre eux par des bus parallèles. Ainsi, sur la carte mère, la communication est assurée par :

- le bus système qui relie le microprocesseur au chipset;
- le bus mémoire qui relie le chipset à la mémoire vive;
- le bus d'extension qui relie le chipset aux connecteurs d'entrée – sortie.

Sur la carte mère sont intégrés différents bus parallèles que l'on peut même parfois distinguer. Ces bus permettent de relier les différents constituants entre eux afin qu'ils communiquent. On retrouve des bus d'adresses, de données et de contrôle.

Entre la carte mère et les différents périphériques (disque dur par exemple), on utilise de plus en plus des bus séries. Ainsi, après le tout IDE, bus parallèle qui a été massivement utilisé dans les ordinateurs grand public, le bus SATA est de plus en plus utilisé.

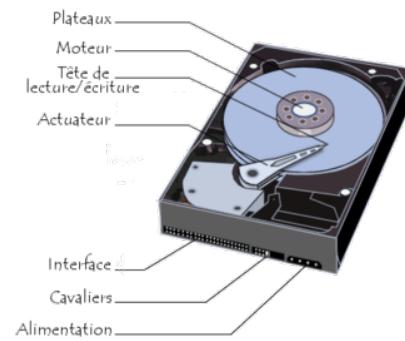
Plusieurs normes SATA ont été développées. Cette technologie présente des débits bien plus importants que les disques durs mécaniques (60 Mo/s au mieux) mais s'adapte bien aux débits des SSD qui avoisinent les 500 Mo/s.

Norme	Débit (Mo/s)
SATA I	187,5
SATA II	375
SATA III	750

## 4.4 Les mémoires

### 4.4.1 Les mémoires de masse – Disque dur

Un disque dur (non SSD) est composé de disques rigides et paramagnétiques (en aluminium, en verre ou en céramique) recouverts d'une couche magnétique et d'un film protecteur. Ces plateaux sont entraînés par un moteur électrique dont la fréquence de rotation est fixe. La fréquence de rotation peut atteindre les  $15\,000 \text{ tr min}^{-1}$ . Un bras, sur lequel sont fixées les têtes de lecture – écriture, est aussi motorisé pour atteindre les différentes parties du disque. La distance entre les têtes et les plateaux est de l'ordre de  $10 \text{ nm}$ . Elle est maintenue par un coussin d'air formé par la rotation des disques.



Les têtes de lecture – écriture sont inductives et permettent de générer un champ magnétique dans deux sens opposés permettant de polariser la couche magnétique du disque. Ainsi, en mode lecture les changements de polarité sont transcrits par un convertisseur analogique numérique (CAN) en bits (0 ou 1).

Les données sont organisées en pistes concentriques, les données les plus rapidement accessibles étant celles situées sur le disque extérieur. Chaque piste est divisée en secteurs (d'un minimum de 512 octets). Enfin, on appelle cylindre l'ensemble de données situé sur une même piste mais sur un plateau différent.

La capacité des disques durs atteint les 4 To (téraoctet). Lorsqu'ils sont connectés en SATA, le débit oscille entre 30 et 60 Mo/s et peut atteindre les 600 Mo/s. Enfin, pour améliorer les performances d'accès aux données, ils possèdent une mémoire cache pouvant aller jusqu'à 128 Mo sur laquelle sont stockées les informations fréquemment utilisées.

Les performances des disques SSD (*Solid State Drive*) sont supérieures à celles des disques durs classiques. Deux critères penchent néanmoins en sa défaveur. D'une part le prix du Go est presque 10 fois plus élevé que pour un disque dur mécanique. D'autre part, certaines technologies de SSD présentent l'inconvénient d'être limité en nombre de cycles d'écriture – lecture.

Caractéristiques	Disque dur mécanique	SSD
Temps d'accès	En moyenne 12ms	0.1ms
Poids	De 400g à 700g	Quelques dizaines de grammes
Consommation en veille	De l'ordre du Watt	100 mW
Consommation en activité	4W environ	900 mW
Bruit	En moyenne 40 dB	0 dB

### 4.4.2 La mémoire cache et mémoire vive (RAM)

La mémoire cache est une mémoire très rapide que l'on trouve par exemple sur les microprocesseurs ou sur les disques durs. Dans le cas des microprocesseurs, elle est située au plus proche de ceux-ci et permet d'accélérer l'exécution des programmes.

La mémoire vive (RAM) est plus rapide d'accès qu'un disque dur (de 1000 à 10 000 fois plus rapide), mais plus lente qu'une mémoire cache. Elle permet de stocker (à court terme) des données. C'est elle qui permet d'utiliser plusieurs applications en même temps et de passer rapidement de l'une à l'autre.

## 4.5 Communication avec les périphériques

### 4.5.1 Le port USB (usb)

USB (*Universal Serial Bus* – bus série universel) est un protocole de communication apparu en 1996 (USB 1.0). Ce protocole série a révolutionné les connexions entre PC et périphériques en instaurant un environnement « tout USB », uniformisant les modes de communication avec l'ordinateur... C'est sans doute le bus de communication le plus utilisé. Ses principaux concurrents sont désormais les protocoles sans fil Bluetooth et WiFi.



Version	USB 1.0	USB 1.1	USB 2.0	USB 3.0
Débit	$0,19 \text{ Mo s}^{-1}$	$1,5 \text{ Mo s}^{-1}$	$60 \text{ Mo s}^{-1}$	$600 \text{ Mo s}^{-1}$

Un connecteur USB est composé de 4 fils :

- un fil d'alimentation (5V – VBUS);
- un fil de masse (GND);
- deux fils de données (D+ et D- torsadés).

Afin de faire transiter l'information, l'USB utilise des paquets. Ainsi une transaction USB est composée de 3 paquets :

- le paquet *token* contient des informations sur la nature de la communication (est-ce l'hôte ou le périphérique qui envoie de l'information?);
- le paquet de données;
- le paquet *handshake* qui contient des informations sur le déroulement de la transaction (le paquet a été reçu correctement, interruptions lors de la transaction...).

### 4.5.2 Le port série

Pour les communications séries, les connecteurs sont de type D-SUB (DB9 par exemple).

Historiquement, le port série est le premier port de communication utilisant une transmission série (norme RS232). Son débit est au maximum de 19200 bps (bits par seconde). Il a été très longtemps utilisé pour sa simplicité de configuration et de pilotage. C'est d'ailleurs le protocole de communication mis en œuvre pour un grand nombre de système du laboratoire : MAXPID, cordeuse, DAE, etc. A noter cependant que si certains PC ont encore un port COM1 RS232, la plupart utilise un adaptateur RS232-USB pour communiquer avec le PC.

### 4.5.3 Le port parallèle

Pour les communications séries, les connecteurs sont de type D-SUB (DB25 par exemple).

Le port parallèle est un port reposant sur un protocole unidirectionnel de communication parallèle. Il est en voie d'extinction, comme le port série RS232. Il était le plus souvent utilisé pour la communication PC-imprimante. Le débit maximum est de 16Mbps.

### 4.5.4 Port PS/2 – Personal System/2

Le port PS/2 utilise un connecteur mini-din. Ce port de communication de taille réduite a succédé au PS/1 (Din, le même en plus encombrant) permettant la connexion du clavier et/ou de la souris créé par IBM (1987) mais démocratisé en 1995 suite à son intégration sur les cartes mères type ATX.

Il est également supplanté par le standard USB depuis quelques années mais également de plus en plus par le bluetooth.

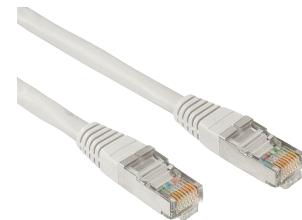


## 4.6 Communication réseau

### 4.6.1 Port RJ45 – Registered Jack

Ce port permet la connexion filaire réseau ethernet (LAN : Local Area Network) par câble. Différents protocoles existent, spécifiant différents débits.

Protocole	Gigabits	10GBase-X	100GBase-X
Débit	128 Mo/s	1280 Mo/s	12 800 Mo/s



### 4.6.2 Wi-Fi

Le Wi-Fi (*Wireless Fidelity* – Norme IEEE 802.11) est un standard définissant les caractéristiques d'un réseau local sans fil. La liaison est assurée par l'intermédiaire d'ondes électromagnétiques.

Différentes normes physiques définissent le débit de la transmission en fonction de la portée.

Standard	Bandes de fréquence	Débit	Portée
WiFi a (802.11a)	5 GHz	54 Mbit/s	10 m
WiFi B (802.11b)	2,4 GHz – 2,5 GHz	11 Mbit/s	100 m
WiFi G (802.11g)	2,4 GHz – 2,5 GHz	54 Mbit/s	75 m
WiFi N (802.11n)	2,4 GHz ou 5	450 Mbit/s	125 m

#### 4.6.3 3G – 4G

La 3G et la 4G sont les troisième et les quatrième générations de transmissions de données pour la téléphonie mobile. La 4G permet d'avoir accès à des débits théoriques très supérieurs à la 3G et possède un cœur de réseau basé sur IP.

Tableau récapitulatif <http://www.commentcamarche.net/>

Standard	Génération	Bandes de fréquence	Débit	
GSM	2G	Permet le transfert de voix ou de données numériques de faible volume.	9,6 kbps	9,6 kbps
GPRS	2.5G	Permet le transfert de voix ou de données numériques de volume modéré.	21,4-171,2 kbps	48 kbps
EDGE	2.75G	Permet le transfert simultanés de voix et de données numériques.	43,2-345,6 kbps	171 kbps
UMTS	3G	Permet le transfert simultanés de voix et de données numériques à haut débit.	0.144-2 Mbps	384 Kbps
LTE	4G	Permet le transfert simultanés de voix et de données numériques à haut débit.	10-300 Mbps	5-75 Mbps

#### 4.7 Son et vidéo

Le port VGA – *Video Graphics Array*

Le connecteur est de type D-SUB, ici DE-15. Ce port est de type analogique.



Le port DVI – *Digital Visual Interface*

Ce port est de type numérique non HD. Il apporte une amélioration en terme de réduction du bruit par rapport au connecteur VGA analogique.



Le port HDMI – *High Definition Multimedia Interface*

Il s'agit d'une interface audio vidéo totalement numérique permettant de raccorder un lecteur de disque, une console, un décodeur etc. à un écran de télévision haute définition.



### Références

- [1] Wack, Conchon, Courant, deFalco, Dowek, Filliatre, Gonnord, Informatique pour tous en classes préparatoires aux grandes écoles, éditions Eyrolles.
- [2] Apprenez à programmer en Python <http://www.siteduzero.com/>.
- [3] François Médevielle, Architecture des ordinateurs – UPSTI.
- [4] Patrick Beynet, ISN – Architectures matérielles – Architecture des ordinateurs. Lycée Rouvière.
- [5] Ressources des enseignants du lycée La Martinière Monplaisir.

## TD

Sources :

**Savoirs et compétences :**

- AA.C01 : Manipuler un OS ou un IDE
- AA.S01 : Se familiariser aux principaux composants d'une machine numérique
- AA.S2 : Se familiariser à la manipulation d'un OS
- AA.S3 : Se familiariser à la manipulation d'un IDE