



# Information, Calcul et Communication

Faculté Informatique et Communications

**Dr Ronan Boulic**

## Les deux composantes du cours : Théorique et Pratique

### Théorie

Cours classique **Mardi 8-10h**

**Moodle: 14002**

Exercices théoriques sur papier

**Vendredi 14-15h**

Salles:

**INM 200** : section **EL**

**CG C3 30** : section **MT** de A. à E.

**CE1**: section **MT** de F. à Z.

10 assistant(e)s

### Pratique (Programmation C++)

Classe inversée **vendredi 8-9h**

avec MOOC dès la semaine 2

**Moodle: 6831**

Travaux Pratiques sur ordinateur

ou laptop personnel

**Vendredi 15-17h**

Salles:

Avec son laptop: **INM 200, CE4, CE6**

Sur ordi EPFL: **CO 020-21-23, CO5**

15 assistant(e)s

## Communication

- Pendant les séances d'exercice/TP : ***interagir avec les assistants***
- A tout moment: vous pouvez ***utiliser les deux forums moodle*** (théorie / pratique) pour poser vos questions relatives au cours (contenu, organisation).
  - Vous pouvez aussi ***contribuer en répondant aux questions***
- Nous utilisons les forums ***pour communiquer avec vous*** pendant tout le semestre (ex: salles des tests, notes, etc...).
  - Il faut donc rester inscrit à ces 2 forums.
- **Le bon usage des forums / comment éviter de créer du spam :**
  - **UTILISER L'OUTIL DE RECHERCHE DU FORUM** pour vérifier si votre question a déjà été posée. Ainsi vous ne perdrez pas de temps à attendre une réponse
  - **NE PAS REMERCIER EN FAISANT REPLY** car tout le monde reçoit ce spam. Vous pouvez remercier en envoyant un email individuel à la place.

## Examens et rendu pratique : pendant le semestre

### Théorie

Examen écrit (module 1 théorie)

**Vendredi 2 Novembre 14-16h**

Aucun document autorisé

### Pratique (Programmation C++)

Examen écrit (Programmation C++)

**Vendredi 23 Novembre 15-17h**

Tout document papier autorisé

Rendu du mini-projet de Programmation

**Dimanche 9 décembre minuit**

Examen écrit (Théorie et Pratique/C++)

**Vendredi 21 Décembre 14-17h**

Tout document papier autorisé

## Pondération et calcul de la moyenne finale

Théorie\_1: 25%

Pratique\_1: 19%

Mini-Projet: 12%

Theorie\_2 : 25%

et

Pratique\_2 : 19%

- 1)  $V$  = moyenne pondérée des notes avec normalisation à 5
- 2)  $V$  est arrondie au quart de point le plus proche
- 3) Votre note académique transmise au SAC est  $(1 + V)$

Remarque: l'absence à un examen est sanctionnée par une note de 0 pour cet examen

# Introduction générale ICC

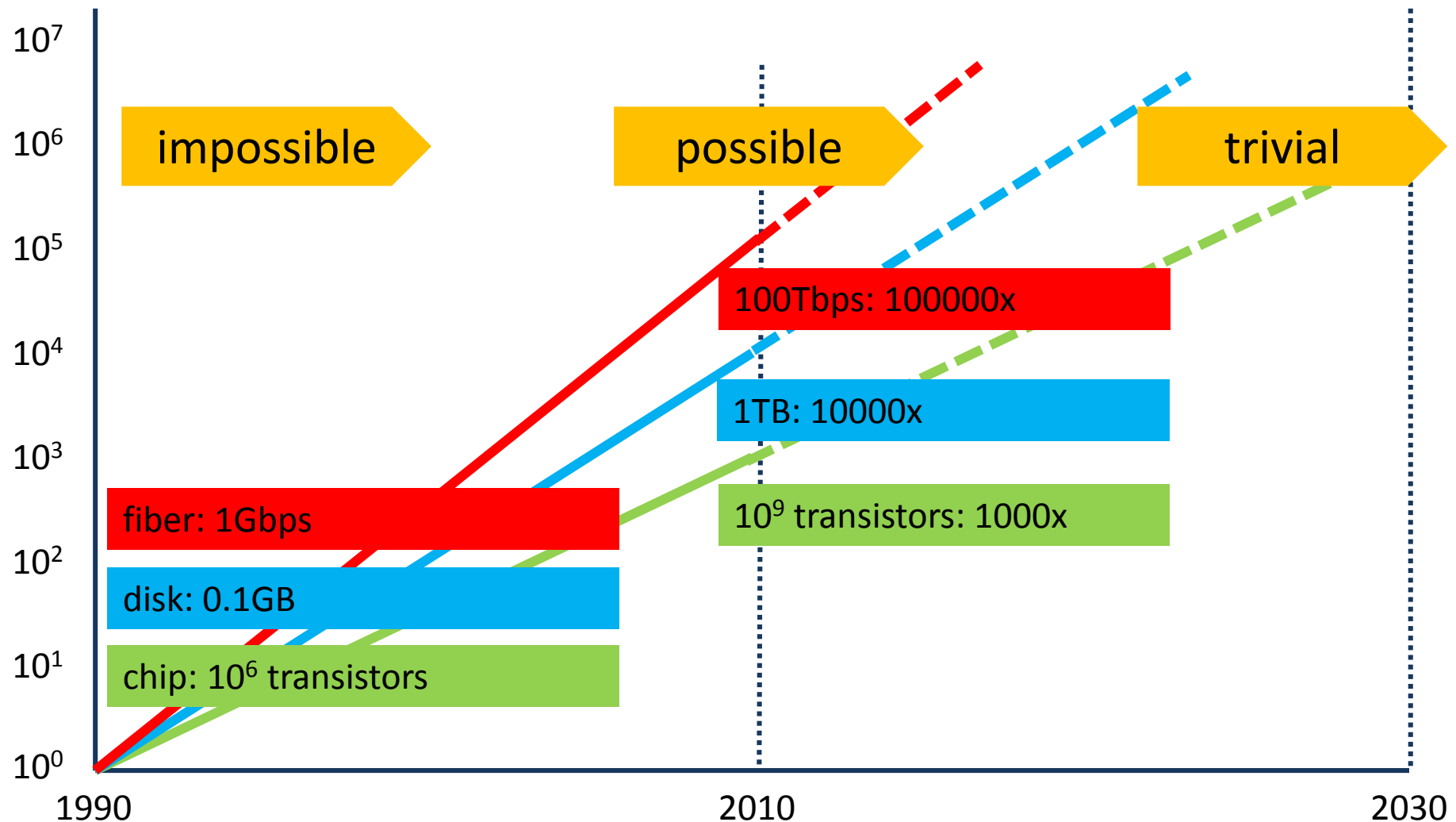
- Une accélération vertigineuse de l'humanité, la technologie, l'informatique
- Tous les secteurs d'activité sont affectés
- L'informatique comme 4ème pilier de la culture
- Plan de la composantes théorique du cours
- Lien avec la composante pratique de programmation
  - qu'est-ce qu'un ordinateur ?

# Les racines de l'informatique

- Préhistoire – le langage et le comptage
- Histoire – écriture, calcul, géométrie, astronomie, – abaquas
- Renaissance – règle à calcul
- 19e siècle – la machine de Babbage, les logiciels de Ada Lovelace
- 1940 – ordinateurs
- 1960 – minis – réseaux
- 1980 – micros – PCs
- 1990 – la toile
- 2000
  - System-On-Chip, smartphones
  - Multimédia
  - Cloud computing, réseaux sociaux
  - Google, Youtube, Skype, Facebook, Twitter, etc.



# Les accélérations spécifiques de l'Informatique





## Les accélérations spécifiques de l'Informatique



Loi de Moore [1965 -2015] ?

<http://spectrum.ieee.org/semiconductors/processors/the-multiple-lives-of-moores-law>

## Tous les secteurs d'activité sont affectés

Biens	←	...	...	→	Services
Agriculture	Industrie	Commerce	Services «publics»	Services professionnels	Admin. Publique
Culture Chasse Pêche	Extraction Traitement Manufact.	Packaging Distribution Transport Voyage	Eau/gaz/élec. Télécom. Médias/showbiz Monde artistique	Conseil Finance Assurance	Médical Education Gouvern. (ONG incl.)
Météo Captage	Prospection	Marketing Logistique	Multimedia	Optimisation Syst. Experts	Multimedia Syst. experts
	CADCAM	Télécom.	Mesure/contrôle Télécom.	Télécom.	Télécom.

Calcul scientifique:  
modélisation, simulation, optimisation

Gestion de données

Gestion de processus

# Calcul scientifique



C'est l'application historique, héritée de la génération des calculateurs (« computer »)

**Utilisation** : simulation de systèmes complexes (**compréhension de fonctionnement, test d'hypothèses, prédiction**)

**Exigences** : Grande puissance de calcul

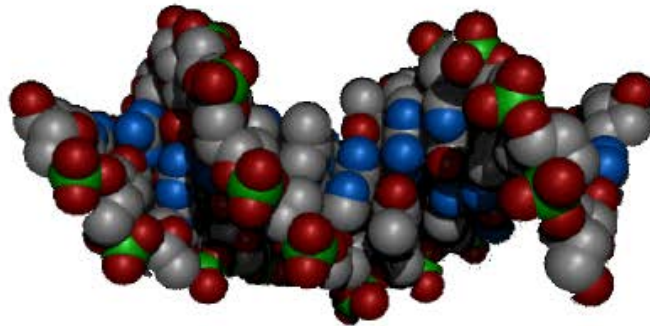
**Exemples** : super calculateurs, massivement parallèles  
(**Cray-1, Cray T3D, SV1, IBM Blue Gene, ...**)  
Bibliothèques de programmes mathématiques

**Nouvelles tendances** : « grappes » d'ordinateurs, network computing

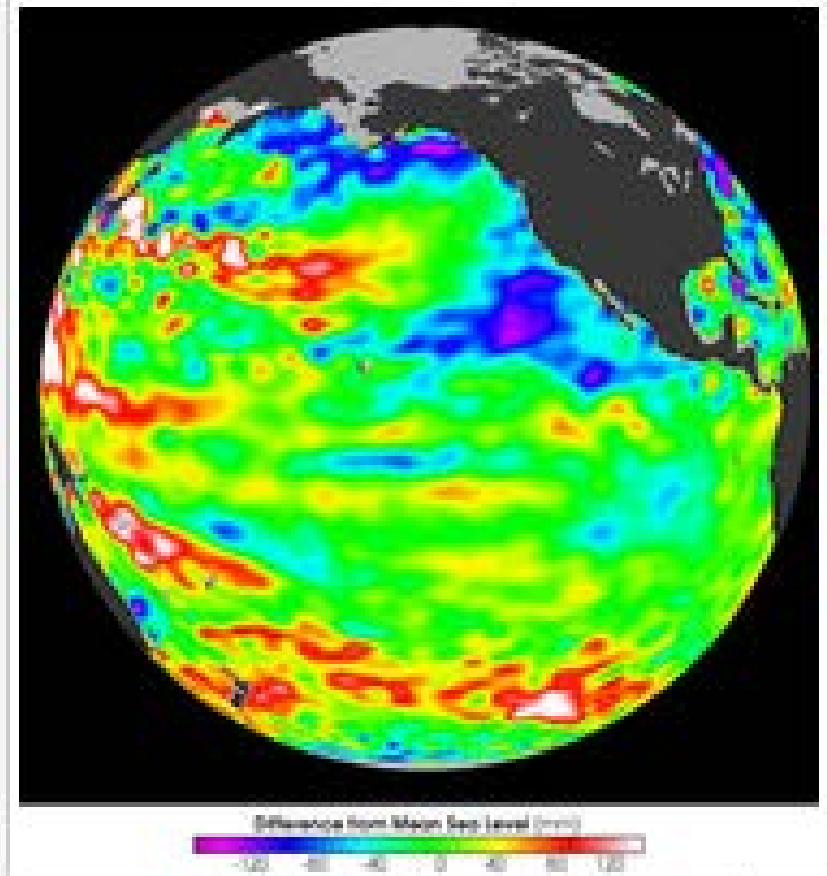


# Calcul scientifique

Applications multiples :



analyser une séquence ADN



En 2003, mise en évidence du "Niño"  
par simulation numérique de la  
circulation océanique - © INRIA / Projet  
IDOPT

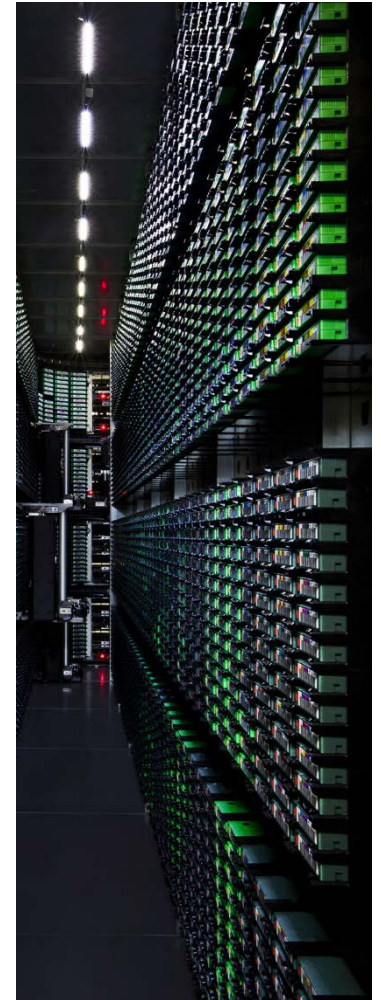
# La gestion d'information

Gestion et traitement des données.

**Utilisation** : gestion de systèmes bancaires ou boursiers, commerce électronique (vente et réservation en ligne), comptabilité d'entreprise, etc..

**Exigences** : importantes capacités de stockage, traitement efficace (rapide, fiable et sécurisé) de gros flux d'information

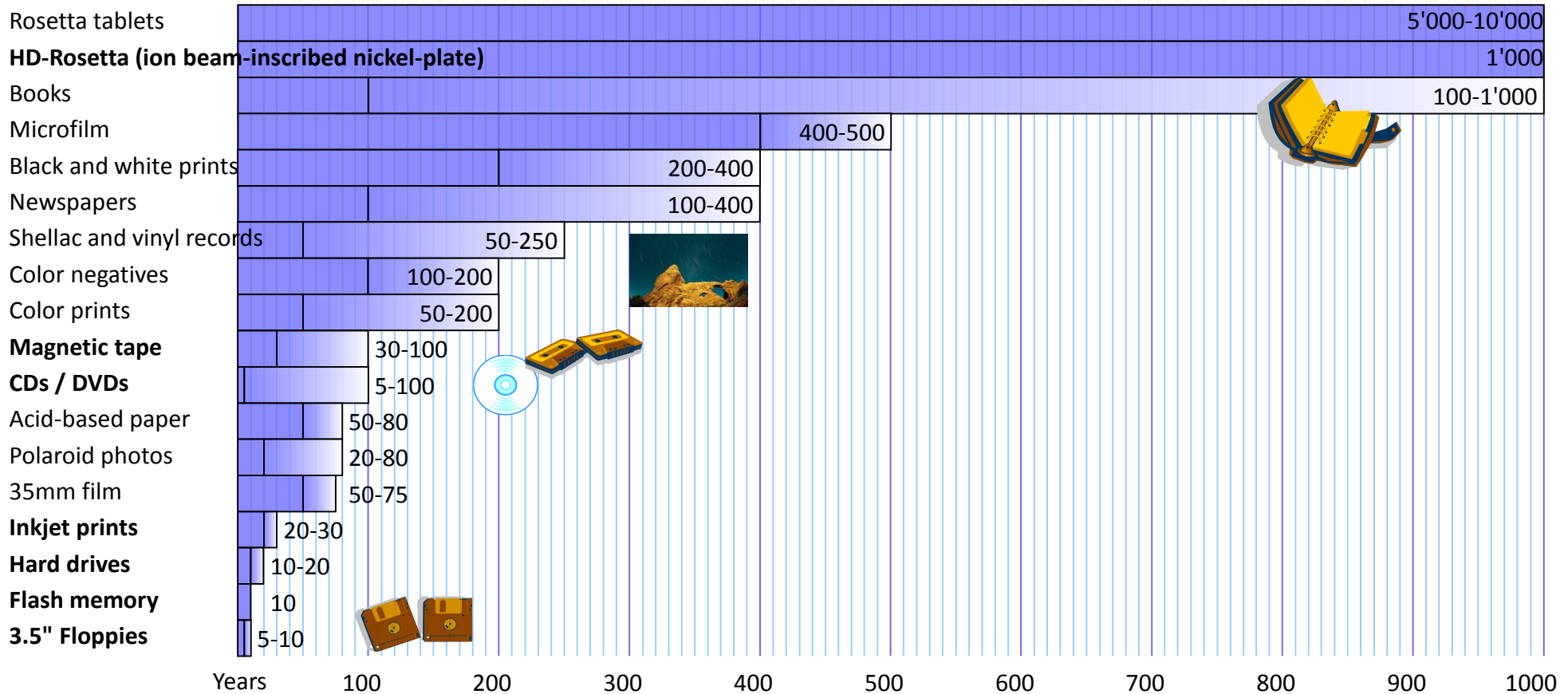
**Exemples** : Ordinateurs avec mémoire de masse importante, et fortes capacités en matière de communications (entrées/sorties)



data center

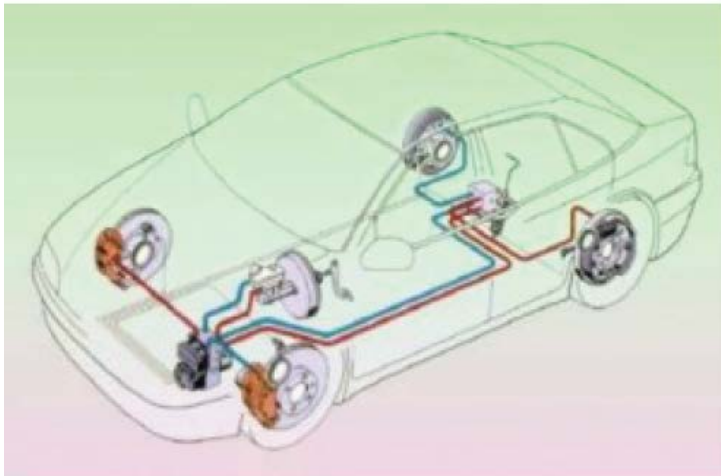
# La gestion d'information doit aussi faire face au problème de la faible longévité des supports-mémoires

Les supports d'information ne sont plus ce qu'ils étaient



Source: Wired June 2002, p 062

# Conduite de processus : systèmes embarqués



Système embarqué



Robot de ménage



ASIMO



# La conduite de processus

Ordinateur = automate de commande

**Utilisation** : très nombreuses applications : pilotage/surveillance de processus industriels (chaînes de fabrication, de montage, réseaux de distribution d'énergie, centrale atomique), fonctionnalités d'objets courants (four micro-ondes, téléphones cellulaires, machines à laver, chronométrage, carburateur de voiture, système de freinage ABS), avionique, robotique, ...

**Exigences** : faible encombrement, consommation réduite, coût minimum, grande fiabilité

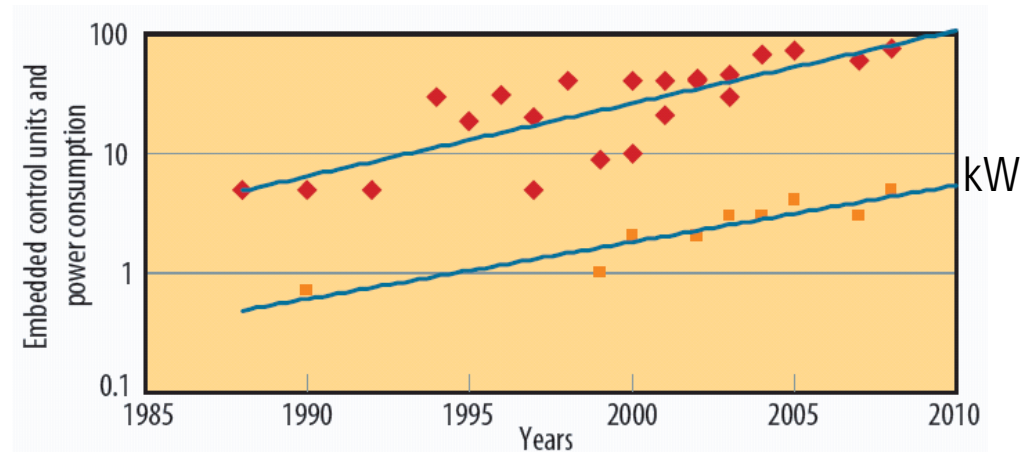
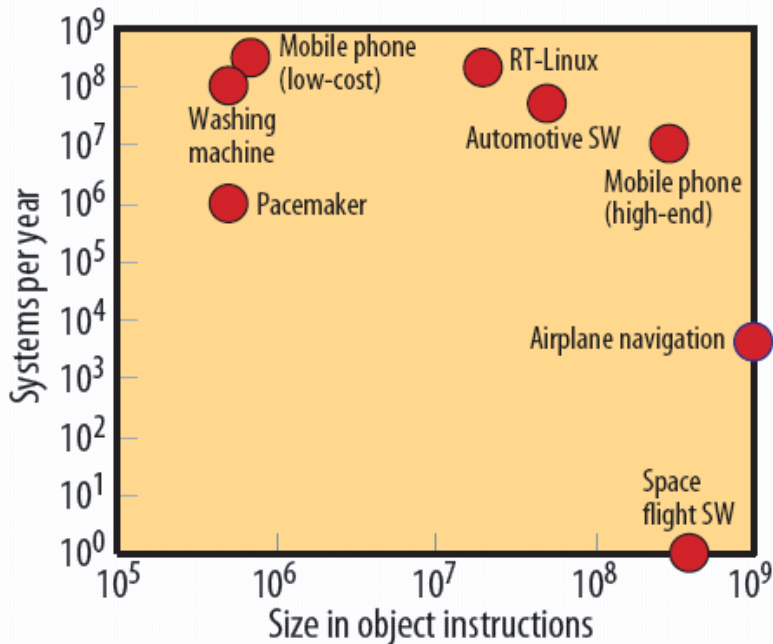
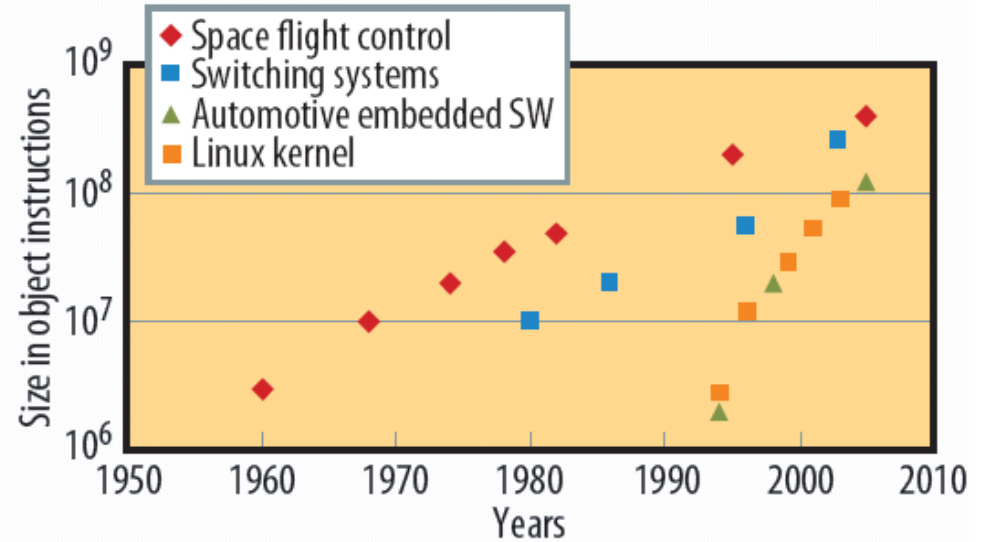
**Exemples** : Initialement l'ensemble des micro-contrôleurs, de plus en plus souvent des processeurs, voire des ordinateurs complets





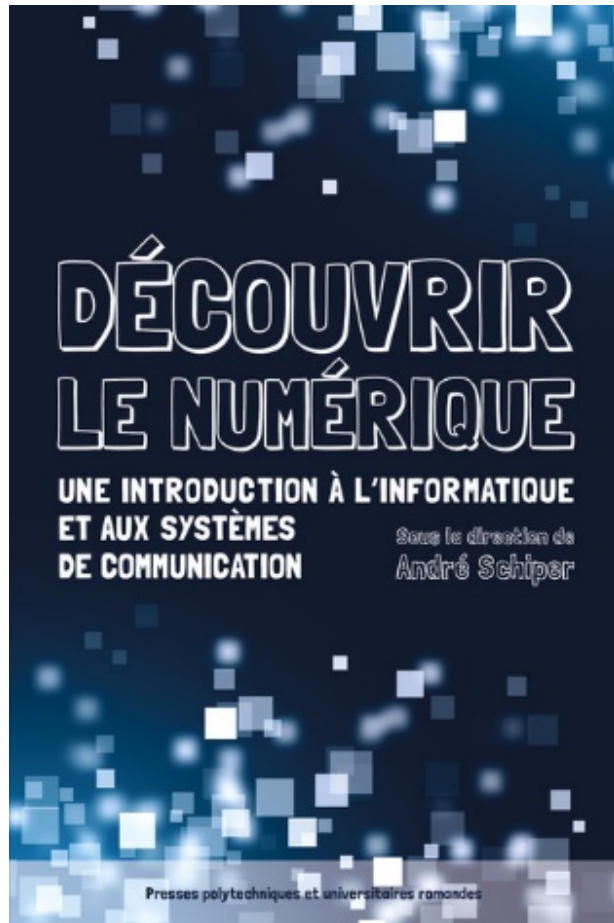
# Exemples de taille de systèmes embarqués en nb d'instructions (code exécutable)

Quelques chiffres  
[IEEE Computer Avril 2009]



Ligne du haut: Nb de systèmes embarqués dans une voiture haut de gamme

# Plan du cours : partie théorique



## Semaine

### 1. Introduction

### • **Fondements du calcul**

2. Représentation de l'information
3. Qu'est-ce qu'un algorithme ?
4. Calcul et algorithmes
5. Stratégies de calcul
6. Théorie du calcul (Alan Turing)

### •

### ▪ **Fondements des communications**

7. Echantillonnage
8. Reconstruction
9. Entropie (Claude Shannon)
10. Compression

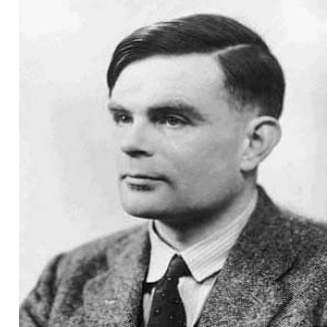
### •

### ▪ **Fondements des systèmes**

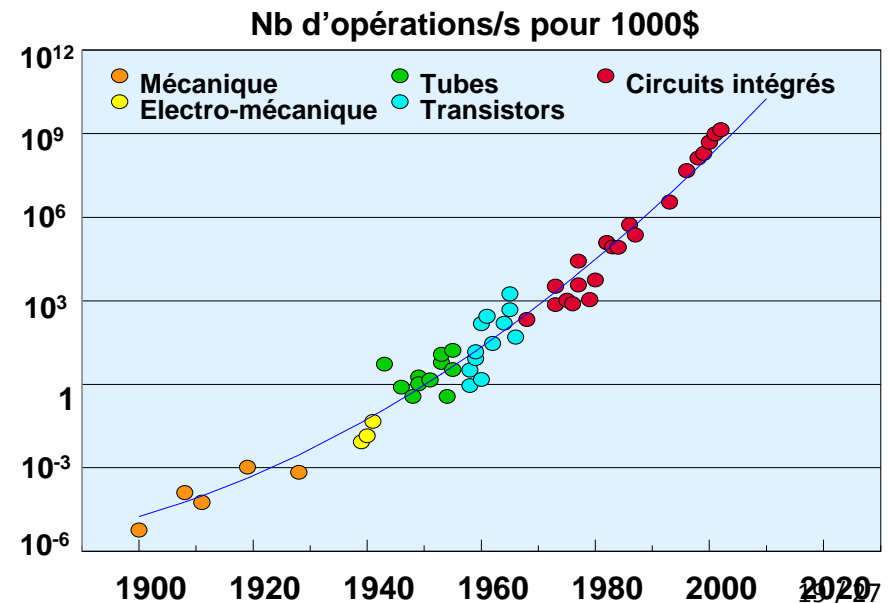
11. Architecture (John Von Neumann)
12. Mémoires hiérarchiques
13. Stockage et Réseaux
14. Sécurité

•

# Fondements du calcul



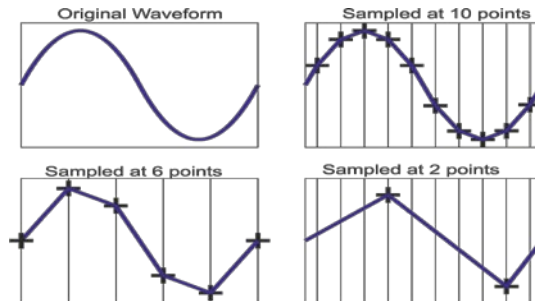
- **Représentation de l'information**
  - Nombres, lettres, images, son, ...
  
- **Qu'est-ce qu'un algorithme ?**
  - Données, traitements, contrôle
  
- **Calcul et algorithmes (Alan Turing)**
  - Recherche, plus court chemin, tri
  
- **Stratégies de calcul**
  - Top-down / bottom-up, «divide & conquer», etc.
  
- **Théorie du calcul**
  - Le possible et l'impossible, le fini et l'infini,



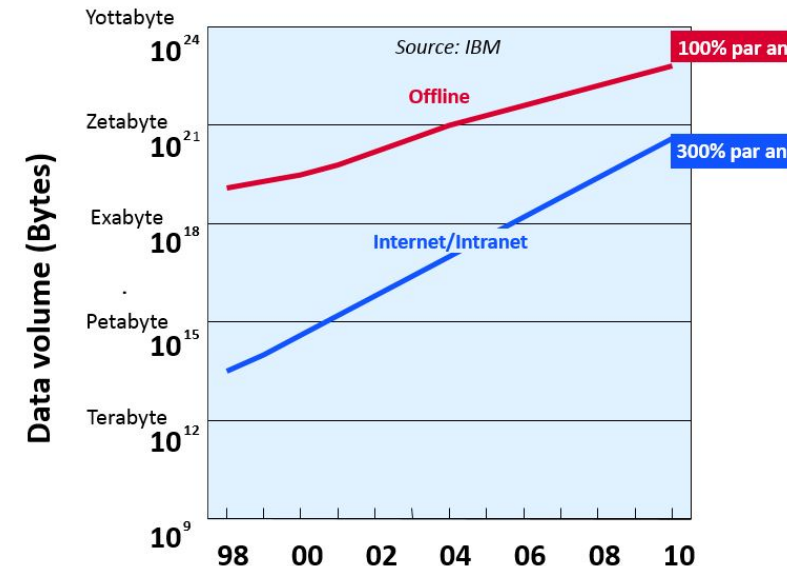
Source: d'après Kurtzweil 1999 et Moravec 1998

# Fondements des communications dans l'espace (stockage) et dans le temps (transmission)

- **Echantillonnage**
  - Conversion A/D
- **Reconstruction**
  - Conversion D/A



- **Entropie (Shannon)**
  - L'entropie comme mesure de complexité de l'information
- **Compression**
  - Économie de bits avec ou sans perte d'information pour économiser temps de transmission ou espace de stockage



# Fondements des systèmes

## ▪ Architecture (John Von Neumann)

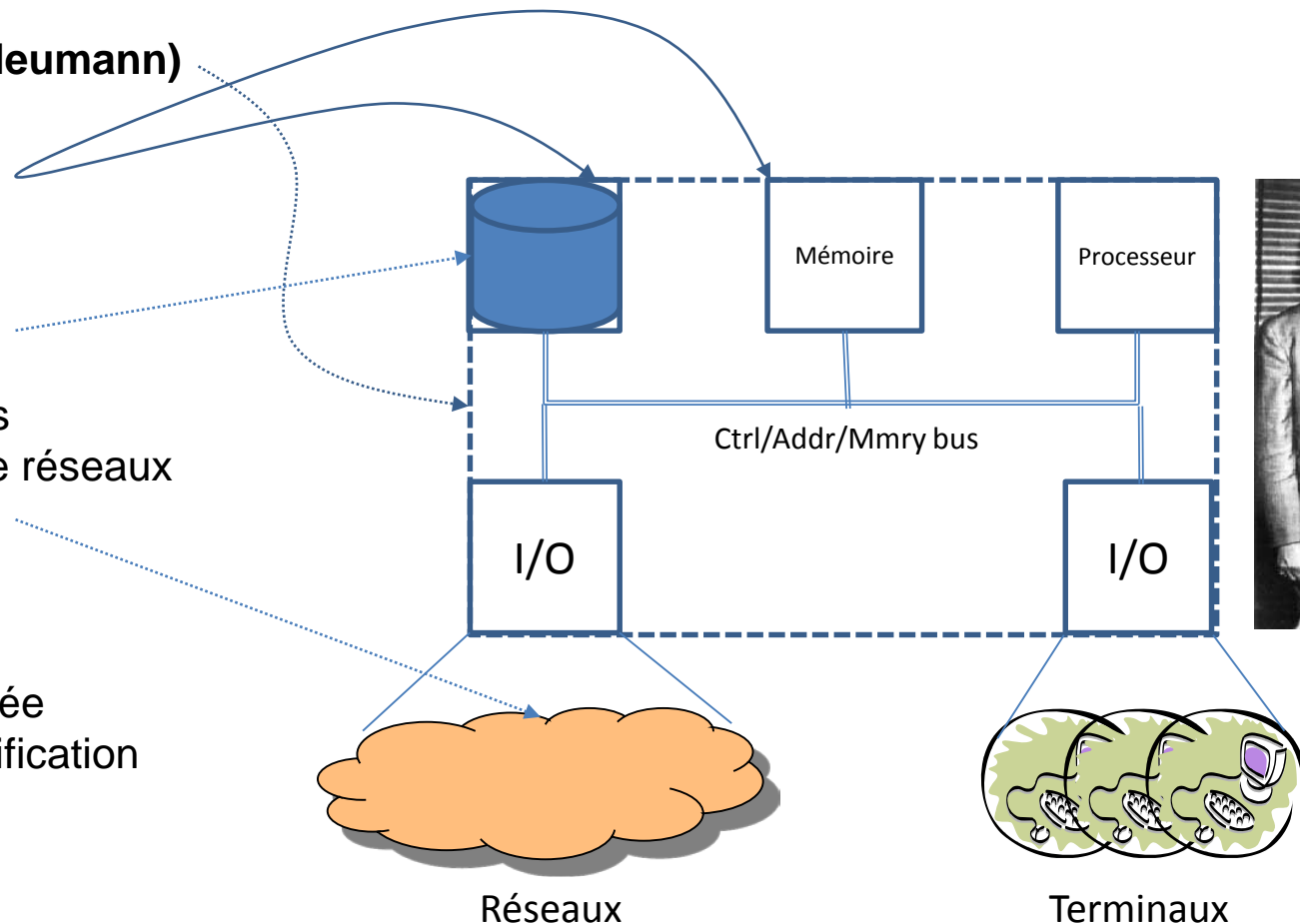
## ▪ Mémoires hiérarchiques

## ▪ Stockage et réseaux

- Systèmes de fichiers et bases de données
- Types et architecture de réseaux (Internet)

## ▪ Sécurité

- Cryptage et sphère privée
- Identification et authentification
- Contrôles d'accès



# Lien avec la composante pratique : Programmation en C++

## Qu'est-ce qu'un ordinateur ?

c'est une machine au coeur de laquelle le **processeur** ou **CPU** peut effectuer une **grande variété de traitements** sous forme de *programmes*.

## L'exemple de l'ordinateur personnel (PC):

Applications bureautiques (calcul, texte, présentation, projet), internet, manipulation d'images, video, son(g)s, jeux, interaction avec une large gamme de périphériques

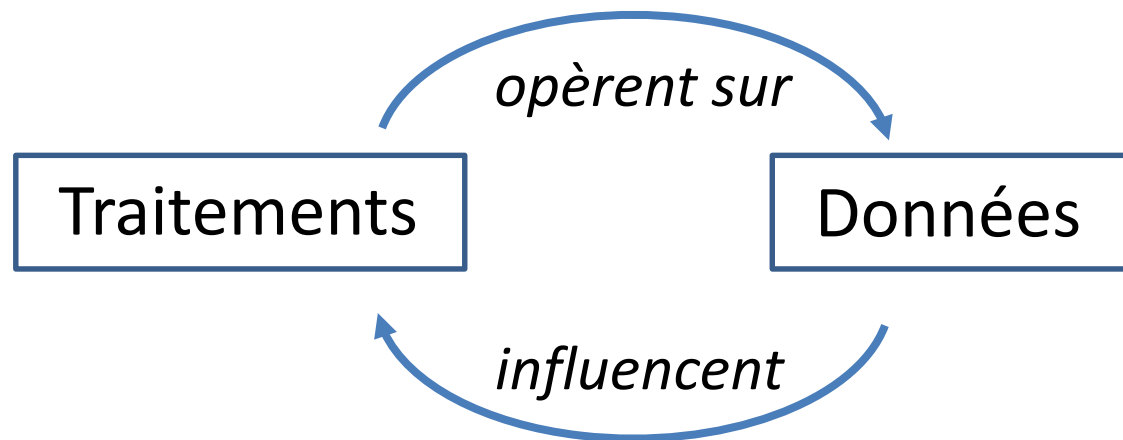


# L'ordinateur est une machine séquentielle

L'ordinateur est une machine de *traitement numérique des données à l'aide d'une séquence d'instructions appelée programme.*

## Cadre du cours:

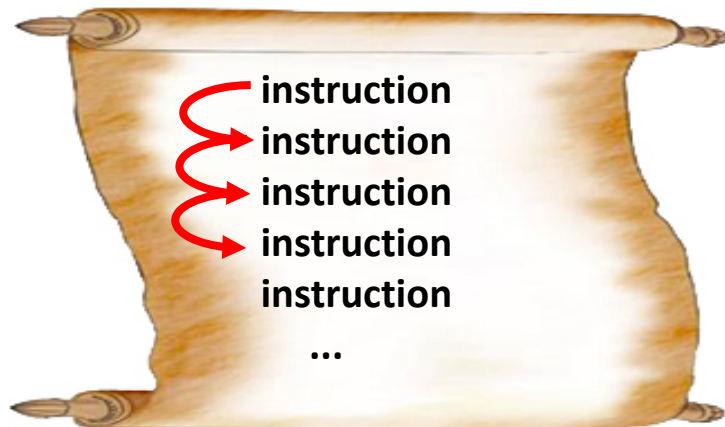
une seule instruction est exécutée à la fois par un seul processeur



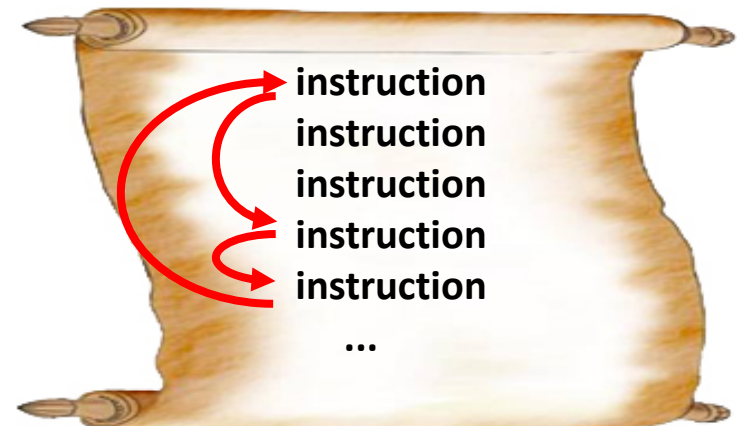


## Deux scénarios d'exécutions possibles selon le **type** d'instruction exécutée:

a) **Traiter une donnée**, ex: effectuer une opération arithmétique. Dans ce cas le programme *passé automatiquement à l'instruction qui suit*.



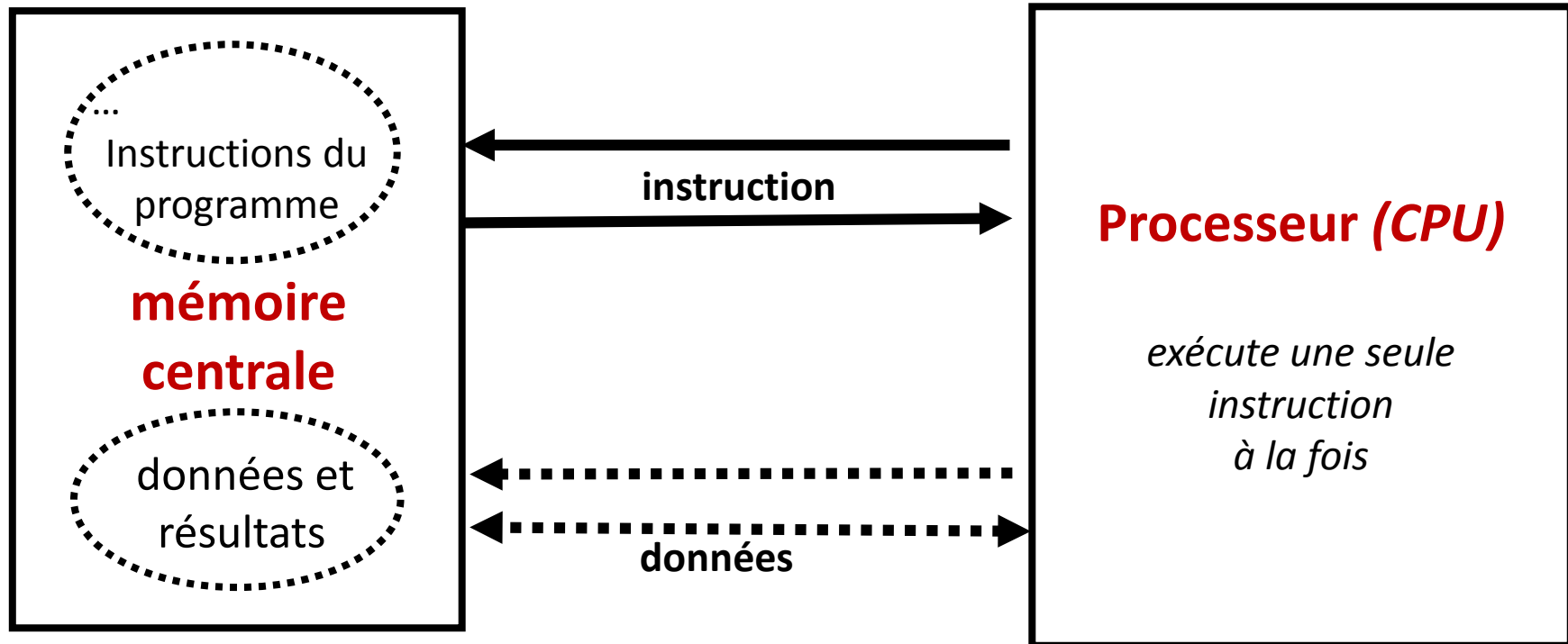
b) **Effectuer un Branchement**, en general lié à une condition sur les données (on parle d'instructions de contrôle): il y a rupture de *l'exécution séquentielle*.





# Le rôle de la mémoire centrale

Les instructions ET les données à traiter ET les résultats du traitement peuvent être stockés de la même manière dans un même espace: la *mémoire centrale*.



## Lien avec la composante pratique : Programmation en C++ (2)

Le Module1 de la partie théorique met l'accent sur la résolution générale de problèmes à l'aide *d'algorithmes* écrit en *pseudocode*. Cette approche permet de raisonner sur la validité et l'efficacité d'une solution, indépendamment de tout langage de programmation.

Pour passer du *pseudocode* au *code* du programme, il faut en plus maîtriser un langage particulier, dans notre cas le **C++**.

Nous recommandons également de faire une ébauche en pseudocode pour les exercices “avancés” proposés en Programmation

## Conclusion

Les 3 modules de la composante théorique veulent offrir des bases à la fois d'un point de vue théorique et "système".

Ces bases sont non seulement utiles pour mieux maîtriser la composante pratique de programmation mais aussi pour mieux appréhender certains grands enjeux de notre société numérique.

Dans la seconde heure, nous allons illustrer l'importance de l'informatique dans vos domaines respectifs de la microtechnique, avec le Prof. Francesco Mondada, et du génie électrique, avec le Prof. Jean-Philippe Thiran