

*Chapitre 1*

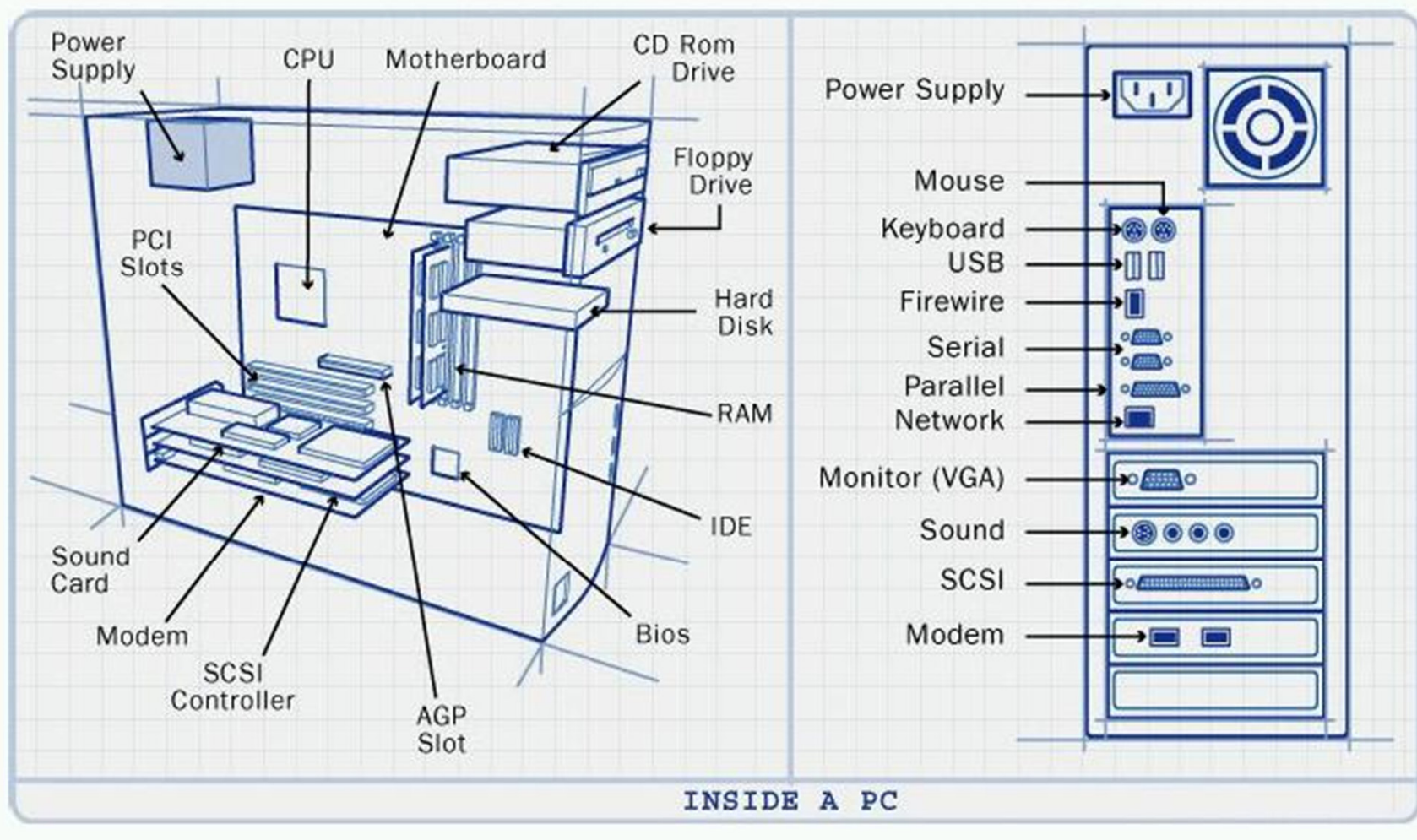
# ENVIRONNEMENT ET PENSÉE INFORMATIQUES

POLYTECHNIQUE  
MONTRÉAL

LE GÉNIE  
EN PREMIÈRE CLASSE

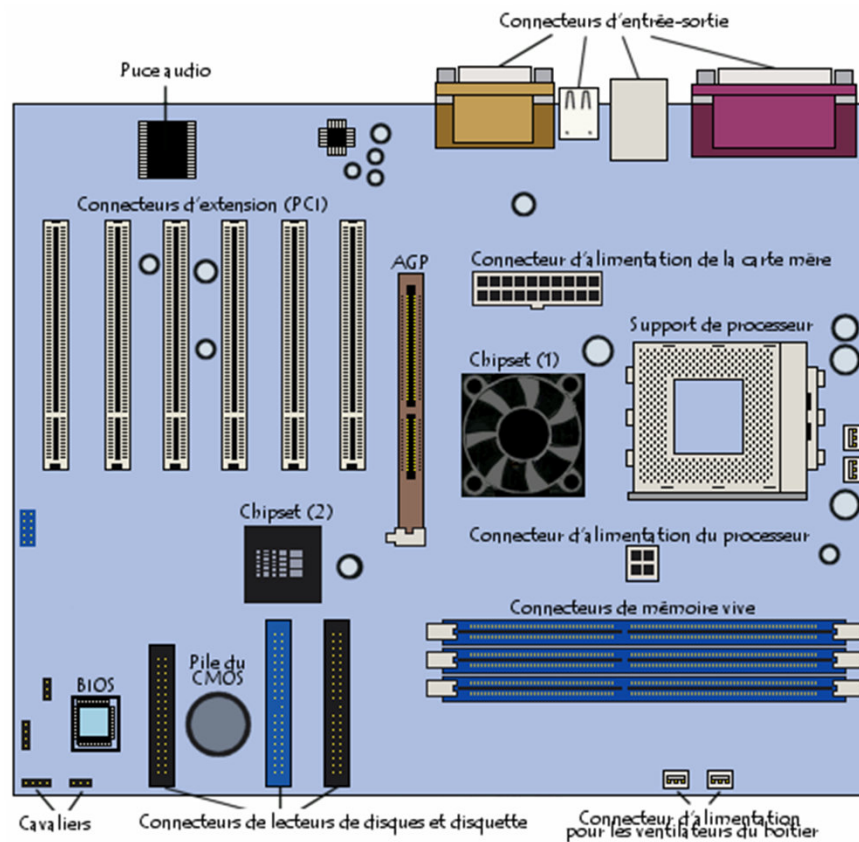


# L'ordinateur





# Carte mère



# Processeur

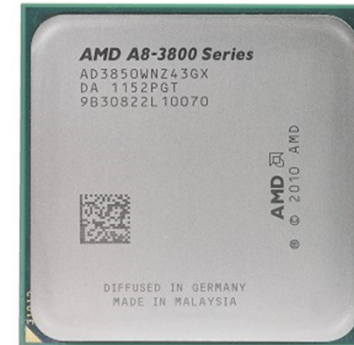
AMD



K5  
(IA-32; 1996)



Athlon 64  
(AMD64; 2003)



A8  
(2011...)

Intel



80386  
(IA-32; 1985)



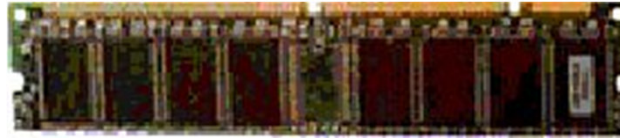
Pentium 4  
(x86-64; 2004)



Core i5  
(2009...)

(pas à l'échelle)

# Mémoire vive



La **SDRAM** (synchronous dynamic random access memory) permet une lecture des données **synchronisée avec le bus de la carte-mère**.

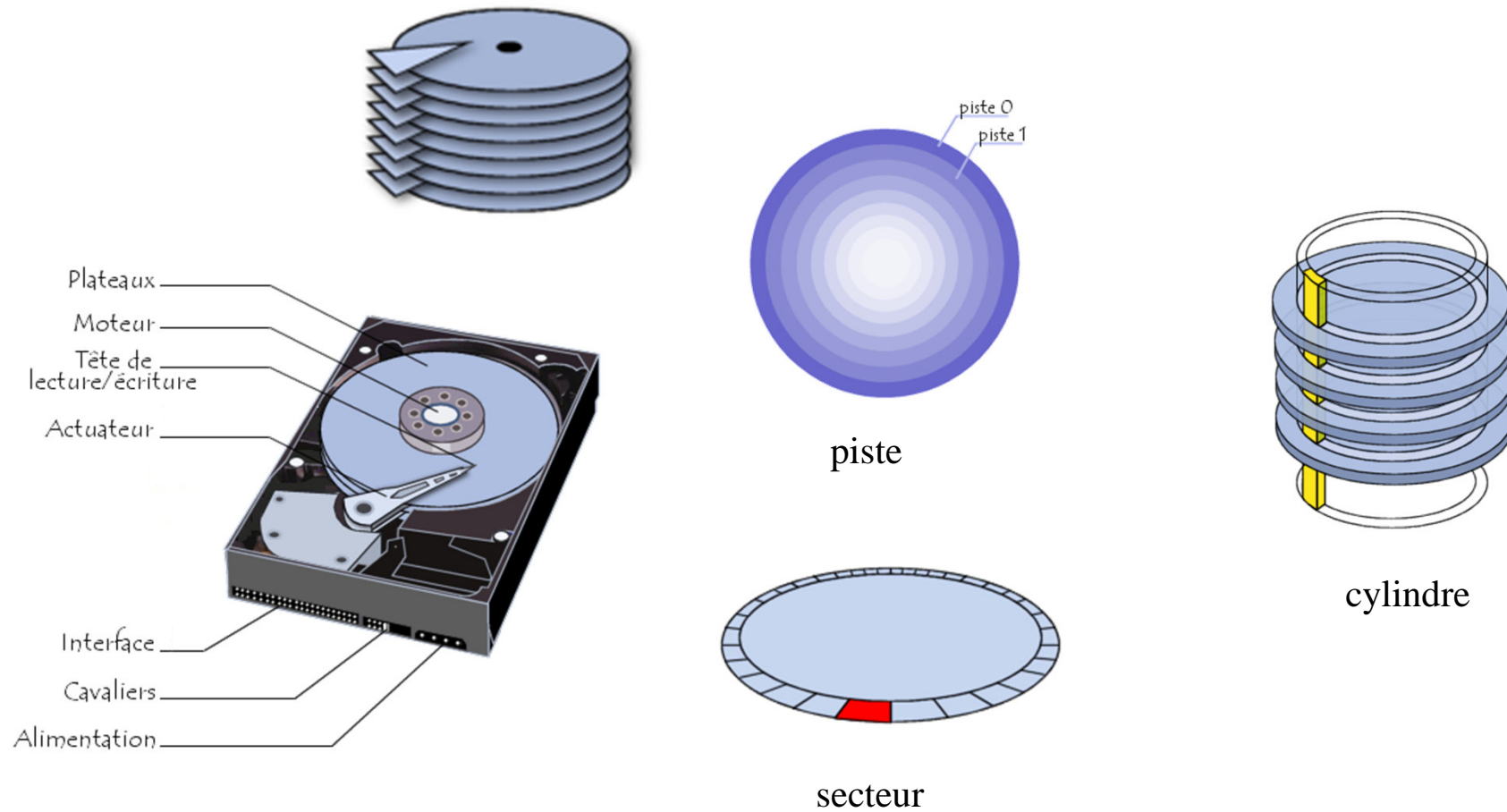


La **DDR** (ou **SDRAM DDR pour Double Data Rate**) permet de doubler le taux de transfert



La **DR-SDRAM** (Direct Rambus DRAM ou encore **RDRAM pour Rambus DRAM**) permet de transférer les données à 1.6 Go par seconde par l'intermédiaire d'un bus de 16 bit de largeur avec une fréquence de 800 MHz.

# Disque dur





## Autres composants



Cartes graphiques



Carte réseau



Carte de son

# L'informatique

- ❑ L'informatique se définit comme la science du traitement automatique de l'information.
- ❑ Le traitement de l'information se résume aux opérations suivantes:



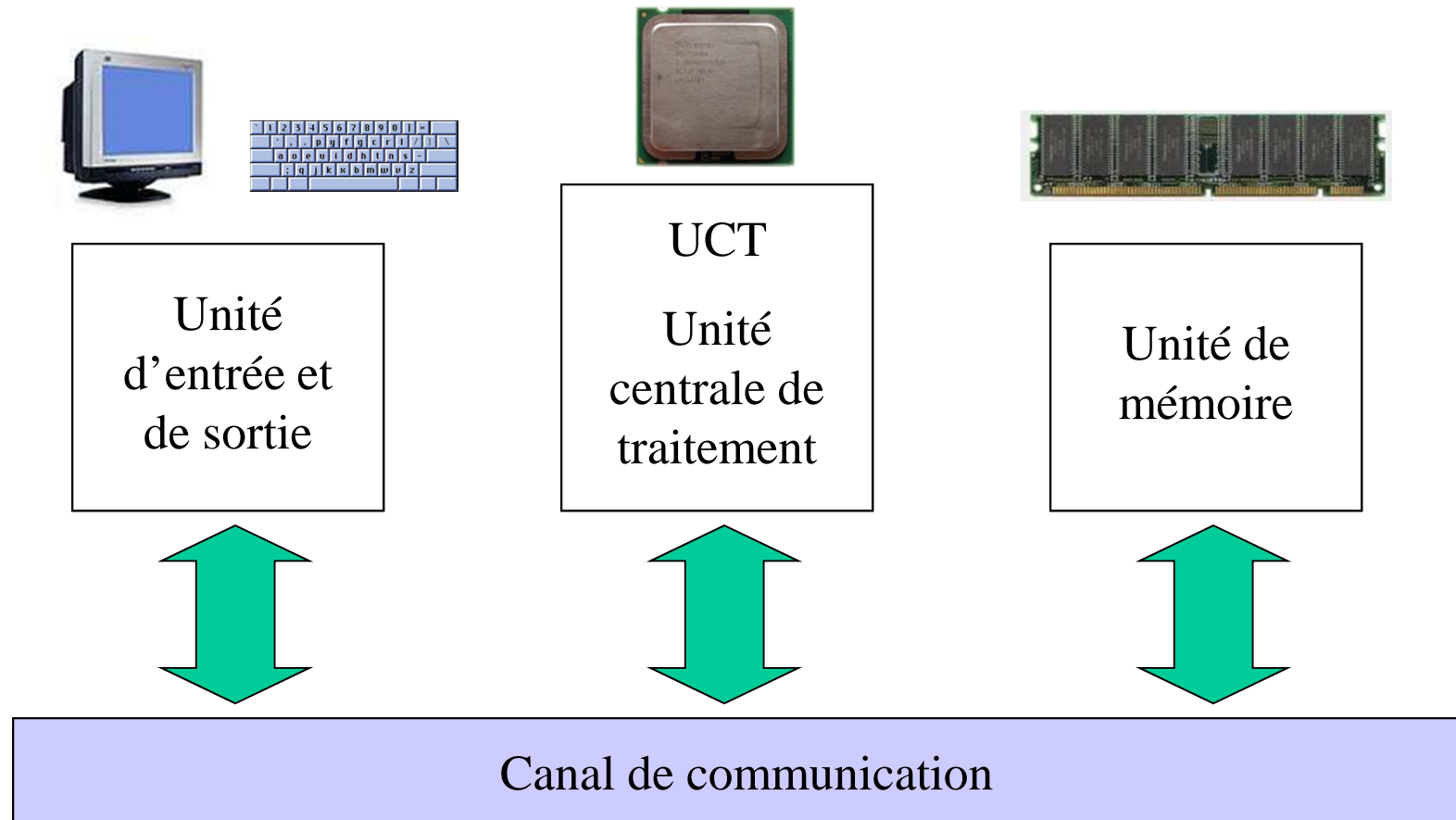


# L'informatique (suite)

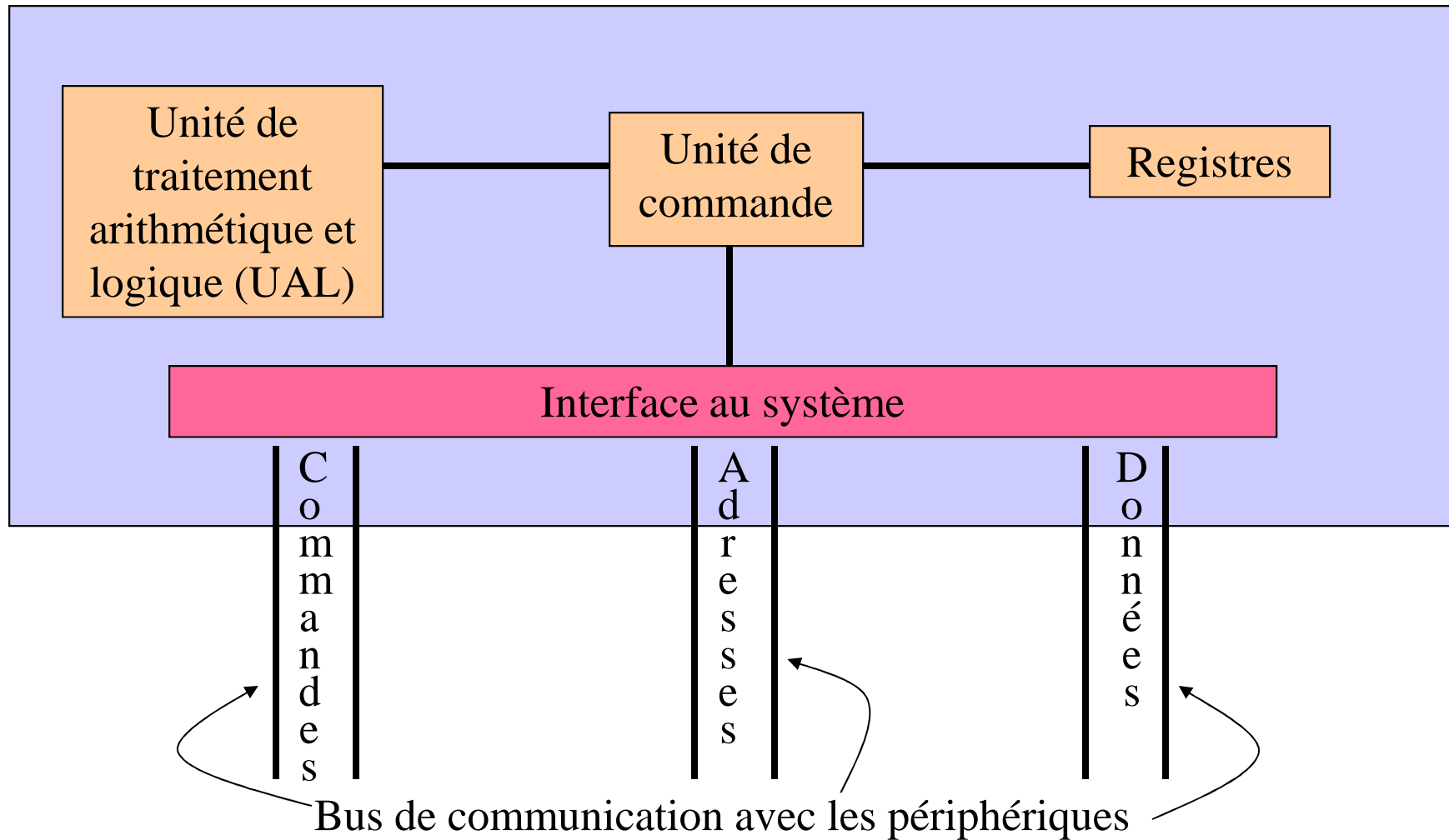
- ❑ Du point de vue matériel, le cheminement de l'information s'effectue à travers les unités suivantes:



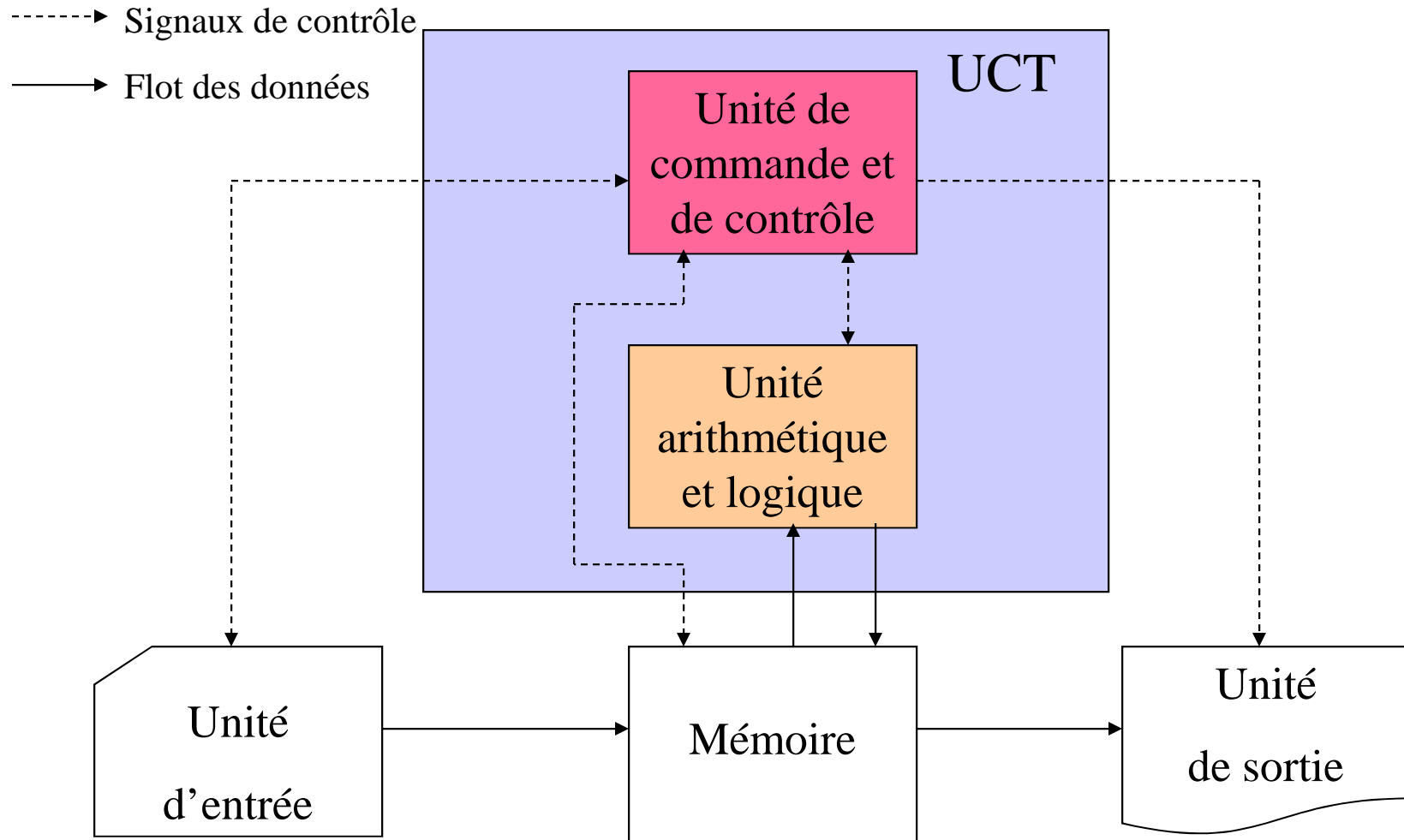
# Architecture de base



# Unité centrale de traitement

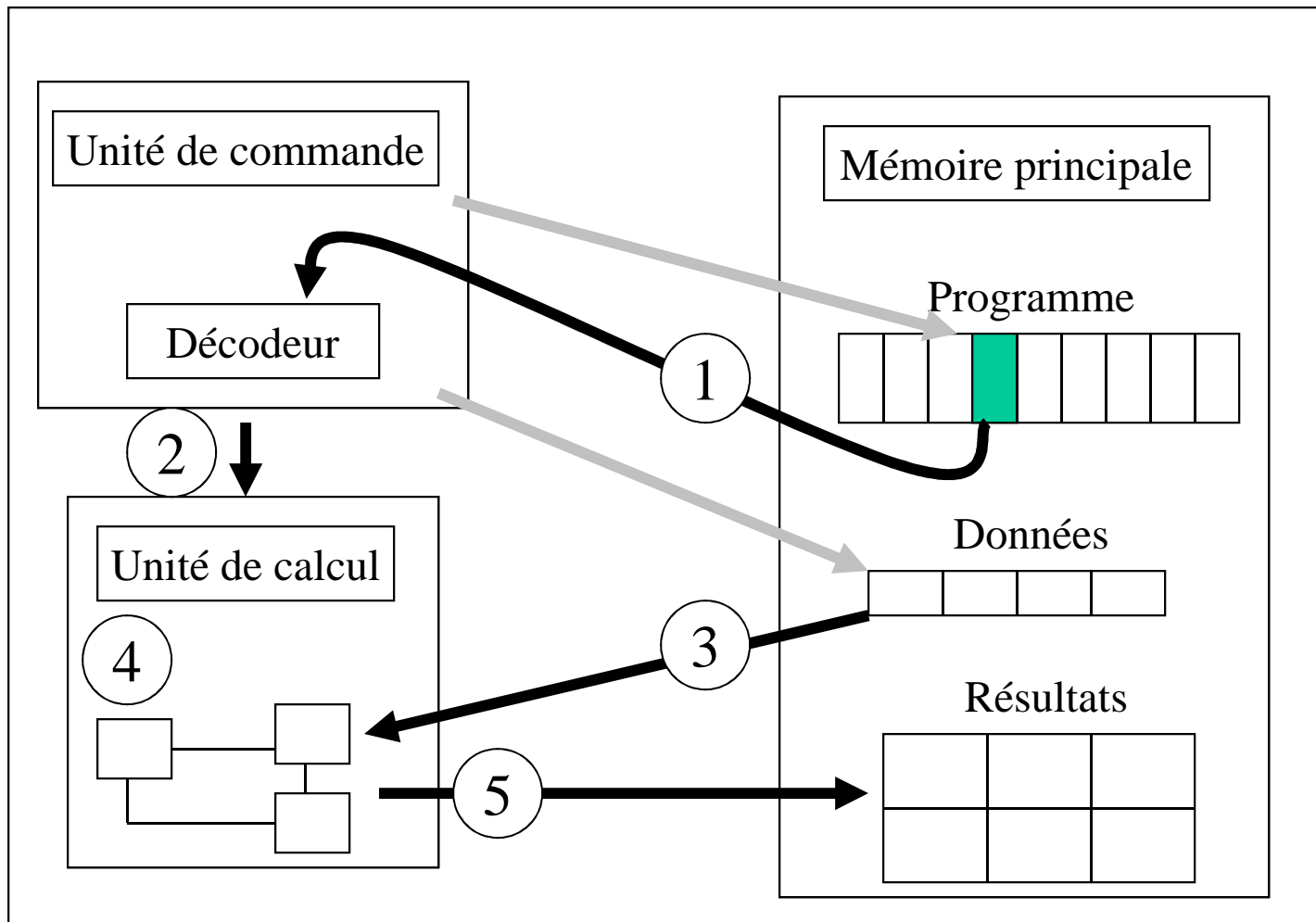


# Composantes d'un ordinateur





# Fonctionnement UCT



## Fonctionnement de l'UCT

- ① L'instruction est chargée dans l'unité de commande à partir de la mémoire principale.
- ② L'unité de commande décode l'instruction et envoie un ordre de préparation à l'unité de calcul.
- ③ L'unité de commande déclenche le transfert des données à traiter vers l'unité de calcul.
- ④ L'unité de calcul effectue le traitement.
- ⑤ Le résultat du calcul est rangée en mémoire principale.

# Mémoire d'un ordinateur

Mot mémoire

Adresse Contenu	→	0	1	2	3	4	5	6	7
	→								
		8	9	10	11	12	13	14	15
		16	17	18	19	20	21	22	23
		24	25	26	27	28	29	30	31
		32	33	34	35	36	37	38	39
		40	41	42	43	44	45	46	47
		48	49	50	51	52	53	54	55
		56	57	58	59	60	61	62	63

# Mémoire d'un ordinateur

 Contenu mémoire d'un programme et ses données.

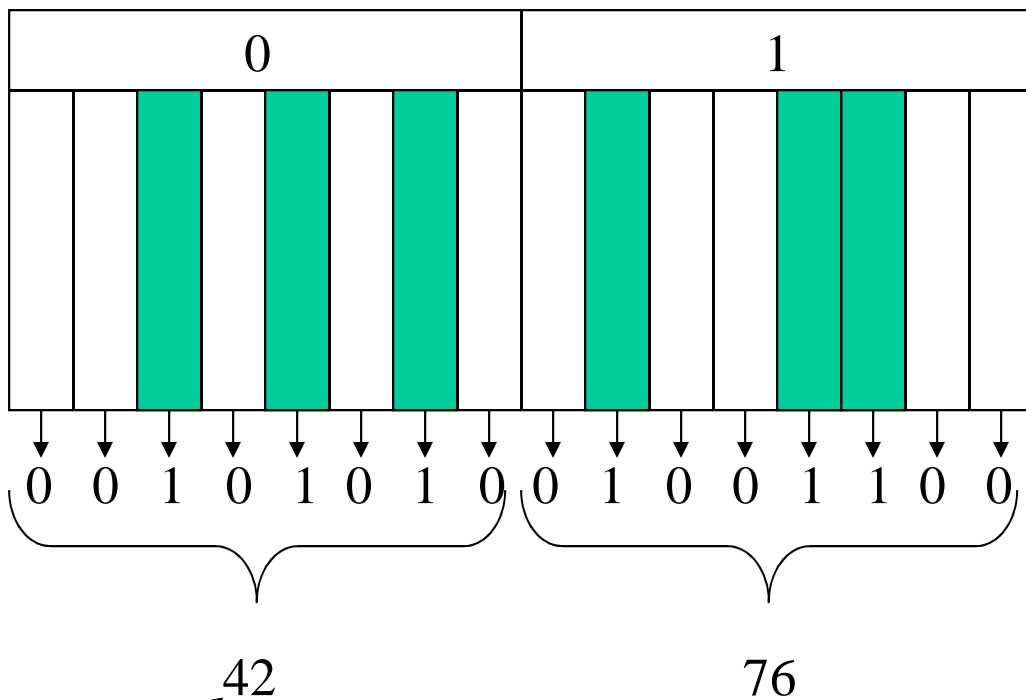
Adresse →	00000	00001	00010	00011
Contenu →	00101010	01001100	00101110	01101011
(en binaire)	00100	00101	00110	00111
	01001110	00101100	10001101	01001100
	01000	01001	01010	01011
	11000010	11100000	00000011	00000100



# Rangement de l'information

Adresses

Position des bascules

Chiffres binaires  
équivalentsValeurs décimales  
équivalentes

$$0 \times 2^0 + 1 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 + 0 \times 2^4 + 1 \times 2^5 = 42$$

# Jeu d'instructions élémentaires

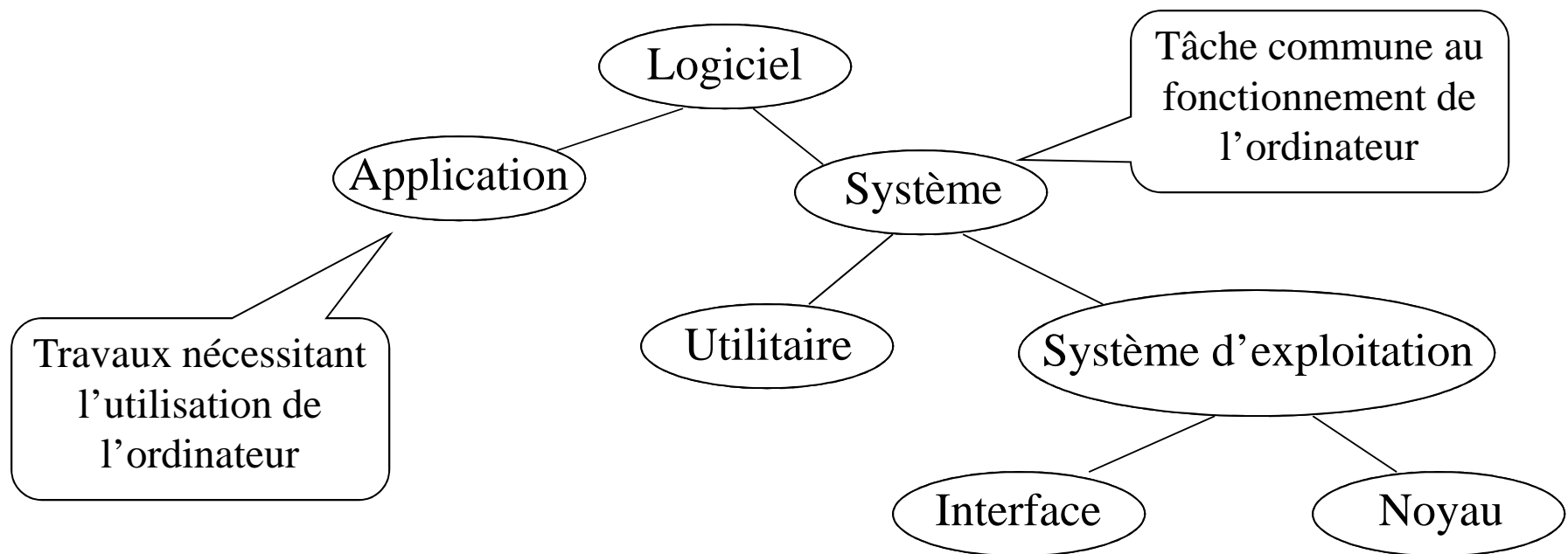
Code	Signification
001	Charger: copie dans l'accumulateur le contenu du mot mémoire.
010	Ranger: copie le contenu de l'accumulateur dans le mot-mémoire adressé.
011	Additionner: remplace le contenu de l'accumulateur par la somme de son présent contenu et du contenu du mot-mémoire adressé.
100	Soustraire: remplace le contenu de l'accumulateur par la différence de son présent contenu et du contenu du mot-mémoire adressé.
101	Sauter: va à l'instruction dont on fournit l'adresse.
110	Sauter si non zéro: va à l'instruction dont on fournit l'adresse seulement si le présent contenu de l'accumulateur n'est pas zéro.
111	Stop: Arrêt de l'exécution du programme

# Programme

Adresse des instructions (en binaire)	Code d'opération	Opérande
00000	001	01010
00001	010	01100
00010	001	01110
00011	011	01011
00100	010	01110
00101	001	01100
00110	100	01101
00111	010	01100
01000	110	00010
01001	111	00000

# Le système d'exploitation

❑ Exploiter les ressources matérielles et logicielles d'un environnement informatique





# Système d'exploitation (s.e.)

## ▣ Application

▣ Éditeur de texte, chiffrier électronique, environnement de développement

## ▣ Utilitaire

▣ Formatage de disque, copier un fichier, affichage de l'heure

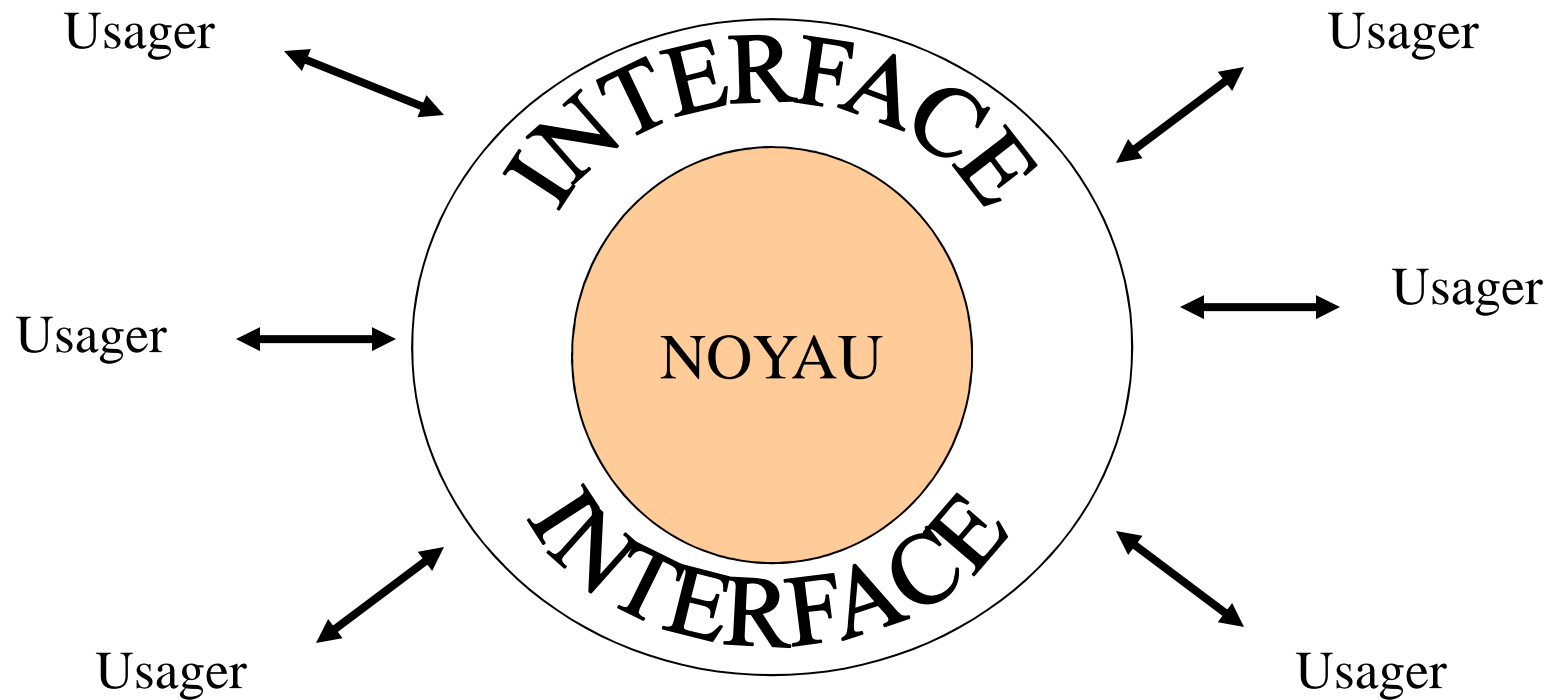
## ▣ Interface

▣ Gestion de l'interaction entre l'utilisateur et les fonctions du s.e.

## ▣ Noyau

▣ Fonctions du s.e.: gestion des fichiers, pilote de périphérique, gestion mémoire...


# Système d'exploitation



# Systeme d'exploitation

## FICHER

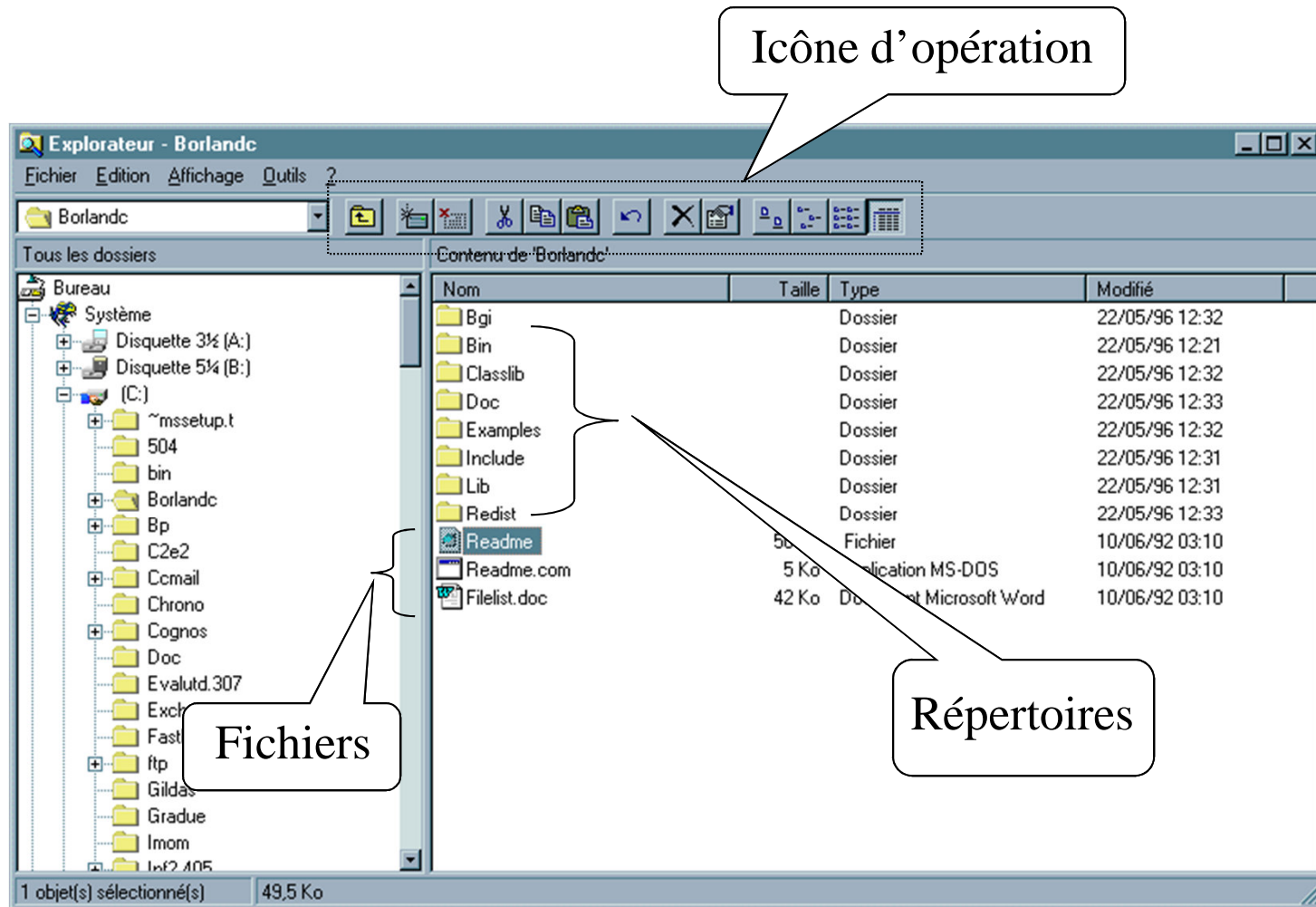
### Définition

-  structure qui permet de mémoriser de manière permanente un ensemble d'information de nature commune avec une représentation unique.

### Spécification

-  correspond à identifier, à nommer le fichier; ce qui permettra son repérage par la suite.
-  Lecteur:NomFichier.Ext

# Système d'exploitation

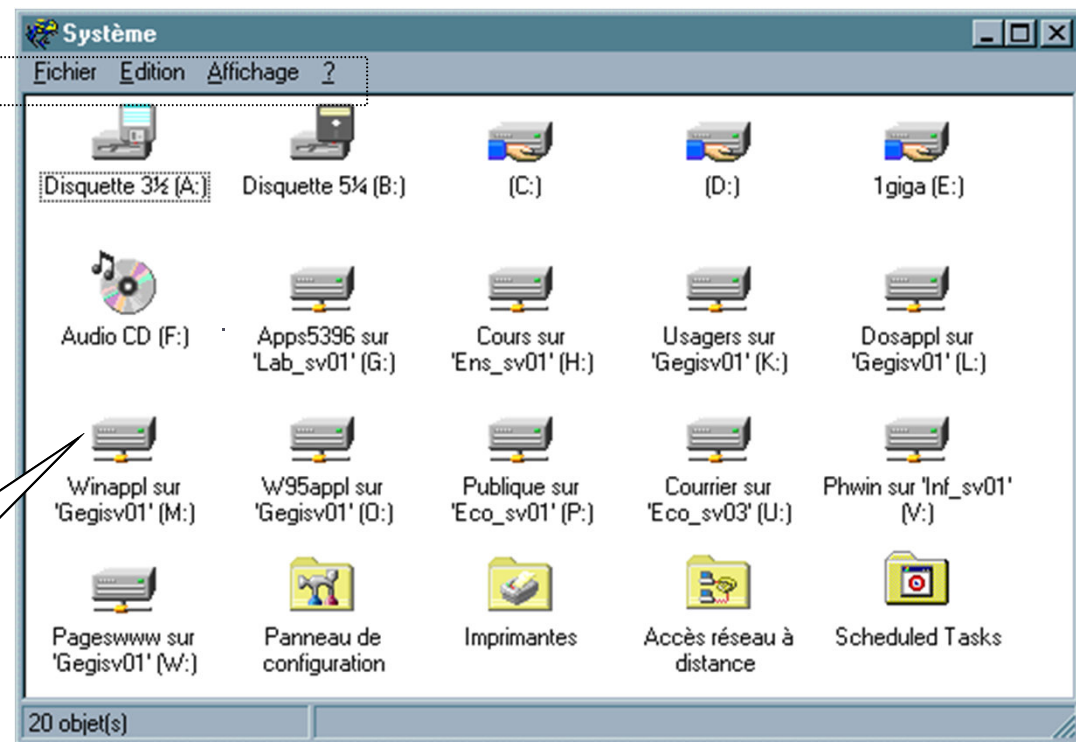




# Système d'exploitation

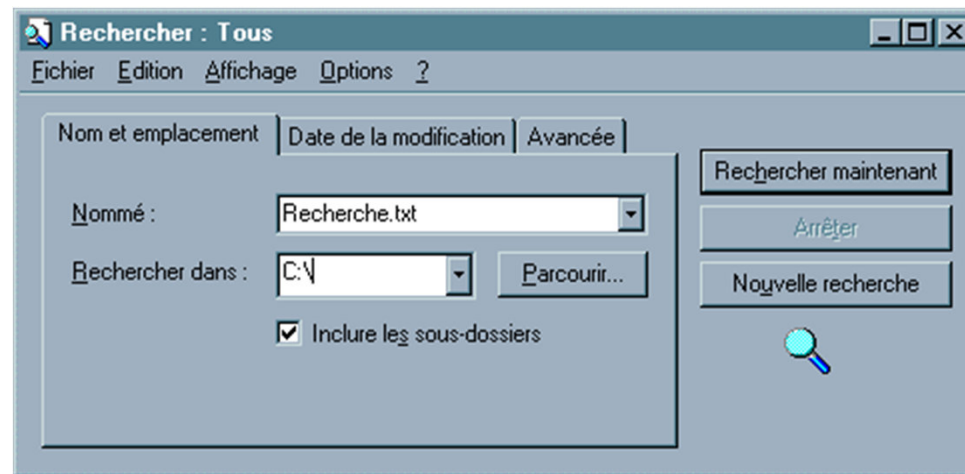
Menu déroulant  
contextuel

Icône à sélectionner  
pour lancer  
l'application désirée



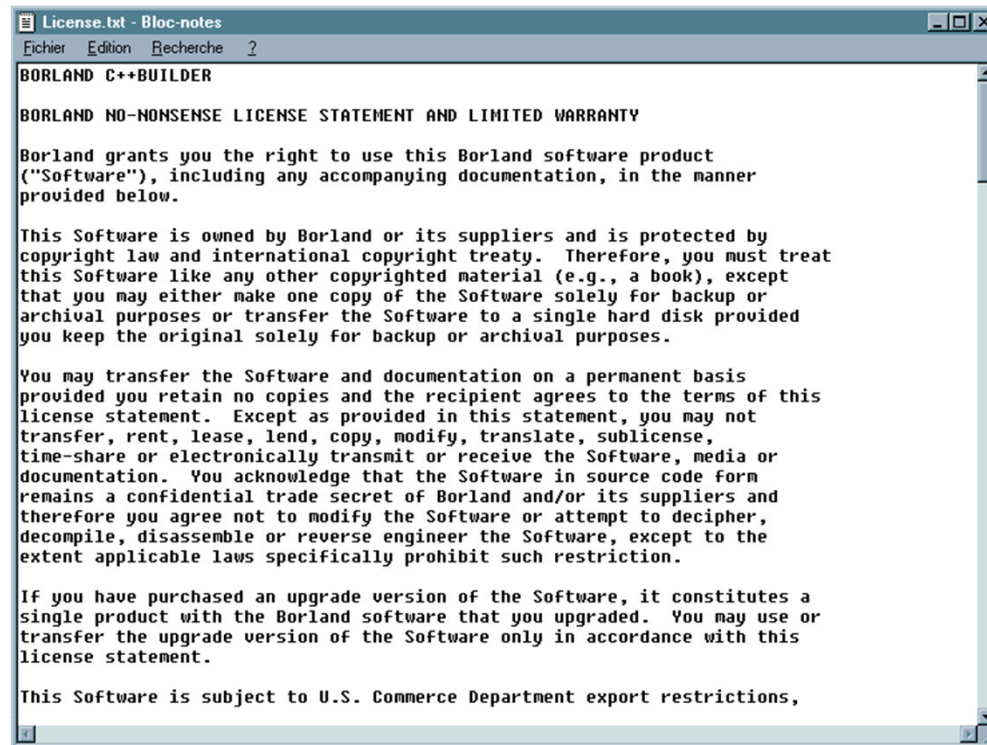
# Systeme d'exploitation

## Recherche d'un fichier



# Systeme d'exploitation

## Affichage du contenu d'un fichier



```
License.txt - Bloc-notes
Fichier Edition Recherche ?

BORLAND C++BUILDER

BORLAND NO-ONSENSE LICENSE STATEMENT AND LIMITED WARRANTY

Borland grants you the right to use this Borland software product
("Software"), including any accompanying documentation, in the manner
provided below.

This Software is owned by Borland or its suppliers and is protected by
copyright law and international copyright treaty. Therefore, you must treat
this Software like any other copyrighted material (e.g., a book), except
that you may either make one copy of the Software solely for backup or
archival purposes or transfer the Software to a single hard disk provided
you keep the original solely for backup or archival purposes.

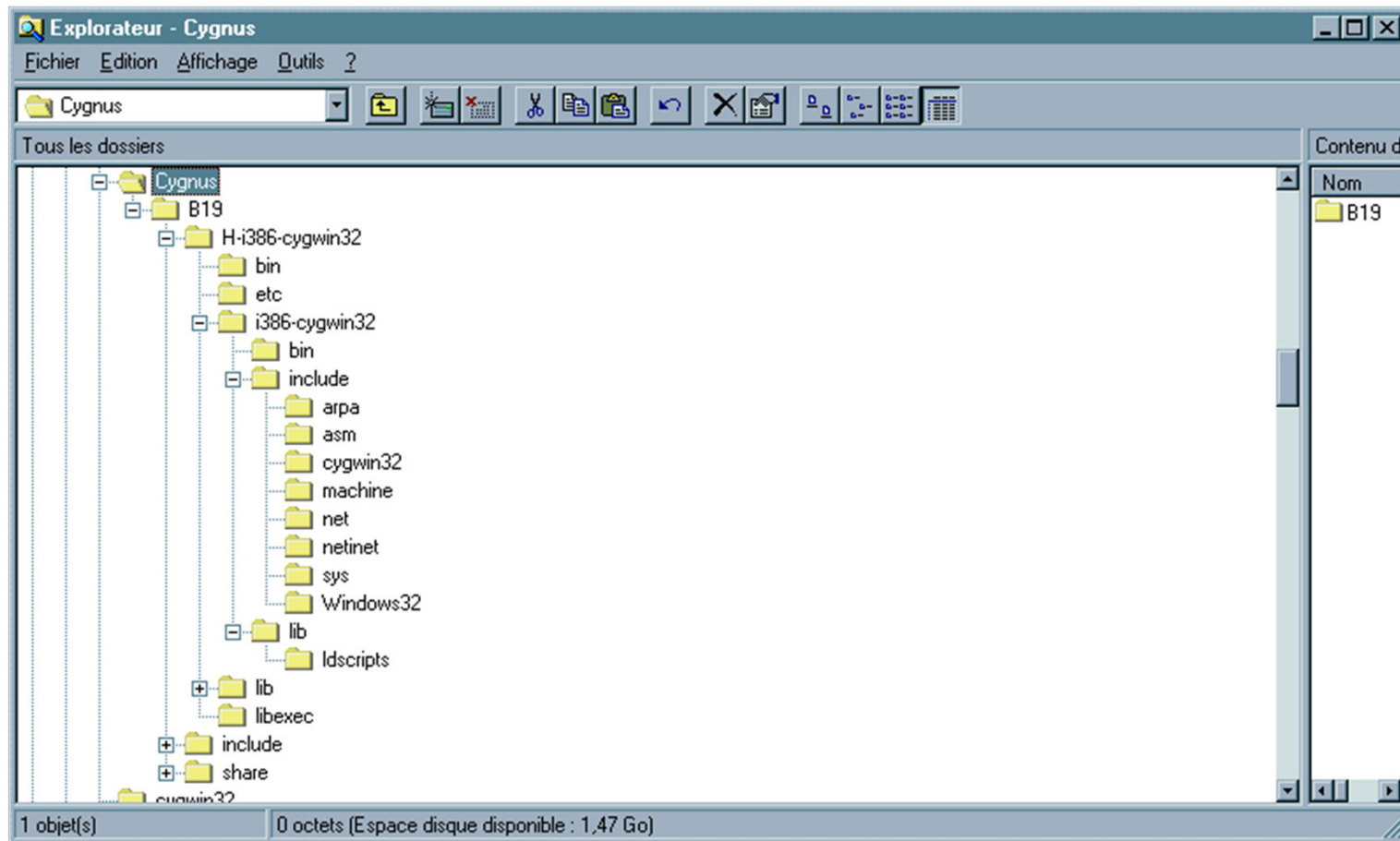
You may transfer the Software and documentation on a permanent basis
provided you retain no copies and the recipient agrees to the terms of this
license statement. Except as provided in this statement, you may not
transfer, rent, lease, lend, copy, modify, translate, sublicense,
time-share or electronically transmit or receive the Software, media or
documentation. You acknowledge that the Software in source code form
remains a confidential trade secret of Borland and/or its suppliers and
therefore you agree not to modify the Software or attempt to decipher,
decompile, disassemble or reverse engineer the Software, except to the
extent applicable laws specifically prohibit such restriction.

If you have purchased an upgrade version of the Software, it constitutes a
single product with the Borland software that you upgraded. You may use or
transfer the upgrade version of the Software only in accordance with this
license statement.

This Software is subject to U.S. Commerce Department export restrictions,
```

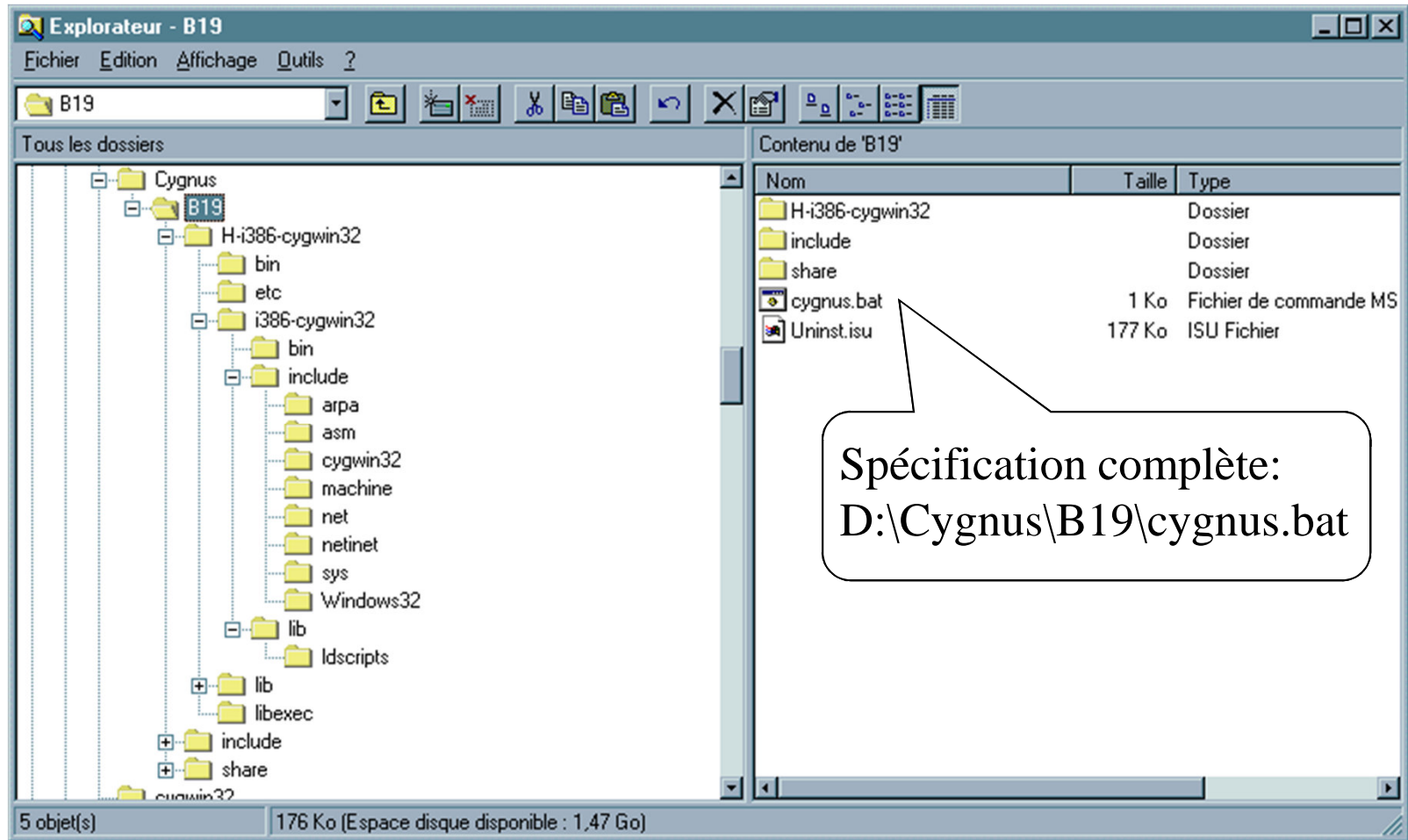
# Système d'exploitation

## 📁 Les répertoires



# Système d'exploitation

## Les répertoires



# Fichier VS Répertoire



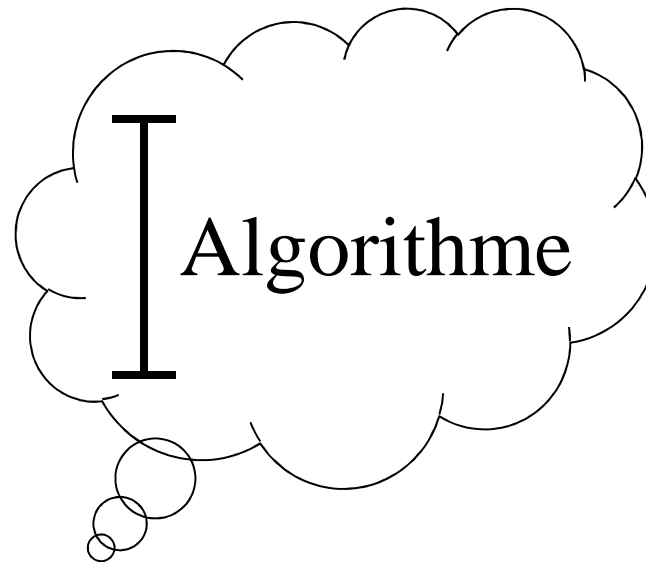
## Fichier

- Rechercher
- Accéder
- Renommer
- Copier
- Détruire
- Déplacer
- Visualiser



## Répertoire

- Rechercher
- Accéder
- Renommer
- Copier
- Détruire
- Déplacer
- Visualiser



# Pensée informatique

# Pensée informatique

## 🕒BUT

- 📄 Décrire une tâche de sorte qu'elle soit facilement programmable

## 🕒Moyens

- 📄 Identifier les différentes opérations et préciser la séquence qui exprime la tâche.
- 📄 Préciser les répétitions en regroupant les opérations à répéter.
- 📄 Préciser les décisions en regroupant les opérations dépendantes d'une condition à vérifier.



# Pensée informatique

## 🔍 Opérations élémentaires

- Nous décrirons les tâches à partir des opérations élémentaires suivantes:

- 
- |                                |               |
|--------------------------------|---------------|
| ➤ Lire                         | ➤ Additionner |
| ➤ Afficher                     | ➤ Soustraire  |
| ➤ TANTQUE condition FAIRE      | ➤ Multiplier  |
| ➤ SI condition ALORS --- SINON | ➤ Diviser     |
| ➤ Affecter =                   |               |
-

## Pensée informatique

🎧 Lire un nombre et afficher la valeur absolue de ce nombre

Afficher "Entrer le nombre"

Lire le nombre

SI le nombre < 0 ALORS

Afficher  $-1 * \text{nombre}$

SINON

Afficher le nombre

Guillemets pour indiquer un texte à afficher tel quel.

Décalage pour démontrer la dépendance du test  
SI nombre < 0

Sans guillemets pour afficher la valeur du nombre.

# Pensée informatique

## 🔊 Séparation du traitement et de la présentation

Afficher "Entrer un nombre "

Lire le nombre

} Collecte des  
données

SI le nombre  $< 0$  ALORS

    valeurAbsolue =  $-1 * \text{nombre}$

SINON

    valeurAbsolue = nombre

} Traitement  
des  
données

Présentation  
des résultats

Afficher "La valeur absolue du nombre est "

Afficher valeurAbsolue

# Pensée informatique

## 📷 Deviner un nombre choisi au hasard (Version 1)

Afficher "Entrer le nombre à deviner"

Lire le nombre à deviner

Afficher "Entrer un nombre"

Lire un nombre

TANT QUE nombre  $\neq$  nombre à deviner

SI nombre < nombre à deviner

Afficher "Trop petit"

SINON

Afficher "Trop grand"

Afficher "Entrer un nombre"

Lire un nombre

Afficher "BRAVO!!!"

Décalage pour  
démontrer la  
dépendance du test  
du TANT QUE

# Pensée informatique

☞ Deviner un nombre choisi au hasard (Version 2)

Afficher "Entrer le nombre à deviner "

Lire le nombre à deviner

deviné = Faux

TANT QUE deviné est Faux FAIRE

Afficher "Entrer un nombre "

Lire un nombre

SI nombre = nombre à deviner ALORS

deviné = Vrai

SINON

SI nombre < nombre à deviner ALORS

Afficher "Le nombre est trop petit"

SINON

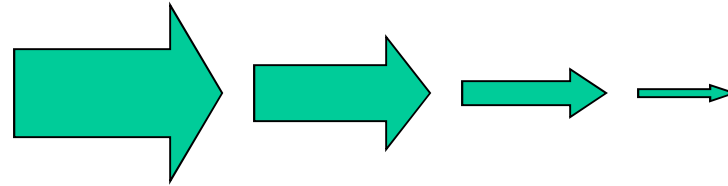
Afficher "Le nombre est trop grand"

Afficher " BRAVO, vous avez trouvé le nombre à deviner "

Lignes verticales parfois utiles pour bien montrer l'alignement

# Pensée informatique

## Raffinement graduel



⇒ Lors de l'élaboration d'un algorithme il est parfois utile d'envisager un processus de raffinement graduel.

⇒ Il s'agit dans un premier temps d'énumérer les opérations « évidentes » qui réalisent l'application désirée.

⇒ Ces opérations évidentes ne sont pas nécessairement des opérations élémentaires.

⇒ Par la suite il suffit de détailler les opérations non élémentaires à l'aide d'opérations élémentaires.

# Pensée informatique

- Lire trois nombres entiers quelconques et vérifier si l'un de ces nombres est le produit des 2 autres. Par exemple, si 2, 10 et 5 sont lus alors le deuxième nombre est le résultat de la multiplication des deux autres.

Afficher "Entrer trois nombres"

Lire les trois nombres

Vérifier si l'un des nombres est le produit des deux autres

Afficher le résultat " est le produit des deux autres"

Opération à raffiner

## Suite

Vérifier si l'un des nombres est le produit de deux autres :

SI le premier = le deuxième \* le troisième ALORS

le résultat = "Le premier nombre"

Le résultat est le texte,  
pas la valeur du nombre.

SINON

SI le deuxième = le premier \* le troisième ALORS

le résultat = "Le deuxième nombre"

SINON

SI le troisième = le premier \* le deuxième ALORS

le résultat = "Le troisième nombre"

SINON

le résultat = "Aucun des trois nombres"



*Chapitre 16 (9)*

# MÉTHODOLOGIE DE LA RÉOLUTION DE PROBLÈMES

POLYTECHNIQUE  
MONTREAL

LE GÉNIE  
EN PREMIÈRE CLASSE



## Les étapes de la méthode de résolution de problèmes

- ↪ La définition du problème
- ↪ L'analyse du problème
- ↪ La conception des algorithmes
- ↪ La mise au point des programmes
- ↪ La rédaction du rapport

# La définition du problème

Répondre à QUI? QUE? QUOI?

- ⇒ Identifier les entités à manipuler
- ⇒ Identifier les traitements à effectuer
- ⇒ Identifier les résultats attendus

## L'analyse du problème

Répondre à COMMENT?

- ⇒ Inventer et comparer diverses solutions possibles.
- ⇒ Nous cherchons deux solutions pour le même problème de sorte à juger de leur qualité et leur validité en les comparant.

# Analyse

Voici trois solutions du problème: déterminer les nombres premiers compris entre 50 et 100.

- ① Pour chaque nombre de 50 à 100 on vérifie tous les diviseurs possibles, c.à d. de 2 à lui-même. Si on trouve une division sans reste, ce n'est pas un nombre premier.
- ② Pour chaque nombre de 50 à 100 on vérifie les diviseurs possibles, mais du moment qu'une division est sans reste, on conclut toute suite que ce nombre n'est pas premier.
- ③ Pour chaque nombre de 50 à 100 on vérifie les diviseurs de deux jusqu'à la racine carrée de lui-même. Cependant si une division est sans reste, on conclut toute suite que ce nombre n'est pas premier.

## L'analyse

Identifier pour ces deux problèmes le plus de solutions possibles.

- ☐ Déterminer si deux mots sont des anagrammes.  
Niche, chine et chien sont trois anagrammes.
- ☐ Déterminer si trois points forment un triangle.

# La conception des algorithmes

- Écrire dans l'ordre les opérations déduites de l'analyse.
- Faire ressortir la structure de base
  - les structures de décision
  - les structures de répétition
- Utiliser adéquatement le pseudo-code schématique.

# La mise au point des programmes

- Faire en sorte que le programme fonctionne bien.
- Quatre types d'erreur
  - syntaxe
  - logique
  - de données
  - manipulation



# La rédaction du rapport

Lors de l'élaboration des étapes précédentes il est essentiel de produire simultanément une documentation complète à la fois claire et précise. Comme la méthode de résolution est un processus itératif, cette documentation facilitera la poursuite des travaux.

