

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique

Université Abderrahmane Mira

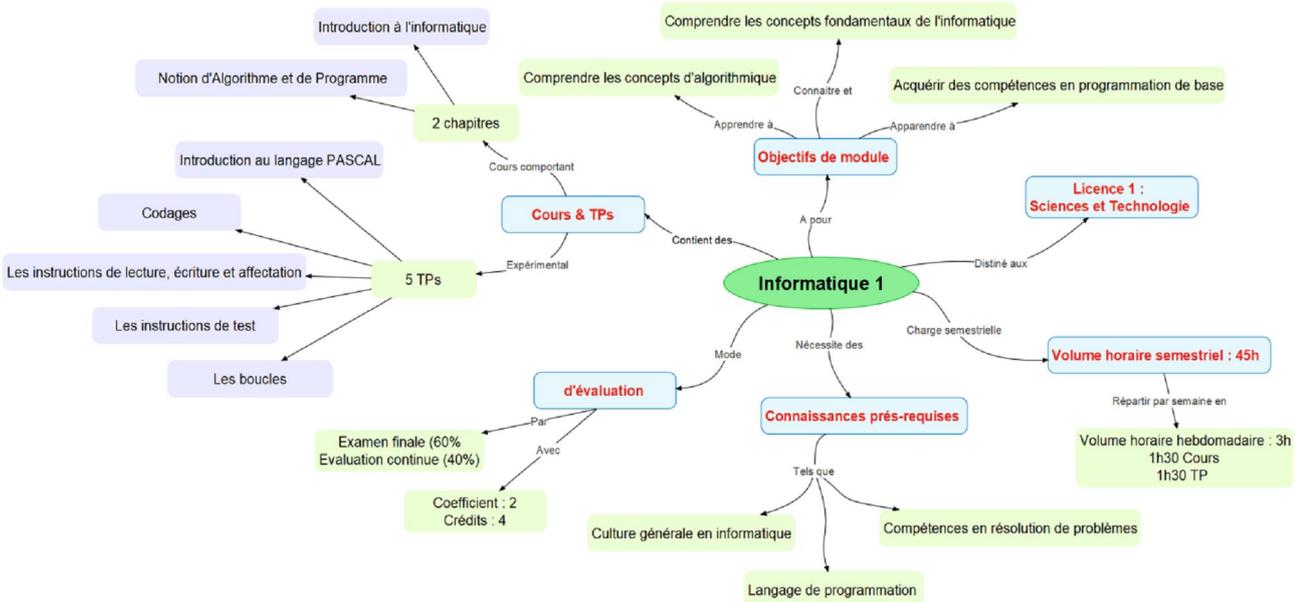
Faculté de la Technologie

Département de Technologie



## Support du cours

# Informatique 1



Enseignant :  
Dr. OUARET Ahmed

Année Universitaire : 2024/2025

## Table des matières

### Chapitre I : Introduction à l'informatique

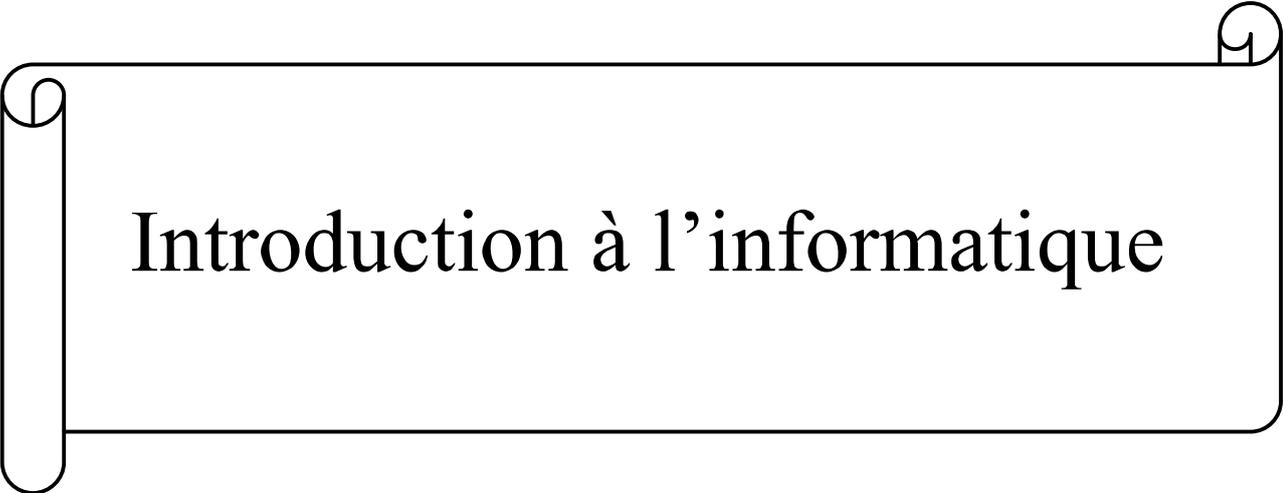
I.1. Objectif de ce chapitre .....	1
I.2. Définition de l'informatique .....	1
I.2.1. Role de l'informatique .....	1
I.3. Evolution de l'informatique et des ordinateurs .....	1
I.3.1. Evolution de l'informatique .....	1
I.3.2. Evolution des ordinateurs .....	2
I.3.2.1. Définition de l'ordinateur .....	2
I.3.2.2. Générations des ordinateurs .....	2
I.4. Architecture et fonctionnement d'un ordinateur .....	3
I.5. Composition d'un ordinateur (Matériel & Logiciel) .....	5
I.5.1. Partie matériel (Hardware) .....	5
I.5.2. Partie logiciel (Software) .....	8
I.6. Les systèmes de codage des informations .....	9
I.6.1. Les systèmes de numérotation .....	9
I.6.2. Conversion d'un nombre d'un système à un autre .....	11
I.6.3. Le code D.C.B. (Decimal coded Binary – Décimal codé binaire) .....	15
I.6.4. La codification Alphanumériques .....	15
I.7. Les expressions arithmétiques, relationnelles et logiques .....	16
I.7.1. Expressions arithmétiques .....	16
I.7.2. Expressions relationnelles .....	16
I.7.3. Expressions logiques (booléennes) .....	17
I.7.4. Les fonctions .....	17
I.7.5. La priorité dans les expressions .....	18
I.7.6. Evaluation des expressions .....	18
I.8. Exercices corrigés .....	20
I.9. Exercices supplémentaires .....	23

### Chapitre II : Notion d'algorithme et de programme

II.1. Objectif de ce chapitre .....	25
II.2. Concept d'un algorithme .....	25
II.3. La démarche et analyse d'un problème .....	26

II.4. Structures d'un algorithme .....	27
II.4.1. L'Entête .....	28
II.4.2. La partie déclarative .....	28
II.4.3. Le corps de programme .....	30
II.5. Types d'instructions .....	30
II.5.1. Instructions d'Entrées/Sorties (Lecteur/Ecriture) .....	30
II.5.1.1. Entrées (Lecteur) .....	30
II.5.1.2. Sorties (Ecriture) .....	31
II.5.2. Instruction d'affectation .....	32
II.5.3. Structures de contrôles .....	33
II.5.3.1. Structures de contrôle conditionnelle .....	33
II.5.3.2. Structure de contrôle répétitives .....	35
II.5.3.3. Structure de contrôle de branchements/sauts (l'instruction Goto) .....	37
II.6. Correspondance Algorithme-Pascal .....	39
II.7. Représentation en organigramme .....	41
II.7.1. Les symboles d'organigramme .....	41
II.8. Représentation des primitives algorithmiques .....	41
II.8.1. L'enchaînement .....	41
II.8.2. La structure alternative simple .....	42
II.8.3. La structure alternative double .....	43
II.8.4. La structure itérative POUR (Boucle POUR) .....	43
II.8.5. La structure itérative Tant-Que (Boucle Tant-Que) .....	44
II.8.6. La structure itérative Répéter (Boucle répéter) .....	44
II.9. Exercices corrigés .....	46
II.10. Exercices supplémentaires .....	55

# CHAPITRE I



Introduction à l'informatique

## Chapitre I : Introduction à l'informatique

### I.1. Objectif de ce chapitre

A l'issue de ce chapitre, l'apprenant sera capable de :

- Se familiariser avec l'environnement informatique
- Identifier les principaux composants d'un ordinateur
- Connaître les systèmes de codage des informations
- Apprendre comment évaluer les expressions arithmétiques et logiques

### I.2. Définition de l'Informatique

L'informatique est la **science** qui s'occupe du **traitement automatique** de l'**information**. Le terme "informatique" vient de la contraction des mots "information" et "automatique".

#### I.2.1. Rôle de l'informatique

L'informatique a pour rôle :

- La conception et la construction des ordinateurs;
- Le fonctionnement et la maintenance des ordinateurs;
- Leur exploitation (utilisation des ordinateurs dans les différents domaines d'activités).

### I.3. Evolution de l'informatique et des ordinateurs

#### I.3.1. Evolution de l'informatique

L'histoire de l'informatique est justement marquée par la volonté des hommes d'automatiser certaines tâches longtemps réalisées à la main, en particulier le calcul.

Le Tableau I.1 représente les différentes évolutions

**Tableau I.1** : Evolution de l'informatique

Date	Inventeur	Machine	Fonction
1642	Blaise Pascal	Pascaline	Effectue des additions et soustractions (des nombres de six chiffres)
1822-1832	Charles Babbage	Machine à Différences ( <i>Difference Engine</i> )	Effectue des calculs numériques compliqués (Tables numériques).
1887	Hermann Hollerith	Machine mécanographique	Accélérer le traitement des données lors du recensement

		(Calculateur de statistiques à cartes perforées)	américain de 1890 (invention de la carte perforée).
<b>1945 (inauguré en 1946)</b>	John Presper Eckert & John William Mauchly	“ENIAC” Electronic Numerical Integrator and Computer	Premier calculateur numérique électronique programmable. 18.000 tubes, 30 tonnes. Multiplication de deux nombres de 10 chiffres signés en moins de millisecondes.
	John Von Neumann	“EDVAC” Electronic Discrete Variable Automatic Computer	Enregistrer le programme mémoire.

### I.3.2. Evolution des ordinateurs

#### I.3.2.1. Définition de l'ordinateur

Un ordinateur est une **machine automatique programmable**, utilisée pour le **traitement de l'information**.

#### I.3.2.2. Générations des ordinateurs

- **1ère Génération (1945 - 1957)** : C'est les ordinateurs construits sur la base des “*tubes électroniques*” (aussi appelés “*tubes à vide*”). **Exemple** : ENIAC (1945) et IBM 701 (1952).
  - **2ème Génération (1958 - 1963)** : Elle est définie par l'invention du “*Transistor*” en 1947 (le transistor est un composant électronique capable de réguler le courant). Ces ordinateurs sont 100 fois plus rapides que ceux de la 1ère génération et consomment moins d'énergie électrique et sont moins volumineux (occupent moins d'espace). **Exemples**: PDP I (1er ordinateur de 2ème génération (1960)), IBM 7030 (1961).
  - **3ème Génération (1964 - 1971)** : Elle est définie après l'apparition du “*Circuit Intégré*” en 1958, les ordinateurs de cette génération utilisent les transistors et les circuits intégrés. **Exemple**: IBM 360 (1964), DEC PDP8 (1964).
  - **4ème Génération (1971 - 1980)** : Elle survient après l'invention du “*microprocesseur*”. Le premier microprocesseur est fabriqué par la société INTEL en 1971. On parle, désormais, de microordinateurs. **Exemple**: MICRAL 8008 (1973), ALTAIR 8800 (1975).
  - **5ème Génération (1980 - 2000)** : Les ordinateurs entrent dans les foyers, on parle alors d'informatique familiale (c'est la naissance des **ordinateurs personnels (PC)**). **Exemple** : IBM PC (1981), Intel Pentium (1993).
- Aussi, les systèmes d'exploitation MS-DOS et Mac OS font leur apparition.

- **6ème Génération (2000 – aujourd'hui) :** On assiste, dans cette génération, à l'apparition des ordinateurs portables et des réseaux sans fil.

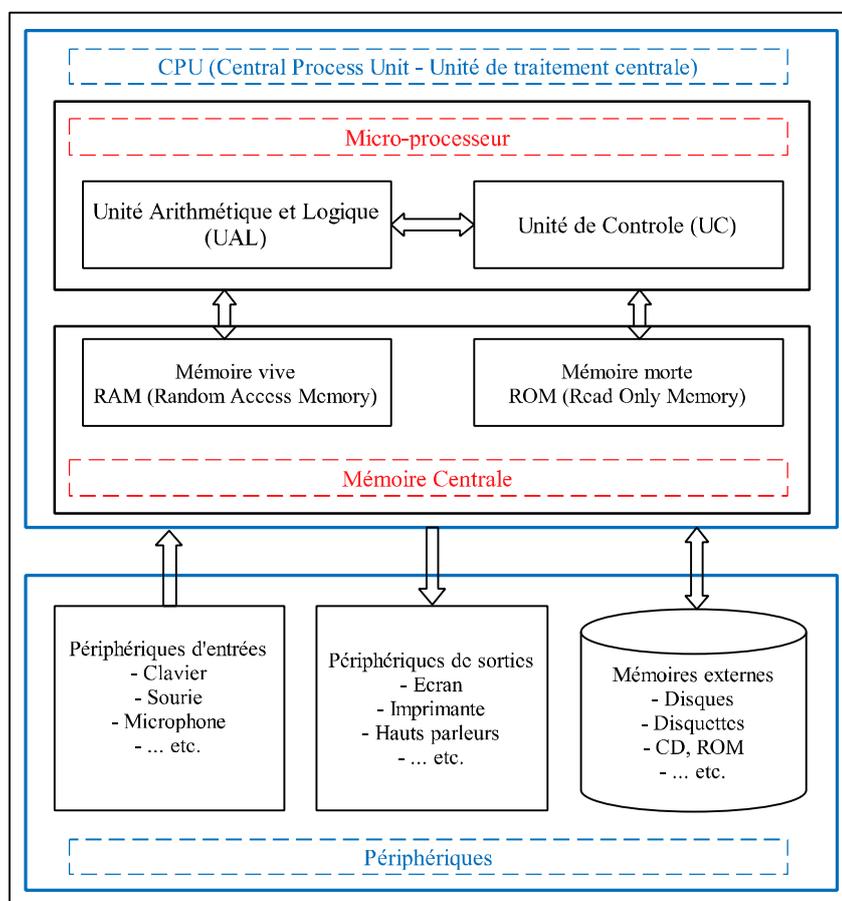
Actuellement l'évolution des ordinateurs tend à exploiter le laser, les fibres optiques, la biochimie et même l'informatique moléculaire.

#### I.4. Architecture et fonctionnement d'un ordinateur

L'architecture et le fonctionnement d'un ordinateur sont des aspects essentiels pour comprendre le système informatique.

L'architecture d'un ordinateur se réfère à la conception globale et à la structure interne de ses composants matériels, tels que le processeur, la mémoire, la mémoire centrale, les périphériques d'entrée/sortie, etc. Cette organisation matérielle permet à l'ordinateur d'exécuter des tâches spécifiques.

La Figure I.1 représente l'architecture générale d'un ordinateur



**Figure I.1 :** Architecture générale d'un ordinateur

**CPU** : Central Process Unit (Unité de traitement centrale)

**UAL** : Unité Arithmétique et Logique

**UC** : Unité de Contrôle ou de commande

**MC** : Mémoire Centrale

Un ordinateur est constitué, d'une manière générale, d'une Unité Centrale de Traitement, d'une mémoire centrale et d'un ensemble de périphériques.

**a) L'Unité Centrale de Traitement (UCT/CPU) ou bien Processeur (Microprocesseur) :**

Le processeur est le cœur de l'ordinateur, chargé de traiter et d'exécuter les programmes (les instructions) de l'ordinateur. Il comporte, essentiellement, deux unités : l'Unité de Contrôle (UC) et l'Unité Arithmétique et Logique (UAL).

✚ **L'unité de Contrôle (UC)** : Elle est responsable de la lecture en mémoire centrale et du décodage des instructions ;

✚ **L'unité Arithmétique et Logique (UAL)** : Aussi appelée unité de traitement, cette dernière exécute les instructions. Elle se charge d'effectuer toutes les opérations arithmétiques, logiques et relationnelles contenues dans l'instruction et effectue aussi des échanges de données (réception/transmission) avec la mémoire vive.

Ces deux unités communiquent, non seulement, avec la mémoire centrale, mais également avec les différents périphériques.

**b) La Mémoire Centrale (ou Principale) :** Elle est constituée d'une mémoire vive (RAM : Random Access Memory) et d'une mémoire morte (ROM : Read Only Memory).

✚ **La RAM** : Une RAM ou mémoire vive sert au stockage temporaire des données et du programme à exécuter. Les mémoires vives sont en général volatiles : elles perdent leurs informations en cas de coupure du courant électrique.

✚ **La ROM** : Une ROM ou mémoire morte, par opposition à la RAM, ne s'efface pas à la coupure du courant. Elle peut être lue mais pas (ou peu de fois) écrite. Elle conserve des programmes nécessaires au fonctionnement du matériel, surtout lors du démarrage (avant le chargement du système d'exploitation dans la RAM).

La mémoire centrale se mesure actuellement par milliers de mégaoctets.

On a :

□ bit : est l'abréviation de "binary digit" (chiffre binaire). C'est l'unité de base de l'information dans les systèmes informatiques et numériques. Un bit peut avoir deux valeurs possibles : 0 ou 1.

- ✓ 1 octet = 8 bits
- ✓ 1 Kiloctets (Ko) =  $2^{10}$  octets = 1024 octets
- ✓ 1 Mégaoctets (Mo) =  $2^{20}$  octets
- ✓ 1 Gigaoctets (Go) =  $2^{30}$  octets

c) **Les périphériques** : Les périphériques sont composés de périphériques d'entrée, de sortie et des périphériques d'entrée/sortie.

✚ **Périphériques d'entrée** : C'est l'ensemble de périphériques qui permet de transmettre des données à l'ordinateur : Le clavier, la souris, le microphone, la webcam, etc.).

✚ **Périphériques de sortie** : C'est l'ensemble de périphériques qui permet de recevoir des données de l'ordinateur et de les renvoyer vers l'extérieur : L'écran, l'imprimante, les haut-parleurs, etc.).

✚ **Périphériques d'entrée/sortie** : C'est l'ensemble de périphériques qui permet, à la fois, de transmettre et de recevoir des données. Ils sont, également, appelés périphériques de stockage ou mémoires externes (mémoires auxiliaires) : La disquette, le CD-ROM, la clé USB, le disque dure externe, etc.).

## I.5. Composition d'un ordinateur (Matériel & Logiciel)

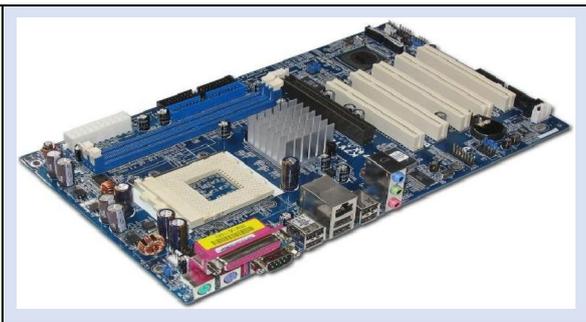
L'ordinateur est composé de deux parties essentielles : le matériel (hardware) et le logiciel (software).

### I.5.1. Partie matériel (Hardware)

Le matériel d'un ordinateur est la composante physique qui constitue la machine. Il inclut les composants électroniques, mécaniques et électriques qui permettent le fonctionnement de l'ordinateur. Voici quelques-unes des principales parties du matériel d'un ordinateur :

**- Carte Mère :**

Carte principale sur laquelle on retrouve tous les composants nécessaires au fonctionnement d'un ordinateur, dont le processeur, la mémoire centrale, les bus et les connecteurs d'extension destinés à recevoir des cartes d'extension (carte graphique externe, carte son, carte réseau, etc.).



**- Processeur :**

Processeur dont tous les éléments sont miniaturisés et rassemblés sur une puce en un seul ou en plusieurs circuits intégrés. Le processeur remplit les fonctions d'unité centrale dans un micro-ordinateur. Un ventilateur est utilisé pour réduire sa température souvent élevée et réduire les risques de surchauffe.



**- Mémoire Vive :**

Mémoire du travail du processeur, tout programme exécuté doit être chargé sur la mémoire vive, dans laquelle les instructions peuvent lire ou écrire des données.



**- Disque Dur :**

Support de mémoire de masse d'accès rapide qui est composé d'un ou de plusieurs disques magnétiques généralement solidaires d'une mécanique, et ayant une très grande capacité de stockage.



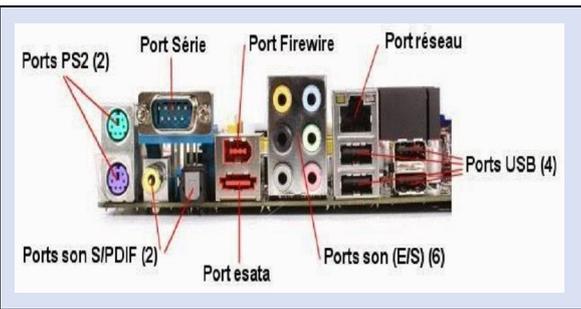
**- Carte graphique :**

Carte d'extension servant à stocker et à interpréter les images reçues par l'ordinateur et à mettre à jour continuellement l'écran du moniteur. Elle n'est pas obligatoire sur certains ordinateurs qui en possèdent une incorporée dans la carte mère. Mais pour de meilleures performances il est conseillé d'en installer une.



**- Connecteurs et Ports :**

Ces ports permettent de relier des périphériques externes (clavier, souris, etc.) au boîtier de l'unité centrale. Il y a plusieurs type de ports : Port USB, port parallèle, port série, etc.



**- Écran :**

Périphérique de sortie, il permet d'afficher les toute sorte de données. Il y a deux mode d'affichage : mode texte et mode graphique. Il y a des écran tactile permettant d'émettre des commandes via de simples touches de doigts, dans ce cas, l'écran représente aussi un périphérique d'entrée.



**- Clavier :**

Périphérique permettant d'entrer des données sous forme textuelle vers le CPU. Donc il représente un périphérique de sortie.



- Sourie :

<p>Permet de lancer des commandes à travers de simples cliques : clique gauche pour la sélection, double clique gauche pour exécuter un programme ou ouvrir un fichier, clique droit affiche le menu contextuel, etc.</p>	
---	--

- Imprimante :

<p>Périphérique de sortie permettant d'afficher des données (textuelles, images ou graphes) sur du papier.</p>	
--	--

La figure I.2 résume la partie hardware de l'ordinateur

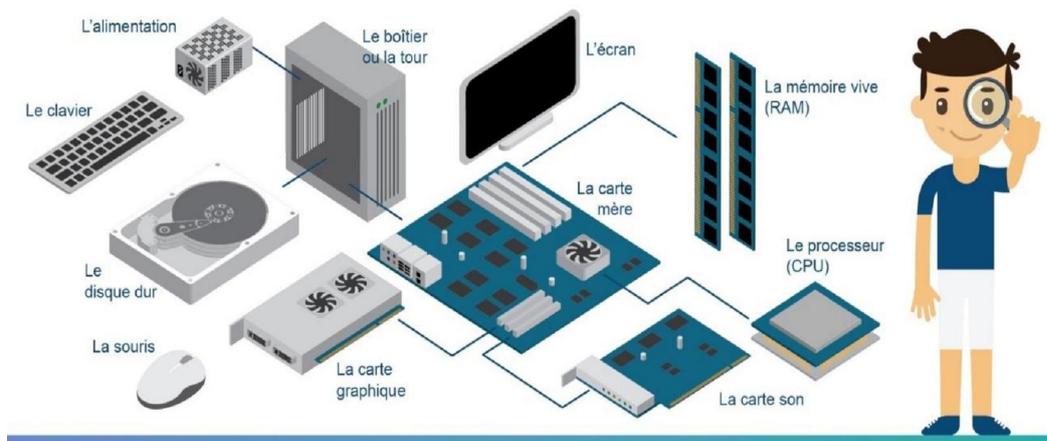


Figure I.2 : Partie hardware de l'ordinateur

**I.5.2. Partie logiciel (Software)**

Le logiciel d'un ordinateur est l'ensemble des programmes et des données qui permettent à l'ordinateur de fonctionner et d'effectuer des tâches spécifiques. On peut diviser cette partie en trois grandes catégories :

- **Systeme de base** : ce sont les système d'exploitation comme Windows, Linux, Mac OS, etc.

- **Langages de Programmation** : Ce sont de langages utilisés par les développeurs d'applications et de logiciels afin d'écrire des programmes. On trouve plusieurs langage de programmation, comme C/C++, PASCAL, Java, Fortran, Matlab, etc.
- **Logiciels d'application** : Ce sont des programmes exécutables spécialement écrits pour un système d'exploitation, et qui permettent de réaliser tout type de fonctions: traitement de texte, jeux et loisirs, retouche d'image, navigation internet, lecture de médias son, image, vidéo... (comme suite de MSOffice, PhotoShop, Auto CAD, Jeux, etc.).

## I.6. Les systèmes de codage des informations

Toute information (numérique, textuelle, image, vidéo, son,.. etc.) manipulée par un ordinateur, est représentée par des séquences de deux chiffres : **0** et **1**, ces derniers sont désignés par **BIT** (**B**inary **digi**T).

En informatique, il y a deux états possibles : il y a du courant électrique ou il n'y a pas de courant électrique, c'est la seule information qu'un ordinateur puisse comprendre. Cet état est déterminé par le bit (C'est la plus petite unité qui existe en informatique). Par conséquence :



### I.6.1. Les systèmes de numérotation

- **Définition** : C'est la représentation d'un entier naturel N en une base b. Ainsi, un système de numérotation se définit par deux éléments :
  1. La base du système (b);
  2. Les symboles du système (l'ensemble des chiffres et des lettres qui représentent la base).

En informatique, les systèmes les plus utilisés sont :

Système	Base	Symboles	Nombre de symboles du système
Décimal	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	10
Binaire	2	0, 1	2
Octal	8	0, 1, 2, 3, 4, 5, 6, 7	8
Hexadécimal	16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F	16

avec :  $(A)_{16}=(10)_{10}$ ,  $(B)_{16}=(11)_{10}$ ,  $(C)_{16}=(12)_{10}$ ,  $(D)_{16}=(13)_{10}$ ,  $(E)_{16}=(14)_{10}$ ,  $(F)_{16}=(15)_{10}$ .

❖ **Remarque :** le nombre de symboles d'un système = b.

### • Notation

Un nombre quelconque Nb exprimé dans une base b sera noté comme suit :

$$Nb = (a_{n-1} a_{n-2} \dots a_1 a_0)_b$$

avec :

$b$  : la base du système de numérotation

$a_i$  : symbole du système de numérotation,  $i=0, \dots, n-1$  et  $a_i < b$

### Exemples :

$$Nb_1 = (1995)_{10} \text{ avec } a_3=1, a_2 = 9, a_1 = 9, a_0 = 5$$

$$Nb_2 = (243)_8 \text{ avec } a_2 = 2, a_1 = 4, a_0 = 3$$

$$Nb_3 = (1011010)_2 \text{ avec } a_6 = 1, a_5 = 0, a_4 = 1, a_3=1, a_2 = 0, a_1 = 1, a_0 = 0$$

$$Nb_4 = (BAC)_8 \text{ avec } a_2 = B, a_1 = A, a_0 = C$$

$$Nb_5 = (248)_8 \text{ Cette notation est erronée, car le nombre contient } C \text{ un symbole supérieur ou égale à la base}$$

### ❖ Remarques :

- Quand la base n'est pas mentionnée, on considère qu'on est en base 10.
- Dans une base b, nous avons b chiffres : 0, 1, 2, ..., (b-1).

**I.6.2. Conversion d'un nombre d'un système à un autre**

**a) Conversion de la base 2, 8, 16 → base 10**

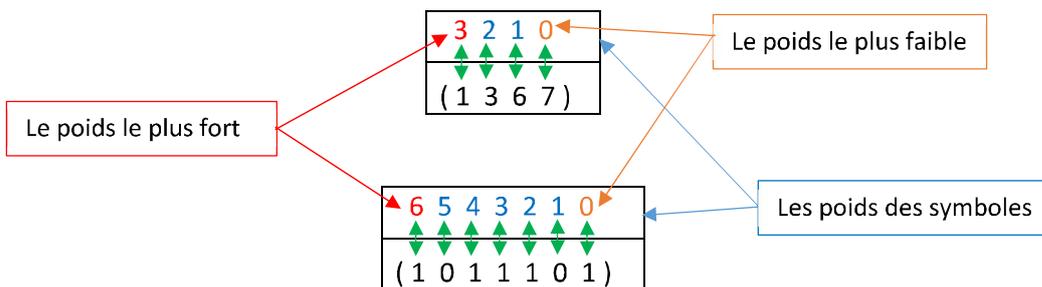
Pour convertir un nombre  $Nb = (a_{n-1} a_{n-2} \dots a_1 a_0)_b$  de la base b vers la base 10, on effectue le calcul suivant :

$$(Nb)_b = (a_{n-1} * b^{n-1} + a_{n-2} * b^{n-2} \dots + a_1 * b^1 + a_0 * b^0)_{10}$$

$$(Nb)_b = \sum_{i=0}^{n-1} a_i * b^i$$

Elle correspond à la somme des produits de chaque symbole du nombre par le poids correspondant.

**Le poids des symboles :** il s'obtient en numérotant les symboles à partir de la droite vers la gauche, en commençant du 0.



**Exemple 1 :**

$$(1367)_8 = (?)_{10}$$

$$(1367)_8 = (1 * 8^3 + 3 * 8^2 + 6 * 8^1 + 7 * 8^0)_{10}$$

$$(1367)_8 = (759)_{10}$$

**Exemple 2 :**

$$(1011101)_2 = (?)_{10}$$

$$(1011101)_2 = (1 * 2^6 + 0 * 2^5 + 1 * 2^4 + 1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0)_{10}$$

$$(1011101)_2 = (93)_{10}$$

**Exemple 3 :**

$$(2B3)_{16} = (?)_{10}$$

$$(2B3)_{16} = (2 * 16^2 + B * 16^1 + 3 * 16^0)_{10}$$

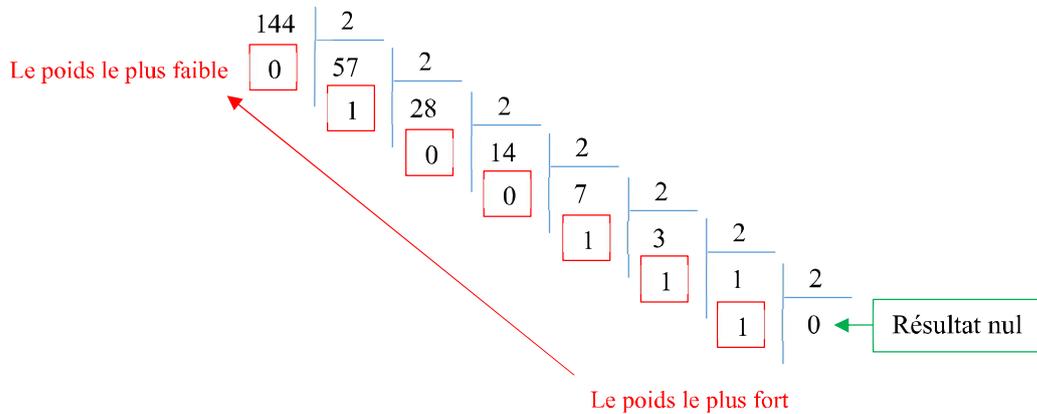
$$(2B3)_{16} = (691)_{10}$$

**b) Conversion de la base 10 → b (base 2, 8, 16,...etc)**

Soit Nb un nombre exprimé dans la base 10, pour trouver son équivalent en base b, on applique la méthode des divisions successives sur b, jusqu'à l'obtention d'un résultat nul. Puis, on récupère les restes des divisions dans le sens inverse, i.e. le dernier reste trouvé représentera le poids le plus fort et le premier reste trouvé sera le poids le plus faible.

**Exemple 1 :**

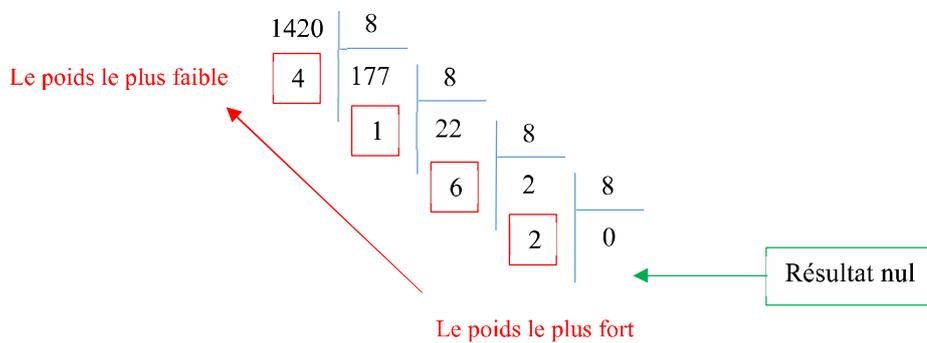
○  $(144)_{10} = (?)_2$



○  $(144)_{10} = (1110010)_2$

**Exemple 2 :**

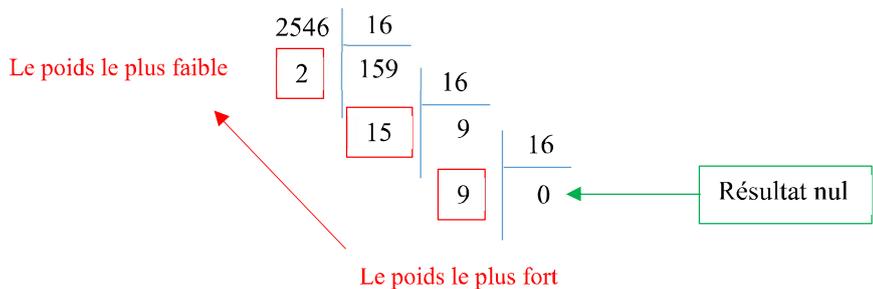
○  $(1420)_{10} = (?)_8$



○  $(1420)_{10} = (2614)_8$

**Exemple 3 :**

○  $(2546)_{10} = (?)_{16}$



○  $(2546)_{10} = (9F2)_{16}$

**c) Conversion de la base 8 → base 2**

Pour convertir un nombre Nb exprimé en base 8 vers la base 2, nous procédons comme suit:

$$8 = 2^3$$

Il faut donc utiliser **3 bits** pour exprimer un seul chiffre octal en binaire.

La représentation des chiffres de la base 8 vers le binaire est comme suit :

- $(7)_8 = (1 * 2^2 + 1 * 2^1 + 1 * 2^0)_2 = (111)_2$
- $(4)_8 = (1 * 2^2 + 0 * 2^1 + 0 * 2^0)_2 = (100)_2$
- $(3)_8 = (0 * 2^2 + 1 * 2^1 + 1 * 2^0)_2 = (011)_2$

Chiffre en octal	Chiffre équivalent en binaire ( $2^2$ $2^1$ $2^0$ )
0	0 0 0
1	0 0 1
2	0 1 0
3	0 1 1
4	1 0 0
5	1 0 1
6	1 1 0
7	1 1 1

**Exemple :**

○  $(743)_8 = (111 100 011)_2$

**d) Conversion de la base 16 → base 2**

Pour convertir un nombre Nb exprimé en base 16 vers la base 2, nous procédons comme suit:

$$16 = 2^4$$

Il faut donc utiliser **4 bits** pour exprimer un seul chiffre hexadécimal en binaire.

La représentation des chiffres de la base 16 vers le binaire est comme suit :

Chiffre en hexadécimal	Chiffre équivalent en binaire ( $2^3$ $2^2$ $2^1$ $2^0$ )
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1

Chiffre en hexadécimal	Chiffre équivalent en binaire ( $2^3$ $2^2$ $2^1$ $2^0$ )
8	1 0 0 0
9	1 0 0 1
A	1 0 1 0
B	1 0 1 1
C	1 1 0 0
D	1 1 0 1
E	1 1 1 0
F	1 1 1 1

- $(7)_{16} = (0 * 2^3 + 1 * 2^2 + 1 * 2^1 + 1 * 2^0)_2 = (0111)_2$
- $(B)_{16} = (1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 1 * 2^0)_2 = (1011)_2$
- $(3)_{16} = (0 * 2^3 + 0 * 2^2 + 1 * 2^1 + 1 * 2^0)_2 = (0011)_2$
- $(A)_{16} = (1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0)_2 = (1010)_2$

**Exemple :**

○  $(7B3A)_{16} = (0111\ 1011\ 0011\ 1010)_2$

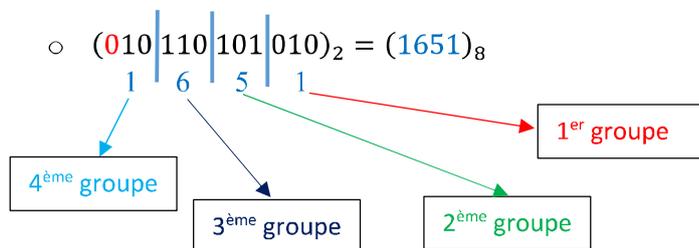
**e) Conversion de la base 2 → base 8**

Pour trouver l'équivalent d'un nombre binaire en octal, il suffit de former des **groupes de 3 bits** chacun ( Puisque  $8 = 2^3$  ), en commençant du poids le plus faible ( à partir de la droite), si le dernier groupe formé possède moins de 3 bits, il suffit de rajouter des 0, puis calculer l'équivalent en octal de chaque groupe.

**Exemple :**

○  $(10110101010)_2 = (?)_8$

○  $(010\ 110\ 101\ 010)_2 = (1651)_8$



**f) Conversion de la base 2 → base 16**

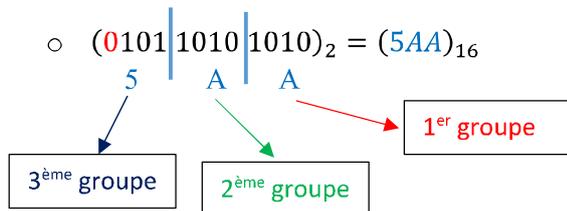
Pour trouver l'équivalent d'un nombre binaire en Hexadécimal, il suffit de former des **groupes de 4 bits** chacun ( Puisque  $16 = 2^4$  ), en commençant du poids le plus faible ( à partir

de la droite), si le dernier groupe formé possède moins de 4 bits, il suffit de rajouter des 0, puis calculer l'équivalent en Hexadécimal de chaque groupe.

**Exemple :**

○  $(10110101010)_2 = (?)_{16}$

○  $(0101|1010|1010)_2 = (5AA)_{16}$



**g) Conversion de la base 16 → base 8**

Pour convertir un nombre Nb exprimé en base 16 vers la base 8 ou vice versa, nous devons **passer par une base intermédiaire** tel que le **décimal ou le binaire**, mais le passage par le binaire est beaucoup plus simple.

**Exemple :**

○  $(C9F)_{16} = (1100\ 1001\ 1111)_2$

○  $(110|010|011|111)_2 = (6237)_8$

**I.6.3. Le code D.C.B. (Decimal Coded Binary – Décimal codé binaire)**

Le code DCB consiste à convertir chaque chiffre décimal en un nombre binaire sur 4 positions.

**Exemple :**

Le nombre décimal 378 est codé en DCB comme suit : 0101 0111 1000

Dans ce code, chaque chiffre décimal est remplacé par 4 chiffre binaires (comme dans le cas de hexadécimal).

**I.6.4. La codification Alphanumériques**

Les chiffres, lettres, signes de ponctuation, les symboles mathématiques, etc., sont représentées généralement en utilisant le code normalisé à 8 positions binaires. On utilise pour cela le code EBDIC (Extended Binary Coded Decimal International Code) ou le code ASCII (American Standard Code Information Interchange).

**Exemples :****Tableau I.2 :** Exemples de codification Alphanumériques

Lettre / Signe / Symbole	En EDCDIC	En ASCII
A (Majuscule)	11000001	10100001
D (miniscule)	10000010	11100100
=	01111110	01011101

**I.7. Les expressions arithmétiques, relationnelles et logiques****I.7.1. Expressions arithmétiques**

C'est une combinaison d'opérande(s) (valeur, constante, variable) et d'opération(s) arithmétiques. Le Tableau I.3 résume ces opérations.

**Tableau I.3 :** Les opérations arithmétiques

Opération	Symbole
Addition	+
Soustraction	-
Multiplication	*
Division réelle	/
Division entière	Div
Modulo (reste de la division entière)	Mod

} Applicables sur des entiers

❖ **Remarque :** La division entière donne comme résultat un quotient entier tandis que la division réelle donne un quotient réel. Le Mod, quant à lui, est le modulo, c.à.d. le reste de la division entière.

$10 \text{ Div } 4 = 2$  tandis que  $10 / 4 = 2.5 \rightarrow 10 \text{ Mod } 4 = 2$

$20 \text{ Div } 6 = 3$  tandis que  $20 / 6 = 3.33 \rightarrow 20 \text{ Mod } 6 = 2$

$6 \text{ Div } 8 = 0$  tandis que  $6 / 8 = 0.75 \rightarrow 6 \text{ Mod } 8 = 6$

**I.7.2. Expressions relationnelles**

C'est les expressions qui utilisent des opérateurs relationnels (ou de comparaison). Le Tableau I.4 résume ces opérateurs.

**Tableau I.4 :** Les opérateurs relationnels

Relation	Symbole
Egalité	=
Inférieur	<
Inférieur ou égal	<=

Supérieur	>
Supérieur ou égal	>=
Différent	<>

### I.7.3. Expressions logiques (booléennes)

C'est une combinaison de variables de type booléen (**True** : Vrai ou **False** : Faux) et d'opérateurs booléens (**Not** : Non, **And** : Et, **Or** : Ou). Le Tableau I.5 représente la table de vérité.

**Tableau I.5** : L'opérateur booléens AND

Opérande 1	Opérande 2	Opérande 1 <b>AND</b> Opérande 2
True	True	True
True	False	False
False	True	False
False	False	False

**Tableau I.6** : L'opérateur booléens OR

Opérande 1	Opérande 2	Opérande 1 <b>OR</b> Opérande 2
True	True	True
True	False	True
False	True	True
False	False	False

**Tableau I.7** : L'opérateur booléens NOT

Opérande	<b>NOT</b> Opérande
True	False
False	True

### I.7.4. Les fonctions

Le Tableau I.8 présente une liste non-exhaustive de fonctions standards (ou prédéfinies) applicables sur des entiers ou des réels.

**Tableau I.8** : Les fonctions

Fonction	Appel	Résultat retourné
Abs	Abs(x)	La valeur absolue d'un nombre x
Exp	Exp(x)	L'exponentiel d'un nombre x
Ln	Ln(x)	Le logarithme népérien d'un nombre x
Log10	Log10(x)	Le logarithme à base 10 d'un nombre x
Sqrt	Sqrt(x)	La racine carrée d'un nombre x
Sqr	Sqr(x)	Le carré d'un nombre x

Arctan	Arctan(x)	L'arc tangente d'un nombre x
Cos	Cos(x)	Le cosinus d'un nombre x
Sin	Sin(x)	Le sinus d'un nombre x
Round	Round(x)	La valeur arrondie d'un nombre x
Trunc	Trunc(x)	La partie entière d'un nombre x
Etc.	..	..

### 1.7.5. La priorité dans les expressions

La priorité des opérateurs dans les expressions arithmétiques, logiques et relationnelles est comme suit :

1. Les parenthèses ;
2. Les fonctions ;
3. Le moins unaire, le Not ;
4. \*, /, Div, Mod, And
5. +, -, Or
6. =, <>, <, >, <=, >=

### 1.7.6. Évaluation des expressions

L'évaluation d'une expression consiste à calculer, au fur et à mesure, les résultats des calculs jusqu'à obtenir un résultat finale. Cela se fait en plusieurs étapes :

- Écrire l'expression sous forme linéaire (Il faut noter qu'en algorithmique, les expressions s'écrivent sous forme linéaire:  $\frac{(x+z)}{(y*2)} \rightarrow (x + z)/(y * 2)$  ;
- Remplacer les identifiants (c'est à dire les noms) des variables et des constantes par leurs valeurs ;
- Évaluer (Calculer) étape par étape chacune des sous-expressions en commençant par les sous-expressions qui sont dans les parenthèses les plus internes.
- Indiquer à chaque calcul, le rang d'évaluation.

❖ **Remarque :** Si les opérateurs ont le même rang de priorité, l'évaluation se fait de gauche à droite.

#### Exemple :

Évaluer l'expression suivante en indiquant l'ordre d'évaluation :

$$E = (sqr(b) \text{ mod } a > c) \text{ or } (d / (a + 3) <> b) \text{ , avec } a = 2, b = 3, c = 1, d = 10$$

**1ère méthode :**

$$E = (\text{sqr}(3) \bmod 2 > 1) \text{or} (10 / \underline{(2 + 3)} <> 3)$$

(1)

$$E = (\underline{\text{sqr}(3)} \bmod 2 > 1) \text{or} (10 / 5 <> 3)$$

(2)

$$E = (\underline{9} \bmod 2 > 1) \text{or} (10 / 5 <> 3)$$

(3)

$$E = (\underline{1} > 1) \text{or} (10 / 5 <> 3)$$

(4)

$$E = \text{False} \text{or} (\underline{10 / 5} <> 3)$$

(5)

$$E = \text{False} \text{or} (\underline{2} <> 3)$$

(6)

$$E = \text{False} \text{or} \underline{\text{True}} = \text{True}$$

(7)

**2ème méthode :**  $E = (\text{sqr}(3) \bmod 2 > 1) \text{or} (10 / (2 + 3) <> 3) = \text{True}$

(2) (3) (4) (7) (5) (1) (6)

## I.8. Exercices corrigés

### Exercice N°01 : (Systèmes de numérotation)

Effectuer les conversions suivantes :

$$2022 = (?)_2 \quad ; \quad (1011001101)_2 = (?)_{10} \quad ; \quad (1011001101)_2 = (?)_8 = (?)_{16} \quad ;$$

$$(32103)_4 = (?)_2 \quad ; \quad (37163)_8 = (?)_2 \quad ; \quad (379)_{10} = (?)_{16} \quad ; \quad (3A2D)_{16} = (?)_{10} \quad ;$$

$$(4D5B)_{16} = (?)_8$$

### Corrigé de l'exercice N°01 : (Systèmes de numérotation)

Effectuer les conversions suivantes :

$$2022 = (11111100110)_2 \quad ;$$

$$(1011001101)_2 = (717)_{10} \quad ;$$

$$(1011001101)_2 = (1315)_8 = (2CD)_{16} \quad ;$$

$$(32103)_4 = (1110010011)_2 \quad ;$$

$$(37163)_8 = (011111001110011)_2 \text{ ou bien } (11111001110011)_2 \quad ;$$

$$(379)_{10} = (17B)_{16} \quad ;$$

$$(3A2D)_{16} = (14893)_{10} \quad ;$$

$$(4D5B)_{16} = (46533)_8$$

### Exercice N°02 : (Expressions arithmétiques en Algorithm/Pascal)

Réécrire les expressions mathématiques en Algorithm/Pascal

Expressions mathématiques	PASCAL
$b^2 - 4ac$	
$\frac{-b - \sqrt{d}}{2a}$	
$e^{3a} +  b $	
$4a < \frac{b}{c}$ ET $(5c \leq 7)$ OU $(a \neq b)$	

Utiliser le tableau suivant :

Expression	PASCAL
$2a$	$2 * a$
$\frac{a}{b}$	$a/b$
$a^2$	$sqr(a)$
$\sqrt{a}$	$sqrt(a)$
$ a $	$abs(a)$
$\ln(a)$	$\ln(a)$
$\log(a)$	$\ln(a) / \ln(10)$
$e^a$	$\exp(a)$
$x^n$	$\exp(n * \ln(x))$



**Expression 2 :**  $a/b + ((d * c + 3)/5 * a) + 2 * c$

$$4/2 + ((\underline{(3 * 4 + 3)}/5 * 4) + 2 * 4$$

(1)

$$4/2 + ((\underline{(12 + 3)}/5 * 4) + 2 * 4$$

(2)

$$4/2 + (\underline{(15/5 * 4)} + 2 * 4$$

$$4/2 + (\underline{(3 * 4)} + 2 * 4$$

(3)

$$\underline{4/2 + 12} + 2 * 4$$

(4)

$$2 + 12 + \underline{2 * 4}$$

(5)

$$\underline{2 + 12} + 8$$

(6)

$$\underline{14 + 8} = 22$$

(7)

(8)

On commence par remplacer les variables par leurs valeurs

**Expression 3 :**  $(a < b) \text{ OR NOT } (c \leq d) \text{ AND } (b > a)$

$$\underline{(1 < 2)} \text{ OR NOT } (4 \leq 6) \text{ AND } (2 > 1)$$

(1)

$$\text{True OR NOT } (\underline{4 \leq 6}) \text{ AND } (2 > 1)$$

(2)

$$\text{True OR NOT True AND } (\underline{2 > 1})$$

(3)

$$\text{True OR NOT True AND True}$$

(4)

$$\text{True OR } \underline{\text{False AND True}}$$

(5)

$$\underline{\text{True OR False}} = \text{True}$$

(6)

## I.9. Exercices supplémentaires

### Exercice 01 :

1. Donner l'ordre de priorité des opérateurs arithmétiques et logiques dans les expressions suivantes :

$$E1 = (2 * x + 3 * y) / x + 6 * y - (5 \text{ MOD } y * 2 * x)$$

$$E2 = (3 \text{ MOD } a * 2 <= b) \text{ OR } \text{NOT}(b >= 8) \text{ NOT}(a = b)$$

2. Traduire les expressions suivantes en langage Pascal (on rappelle que  $a^b = e^{b \ln(a)}$ )

$$E1 = \frac{-\sqrt{a} + e^{3b} + \sqrt{a^2 + ab}}{2a + |b|}, \quad E2 = \frac{5^2 + a^{3b} + \sqrt{e^a + c}}{3\sqrt{b}}$$

3. Effectuer les conversions suivantes (les réponses doivent être justifiées):

$$(127)_{10} = (?)_2 = (?)_8, \quad (\text{B7C})_{16} = (?)_{10}$$

4. a) Classer ce qui suit dans les trois catégories suivantes : 1. Systèmes d'exploitation 2.

Langages de programmation 3. Logiciels spécialisés.

Pascal, Microsoft office, Lecteur Media, Windows vista, Google chrome, Mac OS, Adobe, C++, Matlab, Unix, WinRAR.

- b) Citer 3 noms de systèmes d'exploitation pour PC

### Exercice 02 :

1. a) Classer ce qui suit dans les trois catégories suivantes : 1. Systèmes d'exploitation 2.

Langages de programmation 3. Logiciels spécialisés.

Pascal, Microsoft word, Lecteur Media, Windows vista, Google chrome, Mac OS, C++, Matlab, Linux.

- b) Quelle est la signification des acronymes suivants : CPU, UAL, RAM, ROM

2. Evaluer les expressions suivantes en respectant l'ordre de priorité des opérateurs :

$$E1 = (a/c) - ((d/2 * a + 4)/4 - c)/2b ; \text{ avec } a = 6, b = 1, c = 1, d = 4$$

$$E2 = (1 > c) \text{ AND } \text{NOT}(7 \text{ MOD } a * 2 <= b) \text{ OR } (b >= 8) \text{ AND } (a = b) ; \text{ avec}$$

$$a = 6, b = 3, c = 1$$

3. Traduire les expressions suivantes en langage Pascal

$$E1 = \frac{-\sqrt{a} + e^{3b} + \sqrt{a^2 + ab}}{2a + |b|}, \quad E2 = \frac{a^2 + \sqrt{e^a + c}}{3\sqrt{b}}$$

4. Effectuer les conversions suivantes :

$$(120)_{10} = (?)_2 = (?)_8, \quad (10110101111)_2 = (?)_{16}$$

**Exercice 03 :**

1. Citer deux périphériques de chacune des trois catégories suivantes :

Entrée	Sortie	Entrée/Sortie

2. Effectuer les conversions suivantes :

$$(90)_{10} = (?)_2 = (?)_{16}, (10110101011)_2 = (?)_8$$

3. Traduire les expressions suivantes en langage Pascal (on rappelle que  $a^b = e^{b \ln(a)}$ )

$$E1 = \frac{-\sqrt{2a} + e^{3b} + \sqrt{|2a| + b}}{8 + |b|}, \quad E2 = \frac{5^2 + a^{3b} + \sqrt{e^a + c}}{\sqrt{b}}$$

4. Evaluer l'expression suivante en respectant l'ordre de priorité des opérateurs :

$$E1 = (a/c) - ((d/2 * a + 4)/4 - c)/2b ; \text{ avec } a = 6, b = 1, c = 1, d = 4$$

5. Donner l'expression arithmétique correspondante à l'expression suivante écrite en Pascal :

$$\exp(\text{sqrt}(x))/(2 * y - 1) + \text{abs}(x) - 1/(\text{sqr}(x) + 3)$$

**Exercice 04 :**

1. Quelles sont les trois structures itératives de base ?

2. Citer deux périphériques d'entrée et deux périphériques de sortie.

3. Quels sont les identificateurs valides et invalides parmi les exemples suivants (les identificateurs non valides doivent être justifiés) : `_EX01` ; `EX_01` ; `EX-01` ; `EX01_` .

4. Evaluer l'expression suivante tout en montrant l'ordre des opérations.

$$(a > 9 \text{ DIV } 4) \text{ AND } (a < > b) \text{ OR NOT } (c = b). \text{ avec } a = 4, b = 8, c = 8.$$

5. Donner l'expression arithmétique correspondante à l'expression suivante écrite en Pascal :

$$\text{sqrt}(\text{abs}(2*x + 1 + x*y)) / (\text{sqr}(x) - 2*x*y) + \text{sqrt}(4*x + 5*x).$$

6. Effectuer les conversions suivantes :  $(1EA)_{16} = (?)_8$  ,  $(240)_{10} = (?)_2$  .