

INFORMATIQUE DE BASE

1^{re} année ENCG

Prof. Mohamed BOUDCHICHE
 Email: m.boudchiche@encgasa.ma
 Web: http://mohavic.com

Année universitaire 2023-2024

1

OBJECTIFS DU COURS

- Maîtriser les concepts informatiques de base.
- Clarifier le vocabulaire informatique.
- Etablir une méthodologie de programmation informatique.
- Apprendre un langage informatique.

Prof. Mohamed BOUDCHICHE 02

2

01 Historique de l'informatique

Prof. Mohamed BOUDCHICHE

3

L'informatique ?

```

  graph TD
    A[INFORMATIQUE] --> B[INFORMATION]
    A --> C[AUTOMATIQUE]
    B --> D[Traitement automatique de l'information]
    C --> E[Machine automatique]
    D --> F[ORDINATEUR]
    E --> F
  
```

" Science de l'information " Science qui traite l'automatisation des actions

4 Prof. Mohamed BOUDCHICHE

4

Historique de l'informatique

- ▶ Les premiers êtres humains devaient d'abord imaginer une façon de compter et de calculer.
- ▶ Il fallait, après tout, partager la nourriture parmi la tribu et compter les mois avant l'hiver.

5 Prof. Mohamed BOUDCHICHE

5

Les Mathématiques

À travers le temps, différentes méthodes ont été inventées pour calculer.

6 Prof. Mohamed BOUDCHICHE

6

La mécanisation du calcul

A travers le temps différentes méthodes ont été inventées pour calculer :

Boulier chinois, 700

Prof. Mohamed BOUDCHICHE

7

La mécanisation du calcul

1614 L'écosais John Neper (Napier, 1550-1617) invente les logarithmes

- simplifier les calculs trigonométriques en astronomie
- consiste à remplacer une multiplication par une addition + lecture d'une valeur dans une table

Prof. Mohamed BOUDCHICHE

8

La mécanisation du calcul

Exemple : on veut calculer :

$2^5 \times 2^4$

❖ Première méthode : poser le calcul

$$\begin{array}{r} 32 \\ \times 16 \\ \hline 192 \\ 320 \\ \hline 512 \end{array}$$

❖ Deuxième méthode (Neper) : utiliser une table

$2^5 \times 2^4 = 2^{(5+4)} = 2^9$

N	1	2	3	4	5	6	7	8	9
2 ^N	2	4	8	16	32	64	128	256	512

Prof. Mohamed BOUDCHICHE

9

La mécanisation du calcul

Vers 1642, Blaise Pascal développe une machine à calculer, la **pascaline**.

Ceci allait devenir l'ancêtre lointain de l'ordinateur.

- utilisée pour les calculs scientifiques jusqu'en 1970

Prof. Mohamed BOUDCHICHE

10

La mécanisation du calcul

La calculatrice de Leibniz, 1672, 4 opérations et extrait les racines carrées.

Prof. Mohamed BOUDCHICHE

11

La mécanisation du calcul

1890 Hermann Hollerith construit un calculateur statistique électromécanique

- plus performant que les calculateurs mécaniques
- utilisation de cartes perforées
- utilisé pour le recensement américain de 1890
- fonde la Tabulating Machine Company => IBM (International Business Machines)

Prof. Mohamed BOUDCHICHE

12

Historique de l'informatique

- ▶ L'informatique a une longue et riche histoire. Les premières machines à calculer ont été inventées au 17ème siècle.
- ▶ Au 20ème siècle, les ordinateurs ont commencé à être utilisés à des fins commerciales et scientifiques.

13 Prof. Mohamed BOUDCHICHE

13

02 Les différentes générations d'ordinateurs

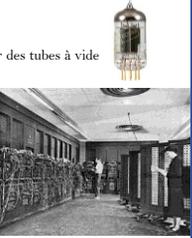
Prof. Mohamed BOUDCHICHE

14

Première génération (1945-1955)

Caractéristiques

- ▶ Les ordinateurs de cette génération étaient basés sur des tubes à vide et des relais. Ils étaient très grands et coûteux.
- ▶ **Exemple** : l'ENIAC (Electronic Numerical Integrator and Computer).
- ▶ **Limitations** : Encombrants, consomment beaucoup d'énergie, généralement peu fiables.



15 Prof. Mohamed BOUDCHICHE

15

Deuxième génération (1955-1965)

Deuxième génération : L'avènement des transistors

- ▶ Les transistors ont remplacé les tubes à vide, rendant les ordinateurs plus petits et plus fiables.
- ▶ **Exemple** : IBM 1401
- ▶ **Avancées** : Réduction de la taille, amélioration de la fiabilité, consommation d'énergie moindre.



16 Prof. Mohamed BOUDCHICHE

16

Les différentes générations d'ordinateurs

Troisième génération : Les circuits intégrés

- ▶ Les circuits intégrés ont permis de regrouper de multiples transistors sur une seule puce de silicium.
- ▶ **Exemple** : IBM System/360.
- ▶ **Avantages** : Réduction significative de la taille, performances accrues.



17 Prof. Mohamed BOUDCHICHE

17

Les différentes générations d'ordinateurs

Quatrième génération : Les microprocesseurs

- ▶ L'invention du microprocesseur a conduit à l'ère des ordinateurs personnels.
- ▶ **Exemple** : Intel 4004.
- ▶ **Révolution** : Miniaturisation, accessibilité, et augmentation des performances.



18 Prof. Mohamed BOUDCHICHE

18

Les différentes générations d'ordinateurs

Cinquième génération : L'intelligence artificielle et les ordinateurs quantiques

- ▶ L'IA et les ordinateurs quantiques ouvrent de nouvelles possibilités pour la résolution de problèmes complexes.
- ▶ **Exemple :** Ordinateurs quantiques d'IBM et de Google.
- ▶ **Enjeux :** Capacités de calcul exponentielles, applications en intelligence artificielle.



19 Prof. Mohamed BOUDCHICHE

19

Le Futur de l'Informatique

Intelligence Artificielle (IA)

- ▶ L'IA est une discipline informatique qui vise à créer des agents intelligents, c'est-à-dire des systèmes capables de simuler l'intelligence humaine. L'IA est en plein essor et elle a le potentiel de transformer de nombreux domaines de notre vie, notamment la santé, l'éducation, la finance et l'industrie.



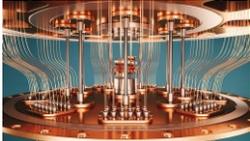
20 Prof. Mohamed BOUDCHICHE

20

Le Futur de l'Informatique

Informatique Quantique

- ▶ L'informatique quantique est une nouvelle discipline informatique qui exploite les propriétés quantiques de la matière pour résoudre des problèmes qui sont impossibles à résoudre avec les ordinateurs classiques.
- ▶ Les ordinateurs quantiques sont encore en développement, mais ils ont le potentiel de révolutionner de nombreux domaines, notamment la recherche, le développement de médicaments et la simulation.



21 Prof. Mohamed BOUDCHICHE

21

Le Futur de l'Informatique

Internet des Objets (IoT)

- ▶ L'IoT est un réseau d'objets physiques connectés à Internet. Ces objets sont capables de collecter et d'échanger des données, ce qui permet de créer de nouveaux services et applications.
- ▶ L'IoT est en plein essor et il est déjà présent dans de nombreux domaines, notamment la maison intelligente, la santé connectée et la ville intelligente.



22 Prof. Mohamed BOUDCHICHE

22

Le Futur de l'Informatique

Sécurité Informatique

- ▶ La sécurité informatique est un domaine en constante évolution qui vise à protéger les systèmes informatiques contre les cyberattaques.
- ▶ Les cyberattaques sont de plus en plus sophistiquées et elles représentent une menace croissante pour les entreprises, les gouvernements et les particuliers.



23 Prof. Mohamed BOUDCHICHE

23

03 Architecture des ordinateurs

Prof. Mohamed BOUDCHICHE

24

Architecture des ordinateurs

- Processeur (CPU : *Central processing unit*)
 - Unité arithmétique et logique (UAL) : effectue les opérations arithmétiques et logiques
 - Registres : mémoires à accès rapide utilisées par le processeur
 - Unité de contrôle (UC) : interprète les instructions du programme et les transmet à l'UAL

25 Prof. Mohamed BOUDCHICHE

25

Architecture des ordinateurs

- Mémoire
 - Mémoire vive (RAM : *Random Access Memory*) : stocke les données et les instructions du programme en cours d'exécution
 - Mémoire morte (ROM : *Read-only memory*) : stocke les instructions du programme de démarrage

26 Prof. Mohamed BOUDCHICHE

26

Architecture des ordinateurs

- Périphériques d'entrée/sortie
 - Fonction : Les périphériques permettent à l'ordinateur d'interagir avec l'utilisateur et le monde extérieur.
 - Périphériques d'entrée : clavier, souris, scanner, webcam
 - Périphériques de sortie : écran, imprimante, haut-parleur, projecteur
 - Périphériques d'entrée/sortie : disque dur, lecteur CD/DVD, clé USB
 - Rôle crucial : Facilitent la communication et l'interaction avec l'ordinateur.

27 Prof. Mohamed BOUDCHICHE

27

Évolution des Composants Informatiques

- Carte mère
 - Rôle central dans l'architecture des ordinateurs.
 - Connecte CPU, RAM, périphériques, et d'autres composants.
- Stockage
 - Disques durs, disques SSD, et stockage cloud.
 - Capacités de stockage en constante augmentation.

28 Prof. Mohamed BOUDCHICHE

28

Évolution des Composants Informatiques

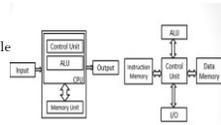
- Carte graphique
 - Gestion des affichages visuels et des graphiques.
 - Cruciale pour les jeux vidéo et le rendu 3D.
- Réseaux
 - Connectivité Internet et intranet.
 - Évolution des protocoles de communication.

29 Prof. Mohamed BOUDCHICHE

29

Architecture de CPU

- Von Neumann vs Harvard
 - Von Neumann : CPU et mémoire partagent le même bus de données.
 - Harvard : CPU et mémoire ont des bus de données distincts.
- Pipeline de traitement
 - Division des tâches en étapes pour améliorer les performances.
- Cœurs multiples
 - Processeurs avec plusieurs cœurs pour le traitement parallèle.



30 Prof. Mohamed BOUDCHICHE

30

Mémoire et Hiérarchie de Stockage

- Cache
 - Mémoire cache de niveau 1 (L1) et de niveau 2 (L2).
 - Stocke des données fréquemment utilisées pour un accès rapide.
- Disque dur vs. SSD
 - Disques durs : Stockage magnétique, grande capacité.
 - SSD (Solid State Drive) : Stockage à base de puces mémoire, vitesse élevée.
- Virtual Memory
 - Utilisation de la RAM comme extension du stockage de données.

31 Prof. Mohamed BOUDCHICHE

31

04 Systèmes d'exploitation

Prof. Mohamed BOUDCHICHE

32

Systèmes d'exploitation

- Définition :
 - ▶ Un système d'exploitation (SE) est un logiciel qui gère les ressources matérielles et facilite l'exécution d'applications.
- Exemples de systèmes d'exploitation :
 - **Windows** : Développé par Microsoft, largement utilisé dans les ordinateurs personnels.
 - **macOS** : Le système d'exploitation d'Apple pour les ordinateurs Mac.
 - **Linux** : Un système d'exploitation open source utilisé dans de nombreux environnements.

33 Prof. Mohamed BOUDCHICHE

33

Systèmes d'exploitation

- Fonctions essentielles :
 - **Gestion des fichiers et des répertoires** : Organise et permet l'accès aux données.
 - **Gestion des processus** : Contrôle l'exécution des programmes.
 - **Interface utilisateur** : Fournit un environnement convivial pour les utilisateurs.
 - **Gestion des périphériques** : Facilite l'interaction avec les composants matériels.

34 Prof. Mohamed BOUDCHICHE

34

Systèmes d'Exploitation Modernes

- Unix/Linux
 - Stabilité et sécurité, largement utilisés dans les serveurs.
- Windows 10/11
 - Interface utilisateur conviviale, utilisé dans les ordinateurs personnels.
- macOS Big Sur/Catalina
 - Système d'exploitation exclusif pour les ordinateurs Apple.
- Systèmes mobiles
 - Android (Google) et iOS (Apple) pour smartphones et tablettes.

35 Prof. Mohamed BOUDCHICHE

35

05 Codage binaire

Prof. Mohamed BOUDCHICHE

36

Introduction

- Le codage binaire est un système de représentation de l'information utilisant deux symboles, 0 et 1.
- Il est la base de l'informatique et des technologies numériques.

37 Prof. Mohamed BOUDCHICHE

37

Définitions

- Bit** : unité de mesure de l'information, pouvant prendre deux valeurs : 0 ou 1.
- Octet** : groupe de 8 bits, utilisé pour représenter un caractère.

38 Prof. Mohamed BOUDCHICHE

38

Système de numération binaire

- Le système de numération binaire est un système de numération positionnelle à base 2.
- Chaque position d'un nombre binaire a une valeur qui est un multiple de 2.

39 Prof. Mohamed BOUDCHICHE

39

Conversion décimal vers binaire

- Pour convertir un nombre décimal en binaire, on procède par division successive par 2.
- Le reste de chaque division est un bit du nombre binaire.

40 Prof. Mohamed BOUDCHICHE

40

Exemple de conversion décimal vers binaire

Nombre décimal	Nombre binaire
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010

41 Prof. Mohamed BOUDCHICHE

41

Conversion binaire vers décimal

- Pour convertir un nombre binaire en décimal, on procède par addition des valeurs des bits, en multipliant chaque bit par la puissance de 2 correspondant à sa position.

42 Prof. Mohamed BOUDCHICHE

42

Exemple de conversion binaire vers décimal

Nombre binaire	Nombre décimal
1	1
10	2
11	3
100	4
101	5
110	6
111	7
1000	8
1001	9
1010	10

43 Prof. Mohamed BOUDCHICHE

43

Applications du codage binaire

Le codage binaire est utilisé dans de nombreuses applications informatiques, notamment :

- La représentation des nombres
- La représentation des caractères
- La représentation des images
- La représentation des sons

44 Prof. Mohamed BOUDCHICHE

44

Exemples de codage binaire

Présentation d'exemples de codage binaire pour différents types d'informations, tels que :

- Les nombres : 11001010 = 1234
- Les caractères : 01000001 = A
- Les images : 00000000 00000000 00000000 ... 11111111 11111111
- Les sons : 00000000 00000000 00000000 ... 11111111 11111111

45 Prof. Mohamed BOUDCHICHE

45

Conclusion

- Le codage binaire est un concept fondamental de l'informatique.
- Il est important de comprendre ce concept pour pouvoir utiliser les technologies numériques.

46 Prof. Mohamed BOUDCHICHE

46

06 Codage d'information

47 Prof. Mohamed BOUDCHICHE

47

Introduction

- Le codage de l'information est le processus de transformation d'une information d'une forme à une autre.
- Il permet de représenter l'information de manière à pouvoir la manipuler, la stocker ou la transmettre.
- Le codage de l'information est utilisé dans de nombreux domaines, tels que l'informatique, la communication et la compression de données.

48 Prof. Mohamed BOUDCHICHE

48

Définitions

- **Information** : ensemble de données qui ont un sens pour un être humain ou un système.
- **Codage** : processus de transformation d'une information d'une forme à une autre.
- **Alphabet** : ensemble de symboles utilisés pour représenter l'information.

49 Prof. Mohamed BOUDCHICHE

49

Types de codage

- **Codage binaire** : utilisation de deux symboles, 0 et 1, pour représenter l'information.
- **Codage numérique** : utilisation d'un nombre fini de symboles pour représenter l'information.
- **Codage analogique** : utilisation d'un signal continu pour représenter l'information.

50 Prof. Mohamed BOUDCHICHE

50

Exemples de codage

- **Codage ASCII** : codage binaire utilisé pour représenter les caractères alphanumériques.
- **Codage JPEG** : codage numérique utilisé pour compresser les images.
- **Codage MP3** : codage numérique utilisé pour compresser les sons.

51 Prof. Mohamed BOUDCHICHE

51

Les bases de numération

- Habituellement, on utilise la base 10 pour représenter les nombres, c'est à dire que l'on écrit les chiffres à l'aide de 10 symboles distincts : (0, 1, 2, 3, 4, 5, 6, 7, 8, 9).

52 Prof. Mohamed BOUDCHICHE

52

Les bases de numération

- Soit b un entier avec $b > 1$, soit X un nombre. Soit a_i la suite des chiffres utilisés pour écrire le nombre X . Alors, dans la base b , le nombre X est donné par :

$$X = (a_{n-1} \dots a_1 a_0)_b$$
- ✓ a_0 est le chiffre de poids faible, et a_{n-1} est le chiffre de poids fort.
- ✓ La notation $()_b$ indique que le nombre est écrit en base b .

53 Prof. Mohamed BOUDCHICHE

53

Les bases de numération

- **Base Décimale, $b = 10$** : $a_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
Ce système de numération est utilisé dans notre vie courante
- **Base binaire, $b = 2$** : $a_i \in \{0, 1\}$
- **Base octale, $b = 8$** : $a_i \in \{0, 1, 2, 3, 4, 5, 6, 7\}$
- **Base hexadécimale, $b = 16$** :
 $a_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$

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

54 Prof. Mohamed BOUDCHICHE

54

Codage des entiers naturels

En base 10, le nombre 7849 (par exemple) s'écrit comme :

$$7849 = 7 * 10^3 + 8 * 10^2 + 4 * 10^1 + 9 * 10^0$$

En général, la représentation décimale de l'entier naturel X représenté par la suite $(a_{n-1} \dots a_1 a_0)$ en base b , est donné par :

$$X = a_{n-1} * b^{n-1} + a_{n-2} * b^{n-2} + \dots + a_1 * b^1 + a_0 * b^0 = \sum_{i=0}^{n-1} a_i * b^i$$

55 Prof. Mohamed BOUDCHICHE

55

Codage des entiers naturels

Donc en **base décimale** $b = 10$:

$$X = a_{n-1} * 10^{n-1} + a_{n-2} * 10^{n-2} + \dots + a_1 * 10^1 + a_0 * 10^0$$

En **base binaire** $b = 2$:

$$X = a_{n-1} * 2^{n-1} + a_{n-2} * 2^{n-2} + \dots + a_1 * 2^1 + a_0 * 2^0$$

En **base octale** $b = 8$:

$$X = a_{n-1} * 8^{n-1} + a_{n-2} * 8^{n-2} + \dots + a_1 * 8^1 + a_0 * 8^0$$

En **base hexadécimale** $b = 16$:

$$X = a_{n-1} * 16^{n-1} + a_{n-2} * 16^{n-2} + \dots + a_1 * 16^1 + a_0 * 16^0$$

56 Prof. Mohamed BOUDCHICHE

56

Codage des entiers naturels

Exemples :

Soit $X = (10011110)_2$, alors X représente le nombre :

$$X = 1 * 2^7 + 0 * 2^6 + 0 * 2^5 + 1 * 2^4 + 1 * 2^3 + 1 * 2^2 + 1 * 2^1 + 0 * 2^0 = (158)_{10}$$

Soit $X = (236)_8$, alors X représente le nombre :

$$X = 2 * 8^2 + 3 * 8^1 + 6 * 8^0 = (158)_{10}$$

Soit $X = (9E)_{16}$, alors X représente le nombre :

$$X = 9 * 16^1 + 14 * 16^0 = (158)_{10}$$

On déduit alors que :

$$X = (10011110)_2 = (236)_8 = (9E)_{16} = (158)_{10}$$

57 Prof. Mohamed BOUDCHICHE

57

Codage des entiers naturels

- ❖ Avec n chiffres on peut coder b^n entiers en base b .
- ❖ Dans un ordinateur on code les entiers sur 8 bits (1 octet), 16 bits (2 octets); 32 bits (4 octets) ou sur 64 bits (8 octets).
- ❖ Les bits sont rangés dans des cellules correspondant à leurs poids, on complète à gauche par des 0. Par exemple, le codage de 10 sur 1 octet sera : **0 0 0 0 0 1 0 1 0**

58 Prof. Mohamed BOUDCHICHE

58

Codage des entiers naturels

- ❖ Sur 8 bits, on peut coder $2^8 = 256$ valeurs différentes (de 0 à 2^8-1).
- ❖ Sur 16 bits, on peut coder $2^{16} = 65\,536$ valeurs différentes (de 0 à $2^{16}-1$).
- ❖ Sur 32 bits, on peut coder $2^{32} = 4\,294\,967\,296$
- ❖ Sur 64 bits, on peut coder $2^{64} = 18\,446\,744\,073\,709\,551\,616$
- ❖ **Problème :** limitation du nombre d'entiers représentables sur ordinateur.

59 Prof. Mohamed BOUDCHICHE

59

07 Changement de bases

60 Prof. Mohamed BOUDCHICHE

60

Passage d'une base quelconque à la base 10

Pour convertir un nombre représenté en une base quelconque à la base 10, il suffit d'effectuer les opérations en décimal.

Exemple :

$$\begin{aligned}(AE)_{16} &= 10 \cdot 16^1 + 14 \cdot 16^0 \\ &= 160 + 14 \\ &= (174)_{10}\end{aligned}$$

61 Prof. Mohamed BOUDCHICHE

61

Passage de la base 10 à une base quelconque :

On procède par division successives. On divise le nombre entier par la base, puis le quotient obtenu par la base, et ainsi de suite jusqu'à obtention d'un quotient nul. La suite des restes obtenus correspond aux chiffres constituant la base visée,

$$a_0 a_1 \dots a_{n-2} a_{n-1}$$

62 Prof. Mohamed BOUDCHICHE

62

Passage de la base 10 à une base quelconque :

Exemple :

Soit à convertir $(35)_{10}$ vers la base 2. Alors on procède comme suit :

$$\begin{aligned}35 &= 17 \cdot 2 + 1 \rightarrow a_0 = 1 \\ 17 &= 8 \cdot 2 + 1 \rightarrow a_1 = 1 \\ 8 &= 4 \cdot 2 + 0 \rightarrow a_2 = 0 \\ 4 &= 2 \cdot 2 + 0 \rightarrow a_3 = 0 \\ 2 &= 1 \cdot 2 + 0 \rightarrow a_4 = 0 \\ 1 &= 0 \cdot 2 + 1 \rightarrow a_5 = 1\end{aligned}$$

Donc $(35)_{10} = (100011)_2$.

63 Prof. Mohamed BOUDCHICHE

63

Conversion du binaire en octale :

Soit X un entier écrit en base 2. La représentation de X en base 8 peut se faire de la façon suivante :

- 1 On assemble les chiffres par groupe de 3 bits, en partant de la droite.
- 2 On complète à gauche par des 0, le groupe dont le nombre de bits est inférieur à 3 (s'il existe).
- 3 On convertit chaque groupe de bits en chiffres (convertir en base 8).

64 Prof. Mohamed BOUDCHICHE

64

Conversion du binaire en octale :

Exemple :

11010101001

↓ 1 : assembler par groupe de 3.

11 010 101 001

↓ 2 : compléter à gauche par des 0.

011 010 101 001

↓ 3 : convertir en chiffres dans la base 8.

3 2 5 1

Donc $(11010101001)_2 = (3251)_8$.

65 Prof. Mohamed BOUDCHICHE

65

Conversion du binaire en hexadécimale :

Soit X un entier écrit en base 2. La représentation de X en base 16 peut se faire de la façon suivante :

- 1 On assemble les chiffres par groupe de 4 bits, en partant de la droite.
- 2 On complète à gauche par des 0, le groupe dont le nombre de bits est inférieur à 4 (s'il existe).
- 3 On convertit chaque groupe de bits en chiffres/lettres (convertir en base 16).

66 Prof. Mohamed BOUDCHICHE

66

Conversion du binaire en hexadécimale :

► **Exemple :**

11010101001

↓ 1 : assembler par groupe de 4.

110 1010 1001

↓ 2 : compléter à gauche par des 0.

0110 1010 1001

↓ 3 : convertir en chiffres dans la base 16.

6 A 9

Donc $(11010101001)_2 = (6A9)_{16}$.

67 Prof. Mohamed BOUDCHICHE

67

Exercices

- 1 Convertir en base binaire les nombres : $(200)_{10}$, $(4713)_8$, $(A8B6)_{16}$
- 2 Convertir en base octale les nombres : $(987)_{10}$, $(11101001)_2$, $(F3B2C)_{16}$
- 3 Convertir en base Hexadécimale les nombres: $(9563)_{10}$, $(110101001001)_2$, $(4367)_8$.

68 Prof. Mohamed BOUDCHICHE

68

08 Opérations usuelles en binaire

Prof. Mohamed BOUDCHICHE

69

Opérations usuelles en binaire

► Les opérations sur les entiers s'appuient sur les deux opérations de bases : l'addition et la multiplication.

► **Table d'addition binaire :**

$0 + 0 = 0$
 $0 + 1 = 1$
 $1 + 0 = 1$
 $1 + 1 = 0$ avec une retenue de 1

70 Prof. Mohamed BOUDCHICHE

70

Opérations usuelles en binaire

► **Exemple**

$$\begin{array}{r} 1111 \longrightarrow 15 \\ + 101 \longrightarrow 5 \\ \hline = 10100 \longrightarrow 20 \end{array}$$

► On procède comme en décimal. Si le résultat de la somme d'une colonne est supérieur à 1, on passe ce bit au voisin de gauche.

71 Prof. Mohamed BOUDCHICHE

71

Opérations usuelles en binaire

► **Table de multiplication binaire :**

$0 \times 0 = 0$
 $0 \times 1 = 0$
 $1 \times 0 = 0$
 $1 \times 1 = 1$

► Dans la multiplication binaire, on procède comme en décimal.

72 Prof. Mohamed BOUDCHICHE

72

Opérations usuelles en binaire

Exemple

$$\begin{array}{r}
 1111 \longrightarrow 15 \\
 \times 101 \longrightarrow 5 \\
 \hline
 1111 \\
 0000 \\
 1111 \\
 \hline
 = 1001011 \longrightarrow 75
 \end{array}$$

73 Prof. Mohamed BOUDCHICHE

73

Exercices

- Sur 8 bits, calculer la somme binaire des nombres : 10010110 et 01110101. Commenter le résultat.
- Faites la multiplication (1011 * 1010). Le résultat est-il vrai sur 4 bit, sur 7 bit, sur 8 bit?

74 Prof. Mohamed BOUDCHICHE

74

Opérations usuelles en binaire

Soustraction

Dans la soustraction binaire, on peut procéder comme en décimale. Quand la quantité à soustraire est supérieure à la quantité dont on soustrait, on emprunte 1 au voisin de gauche.

En binaire, le 1 emprunté va "ajouter 2" à la quantité dont on soustrait, tandis qu'en décimal il ajoute 10.

75 Prof. Mohamed BOUDCHICHE

75

Opérations usuelles en binaire

Soustraction

Dans l'exemple suivant, on doit faire la soustraction 0-1 pour la 1ère colonne. On emprunte 1 du bit de gauche et on a alors 10-1, i.e. 2-1=1. Ensuite, on retranche cet emprunt du bit de gauche, et on a 1-0-1=0

$$\begin{array}{r}
 10 \\
 - 01 \\
 \hline
 = 01
 \end{array}$$

76 Prof. Mohamed BOUDCHICHE

76

Opérations usuelles en binaire

Exemple

$$\begin{array}{r}
 10100110 \\
 - 00111100 \\
 \hline
 = 01101010
 \end{array}$$

77 Prof. Mohamed BOUDCHICHE

77

Opérations usuelles en binaire

La division

La division binaire s'effectue à l'aide des soustractions et des décalages, comme la division décimale, sauf que les digits du quotient ne peuvent être que 1 ou 0.

Le bit du quotient est 1 si on peut soustraire le diviseur, sinon il est 0.

78 Prof. Mohamed BOUDCHICHE

78

Opérations usuelles en binaire

Exemple

$$\begin{array}{r} 10110 \\ - 11 \\ \hline 0101 \\ - 11 \\ \hline 0100 \\ - 11 \\ \hline 001 \end{array} \quad \begin{array}{r} 11 \\ 0111 \end{array}$$

$$10110 = 11 \times 111 + 1$$

79 Prof. Mohamed BOUDCHICHE

79

Exercices

- Faites les soustractions binaires suivantes :

$$100 - 010$$

$$110 - 010$$

$$1010 - 0111$$
- Faites les divisions binaires suivantes :

$$1001 \div 11$$

$$1100 \div 10$$

$$10010000111 \div 1011$$

80 Prof. Mohamed BOUDCHICHE

80

Solutions

$$\begin{array}{r} 100 \\ - 010 \\ \hline 010 \end{array} \quad \begin{array}{r} 110 \\ - 010 \\ \hline 100 \end{array} \quad \begin{array}{r} 1010 \\ - 0111 \\ \hline 011 \end{array}$$

$$\begin{array}{r} 1001 \\ - 11 \\ \hline 100 \\ - 11 \\ \hline 0011 \\ - 11 \\ \hline 00 \end{array} \quad \begin{array}{r} 11 \\ 011 \end{array} \quad \begin{array}{r} 1100 \\ - 10 \\ \hline 10 \\ - 10 \\ \hline 00 \\ - 10 \\ \hline 00 \end{array} \quad \begin{array}{r} 10 \\ 110 \end{array}$$

$$\begin{array}{r} 1001000011 \\ - 1011 \\ \hline 10010 \\ - 1011 \\ \hline 1110 \\ - 1011 \\ \hline 0110 \\ - 1011 \\ \hline 1100 \\ - 1011 \\ \hline 11 \\ - 11 \\ \hline 1011 \\ - 11 \\ \hline 1011 \\ - 1111 \\ \hline 1011 \\ - 1011 \\ \hline 100 \end{array} \quad \begin{array}{r} 1011 \\ 01101001 \end{array}$$

81 Prof. Mohamed BOUDCHICHE

81

09 Codage des caractères

82 Prof. Mohamed BOUDCHICHE

82

Codage des caractères

- On désigne par caractère:
 - Un alphabet de lettres minuscules = {a,b,c,...,z} soient 26 caractères.
 - Un alphabet de lettres majuscules = {A,B,C,...,Z} soient 26 caractères.
 - Des chiffres {0,1,...,9} soient 10 caractères.
 - Des symboles syntaxiques {? , ; (" : + , - , * , / , ... au minimum 10 caractères.
 - TOTAL minimal = 72 caractères.

83 Prof. Mohamed BOUDCHICHE

83

Codage des caractères

- Le codage des caractères est fait par une table de correspondance indiquant la configuration binaire représentant chaque caractère (chaque caractère possède donc son équivalent en code numérique).
- Parmi les codages les plus connus et utilisés, le codage **ASCII** (American Standard Code for Information Interchange).

84 Prof. Mohamed BOUDCHICHE

84

Codage des caractères

- Si l'on avait choisi un code à 6 bits le nombre de caractères codifiables aurait été de $2^6 = 64$, nombre donc insuffisant pour nos besoins. Il faut au minimum 1 bit de plus qui permet de définir ainsi $2^7 = 128$ nombres binaires différents, autorisant le codage de 128 caractères différents.
- Initialement le code ASCII est un code à 7 bits (128 caractères) ; il a été étendu à un code 8 bits ($2^8 = 256$ caractères) permettant le codage des caractères français (comme : ù, à, è, é, â, ... etc).

85 Prof. Mohamed BOUDCHICHE

85

Codage des caractères

- Les codes 0 à 31 ne sont pas des caractères. On les appelle caractères de contrôle car ils permettent de faire des actions telles que : retour à la ligne (CR), Bip sonore (BEL),
- Les codes 65 à 90 représentent les majuscules
- Les codes 97 à 122 représentent les minuscules.
- Pour les autres codes voir la table des caractères ASCII.

86 Prof. Mohamed BOUDCHICHE

86

Codage des caractères

Code ASCII d'une lettre :

- Code ASCII d'une lettre minuscule = Code ASCII de cette lettre en majuscule + 32
- Le codage binaire d'une lettre minuscule est déduit de celle de la lettre majuscule en modifiant le 6^{ème} bit (à partir de la droite) à 1 au lieu de 0.

Exemples :

- La lettre A (majuscule) a pour code ASCII 65, soit 01000001.
- La lettre a (minuscule) a pour code ASCII 97, soit 01100001
- La lettre B (majuscule) a pour code ASCII 66, soit 01000010.
- La lettre b (minuscule) a pour code ASCII 98, soit 01100010.

87 Prof. Mohamed BOUDCHICHE

87

Codage des caractères

Code ASCII d'un chiffre :

- Code ASCII d'un chiffre = Ce chiffre + 48

Exemples :

- Le chiffre 1 a pour code ASCII 49, soit 00110001
- Le chiffre 2 a pour code ASCII 50, soit 00110010

Remarque :

- La touche ALT du clavier suivie du code ASCII d'un caractère affiche ce caractère.

88 Prof. Mohamed BOUDCHICHE

88

Codage des caractères

- Il existe un codage récent dit « universel » : il s'agit du codage **Unicode** sur 16 bits ($2^{16} = 65536$ caractères). Ce codage permet de représenter n'importe quel caractère par un code sur 16bits, indépendamment de tout système d'exploitation ou langage de programmation. Il regroupe ainsi la quasi-totalité des alphabets existants (arabe, grec, hébreu, latin, ...) et il est compatible avec le code ASCII.

89 Prof. Mohamed BOUDCHICHE

89

10 Opérations logiques

90 Prof. Mohamed BOUDCHICHE

90

Opérations logiques

- ▶ En logique binaire, une variable logique (Booléenne) est vraie (TRUE=1) ou Fausse (FALSE=0).
- ▶ Une expression logique est constituée de plusieurs variables logiques combinées par des connecteurs (opérateurs) logiques.
- ▶ Les opérateurs élémentaires logiques sont :
 - ▶ NON [NOT]
 - ▶ ET [AND]
 - ▶ OU [OR]

91 Prof. Mohamed BOUDCHICHE

91

Opérations logiques

- ▶ **Table de vérité de ET :**
 - 0 ET 0 = 0
 - 0 ET 1 = 0
 - 1 ET 0 = 0
 - 1 ET 1 = 1
- ▶ **Table de vérité de OU :**
 - 0 OU 0 = 0
 - 0 OU 1 = 1
 - 1 OU 0 = 1
 - 1 OU 1 = 1

92 Prof. Mohamed BOUDCHICHE

92

Opérations logiques

- ▶ **Table de vérité de NON :**
 - NON 0 = 1
 - NON 1 = 0
- ▶ **Exemple :**
 - (12 est multiple de 4) ET (4 est un nombre premier) = 0
 - (55 est un nombre pair) OU (99 est multiple de 3) = 1
 - NON (11 est divisible par 4) = 1

93 Prof. Mohamed BOUDCHICHE

93

11 Exercices

94 Prof. Mohamed BOUDCHICHE

94

Questions :

- ▶ Quelle est la première génération d'ordinateurs ?

- 1 Les ordinateurs à transistors.
- 2 Les ordinateurs à circuits intégrés.
- 3 Les ordinateurs à tubes à vide.
- 4 Les microprocesseurs.

95 Prof. Mohamed BOUDCHICHE

95

Questions :

- ▶ Quelle est la principale limitation des ordinateurs à tubes à vide ?

- 1 Ils sont lents.
- 2 Ils sont encombrants, consomment beaucoup d'énergie et sont généralement peu fiables.
- 3 Ils ne sont pas capables d'effectuer des calculs complexes.
- 4 Ils ne sont pas compatibles avec les logiciels modernes.

96 Prof. Mohamed BOUDCHICHE

96

Questions :

▶ Quelle est la principale avancée de la deuxième génération d'ordinateurs ?

- 1 L'utilisation des transistors a permis de rendre les ordinateurs plus petits et plus fiables.
- 2 L'utilisation des circuits intégrés a permis de réduire la taille des ordinateurs.
- 3 L'utilisation des microprocesseurs a permis de miniaturiser les ordinateurs.
- 4 L'utilisation de l'intelligence artificielle a permis de rendre les ordinateurs plus intelligents

97 Prof. Mohamed BOUDCHICHE

97

Questions :

▶ Quelle est la fonction du processeur ?

- 1 Il stocke les données et les programmes en cours d'exécution.
- 2 Il exécute les instructions des programmes.
- 3 Il gère les ressources matérielles.
- 4 Il permet à l'ordinateur d'interagir avec l'utilisateur et le monde extérieur.

98 Prof. Mohamed BOUDCHICHE

98

Questions :

▶ Quelle est la fonction de la mémoire RAM ?

- 1 Elle permet à l'ordinateur d'interagir avec l'utilisateur et le monde extérieur.
- 2 Elle exécute les instructions des programmes.
- 3 Elle gère les ressources matérielles.
- 4 Elle stocke temporairement les données et les programmes en cours d'exécution.

99 Prof. Mohamed BOUDCHICHE

99

Questions :

▶ Qu'est-ce qu'un système d'exploitation ?

- 1 C'est un programme qui permet à l'utilisateur d'interagir avec l'ordinateur.
- 2 C'est un logiciel qui permet d'exécuter des programmes.
- 3 C'est un logiciel qui stocke les données et les programmes en cours d'exécution.
- 4 C'est un logiciel qui gère les ressources matérielles et facilite l'exécution d'applications.

100 Prof. Mohamed BOUDCHICHE

100

Questions :

▶ Quel est le principal avantage des disques SSD par rapport aux disques durs ?

- 1 La capacité de stockage.
- 2 La fiabilité.
- 3 La vitesse d'accès.
- 4 La consommation d'énergie.

101 Prof. Mohamed BOUDCHICHE

101

Questions :

▶ Dans quelle architecture, la CPU et la mémoire partagent le même bus de données ?

- 1 Harvard.
- 2 Von Neumann.
- 3 Pipeline de traitement.
- 4 Coeurs multiples.

102 Prof. Mohamed BOUDCHICHE

102

Questions :

► Quelle est la différence entre un processeur mono-cœur et un processeur multi-cœur ?

- 1 Un processeur mono-cœur peut exécuter une instruction à la fois, tandis qu'un processeur multi-cœur peut exécuter plusieurs instructions simultanément.
- 2 Un processeur mono-cœur est plus rapide qu'un processeur multi-cœur.
- 3 Un processeur multi-cœur est plus cher qu'un processeur mono-cœur.

103 Prof. Mohamed BOUDCHICHE

103

Questions :

► Quel est le rôle de la mémoire cache dans un ordinateur ?

- 1 Elle stocke les instructions du programme en cours d'exécution.
- 2 Elle stocke des données fréquemment utilisées pour un accès rapide.
- 3 Elle stocke les données du système d'exploitation.

104 Prof. Mohamed BOUDCHICHE

104