

PATRICK CLOUDY

**L'archi : C'est pas
sorcier !**



EDITIONS
FAMILINEC

Conversions

Binaire vers hexadécimal

Pour convertir un nombre binaire en hexadécimal, il faut le diviser en groupes de 4 chiffres binaires, puis convertir chaque groupe en un chiffre hexadécimal. Par exemple, pour convertir le nombre binaire 11011010 en hexadécimal, on peut le diviser en 1101 et 1010, qui correspondent respectivement aux chiffres hexadécimaux D et A. Ainsi, 11011010 en binaire est équivalent à DA en hexadécimal.

Hexadécimal vers binaire

Pour convertir un nombre hexadécimal en binaire, il faut convertir chaque chiffre hexadécimal en un groupe de 4 chiffres binaires. Par exemple, pour convertir le nombre hexadécimal 3F en binaire, on peut convertir 3 en 0011 et F en 1111. Ainsi, 3F en hexadécimal est équivalent à 00111111 en binaire.

Décimal vers hexadécimal

Pour convertir un nombre décimal en hexadécimal, il faut le diviser par 16 jusqu'à ce que le quotient soit inférieur à 16. Les restes de chaque division, lus de bas en haut, correspondent aux chiffres hexadécimaux. Par exemple, pour convertir le nombre décimal 255 en hexadécimal, on peut effectuer les divisions suivantes:

$$255 / 16 = 15 \text{ reste } 15 \text{ (F)}$$

$$15 / 16 = 0 \text{ reste } 15 \text{ (F)}$$

Ainsi, 255 en décimal est équivalent à FF en hexadécimal.

Hexadécimal vers décimal

Pour convertir un nombre hexadécimal en décimal, il faut multiplier chaque chiffre hexadécimal par la puissance de 16 correspondante, en commençant par la droite. Par exemple, pour convertir le nombre hexadécimal 3F en décimal, on peut effectuer les multiplications suivantes:

$$3 * 16^1 = 48$$

$$F * 16^0 = 15$$

Ensuite, on additionne les résultats: $48 + 15 = 63$. Ainsi, 3F en hexadécimal est équivalent à 63 en décimal.

Opérations binaires

Not

L'opération Not est un opérateur unaire qui inverse tous les bits d'un nombre binaire. Par exemple, si on applique l'opération Not au nombre binaire 1101, on obtient 0010.

And

L'opération And est un opérateur binaire qui compare les bits de deux nombres binaires et renvoie un nouveau nombre binaire dont chaque bit est égal à 1 si les bits correspondants des deux nombres sont également égaux à 1. Par exemple, si on applique l'opération And aux nombres binaires 1101 et 1011, on obtient 1001.

Or

L'opération Or est un opérateur binaire qui compare les bits de deux nombres binaires et renvoie un nouveau nombre binaire dont chaque bit est égal à 1 si au moins l'un des bits correspondants des deux nombres est égal à 1. Par exemple, si on applique l'opération Or aux nombres binaires 1101 et 1011, on obtient 1111.

Xor

L'opération `Xor` est un opérateur binaire qui compare les bits de deux nombres binaires et renvoie un nouveau nombre binaire dont chaque bit est égal à 1 si les bits correspondants des deux nombres sont différents. Par exemple, si on applique l'opération `Xor` aux nombres binaires `1101` et `1011`, on obtient `0110`.

Décalage à droite

L'opération de décalage à droite est un opérateur binaire qui décale tous les bits d'un nombre binaire vers la droite d'un certain nombre de positions. Les bits décalés sont remplacés par des zéros. Par exemple, si on applique l'opération de décalage à droite de 2 positions au nombre binaire `1101`, on obtient `0011`.

Encodages

- Encoder une liste de noms au format LDM, un format binaire simple qui stocke le nombre de mots, la taille de chaque mot et la chaîne de caractères correspondant au mot.
- **Format LDM** : `nb_mot taille_mot1 mot1 taille_mot2 mot2 ...`, où chaque élément est écrit directement à la suite du précédent dans l'écriture binaire. `nb_mot` est un entier sur 4 octets en signed-32, `taille_mot` est un entier sur 2 octets en signed-16, et `mot` est une chaîne de caractères en UTF-8.
- **Little-endian** : Tous les entiers sont stockés en little-endian, c'est-à-dire que le bit de poids faible est stocké en premier et le bit de poids fort en dernier.
-
- **Exemple** : La liste ["Alice", "Bob", "Charles"] serait encodée comme suit :
00000000 00000000 00000011 00000101 00000000 01000001 01101100 01101001
01100011 01100101 00000011 00000000 01000010 01101111 01100010 00000111
00000000 01000011 01101000 01100001 01110010 01101100 01100101 01110011

Big-endian

- Le big-endian est une convention d'ordre des octets dans laquelle l'octet le plus significatif d'un nombre est stocké ou transmis en premier, à l'adresse mémoire la plus petite ou au début du flux de données.
- Par exemple, le nombre décimal 1234, qui s'écrit en hexadécimal 04D2, est stocké en big-endian comme suit : 04 D2. L'octet 04, qui correspond aux milliers et aux centaines, est placé avant l'octet D2, qui correspond aux dizaines et aux unités.

Décomposition d'une image bitmap

- Une image bitmap, ou matricielle, est une image numérique composée d'un ensemble de pixels, chacun ayant une couleur définie par un code binaire.
- Pour décomposer une image bitmap, il faut connaître les informations stockées dans son en-tête et dans ses données. L'en-tête contient des informations générales sur l'image, comme sa taille, son format, sa résolution, etc. Les données contiennent les valeurs binaires des pixels, selon l'ordre et le codage des couleurs définis dans l'en-tête.
- Par exemple, une image bitmap de 4 pixels sur 4, en niveaux de gris, avec 2 bits par pixel, peut avoir l'en-tête et les données suivants :

En-tête	Valeur	Description
Signature	BM	Identifie le fichier comme un bitmap
Taille	$14 + 40 + 16 = 70$	Taille du fichier en octets
Champ réservé	00 00 00 00	Non utilisé
Offset	$14 + 40 = 54$	Position des données en octets
Taille en-tête	40	Taille de l'en-tête en octets
Largeur	4	Largeur de l'image en pixels
Hauteur	4	Hauteur de l'image en pixels
Plans	1	Nombre de plans de couleur
Bits par pixel	2	Nombre de bits pour coder la couleur d'un pixel
Compression	0	Type de compression utilisé
Taille image	16	Taille des données en octets
Résolution horizontale	0	Résolution horizontale en pixels par mètre
4		
Résolution verticale	0	Résolution verticale en pixels par mètre

En-tête	Valeur	Description
Nombre de couleurs	4	Nombre de couleurs dans la palette
Couleurs importantes	0	Nombre de couleurs importantes

Données	Valeur	Description
Palette	00 00 00 00	Couleur 0 : noir
	55 55 55 00	Couleur 1 : gris foncé
	AA AA AA 00	Couleur 2 : gris clair
	FF FF FF 00	Couleur 3 : blanc
Pixels	00 01 10 11	Première ligne de pixels : noir, gris foncé, gris clair, blanc
	11 10 01 00	Deuxième ligne de pixels : blanc, gris clair, gris foncé, noir
	01 01 01 01	Troisième ligne de pixels : gris foncé, gris foncé, gris foncé, gris foncé
	10 10 10 10	Quatrième ligne de pixels : gris clair, gris clair, gris clair, gris clair

1. Quelle est la convention d'ordre des octets dans laquelle l'octet le plus significatif d'un nombre est stocké ou transmis en premier ?
 - o A) Big-endian
 - o B) Little-endian
 - o C) Middle-endian
 - o D) Mixed-endian
2. Quelle est la formule pour convertir un nombre décimal en hexadécimal ?
 - o A) Diviser le nombre par 16 jusqu'à ce que le quotient soit inférieur à 16, et lire les restes de bas en haut
 - o B) Diviser le nombre par 16 jusqu'à ce que le quotient soit égal à 0, et lire les restes de haut en bas
 - o C) Multiplier le nombre par 16 jusqu'à ce que le produit soit supérieur à 16, et lire les quotients de bas en haut
 - o D) Multiplier le nombre par 16 jusqu'à ce que le produit soit égal à 0, et lire les quotients de haut en bas
3. Quelle est l'opération binaire qui inverse tous les bits d'un nombre binaire ?
 - o A) Not
 - o B) And
 - o C) Or
 - o D) Xor
4. Quelle est l'opération binaire qui compare les bits de deux nombres binaires et renvoie un nouveau nombre binaire dont chaque bit est égal à 1 si au moins l'un des bits correspondants des deux nombres est égal à 1 ?
 - o A) Not
 - o B) And
 - o C) Or
 - o D) Xor
5. Quelle est l'opération binaire qui décale tous les bits d'un nombre binaire vers la droite d'un certain nombre de positions ?
 - o A) Not
 - o B) And
 - o C) Or
 - o D) Décalage à droite
6. Quel est le format binaire simple qui stocke le nombre de mots, la taille de chaque mot et la chaîne de caractères correspondant au mot ?
 - o A) LDM
 - o B) BMP
 - o C) UTF-8
 - o D) Signed-32
7. Quel est le format d'encodage des caractères qui utilise un nombre variable d'octets pour représenter chaque caractère ?
 - o A) LDM
 - o B) BMP
 - o C) UTF-8
 - o D) Signed-32
8. Quel est le format d'image numérique composé d'un ensemble de pixels, chacun ayant une couleur définie par un code binaire ?
 - o A) LDM
 - o B) BMP
 - o C) UTF-8
 - o D) Signed-32
9. Quel est le type de compression utilisé pour réduire la taille d'un fichier sans perte de qualité ?

- ☐ A) Compression avec perte
 - ☐ B) Compression sans perte
 - ☐ C) Compression par hachage
 - ☐ D) Compression par cryptage
- o
10. Quel est le protocole de communication qui utilise le big-endian pour transmettre les données sur Internet ?
- ☐ A) TCP/IP
 - ☐ B) HTTP
 - ☐ C) FTP
 - ☐ D) SMTP

Réponses :

1) A 2) A 3) A 4) C 5) D 6) A 7) C 8) B 9) B 10) A