

3. On démontre que tout nombre entier peut s'écrire de manière unique sous la forme d'une somme de puissances de 2. Pour écrire un nombre **décimal** en **binaire** on utilise la propriété précédente. On code par le chiffre 1 la présence d'une puissance de 2 et par 0 son absence. Par exemple en écrivant le nombre 45 sous la forme $32 + 8 + 4 + 1$ on peut le compléter le tableau suivant :

Puissances de 2	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Valeur décimale	128	64	32	16	8	4	2	1
Décomposition décimale de 45			32		8	4		1
Écriture binaire de 45	0	0	1	0	1	1	0	1
Décomposition décimale de 1								
Écriture binaire de 1								
Décomposition décimale de 2								
Écriture binaire de 2								
Décomposition décimale de 3								
Écriture binaire de 3								
Décomposition décimale de 4								
Écriture binaire de 4								
Décomposition décimale de 5								
Écriture binaire de 5								
Décomposition décimale de 17								
Écriture binaire de 17								
Décomposition décimale de 63								
Écriture binaire de 63								
Décomposition décimale de 200								
Écriture binaire de 200								
Décomposition décimale de								
Écriture binaire de	1	1	1	0	1	0	0	1

4. En vous inspirant du tableau ci-dessus, compter de 1 à 20 en binaire.

Pour simplifier la compréhension et le stockage des bits on les regroupe par paquet de 8. On appelle cela un octet. Par exemple 00101110 est un octet puisqu'il est constitué de 8 bit. On ajoute des « zéros inutiles » à gauche pour obtenir 8 bits.

5. Quel est le plus grand nombre entier que l'on peut coder avec un octet ?

6. Numériser l'information « VIVE LA TECHNOLOGIE ! » en regroupant les bits en octet. Vous utiliserez pour cela le codage ASCII obtenu à la question 1.. Combien d'octets sont nécessaires à cette numérisation ?

7. Décoder le message suivant présenté sous forme de bits regroupés en octet.

01001111 01001110 00100000 01000001 01000100 01001111 01010010 01000101 00100000 01001101 01000101 01001100 01000001 01001110
 01000111 01000101 01010010 00100000 01010100 01000101 01000011 01001000 01001110 01001111 00100000 01000101 01010100 00100000
 01001101 01000001 01010100 01001000 01010011 00100000 00100001

Voici les unités habituelles utilisées en informatique :

o octet
ko kilo octet 1000 o
Mo Méga octet 1 000 000 o
Go Giga octet 1 000 000 000 o
To Téra octet 1 000 000 000 000 o
Po Péta octet 1 000 000 000 000 000 o

Une clé USB de 64 Go peut stocker :

- 64 000 000 000 de caractères soit 512 000 000 000 de bits ;
- 25 films en HD ;
- 16 000 photos d'excellente qualité ;
- 10 000 livres de 400 pages.