

# Les fichiers informatiques

## « numériques »

« Le langage de l'ordinateur ne comprend que 2 lettres ( 0 et 1 ) »

Cf : processeur

Pour exprimer

**Des nombres entiers**

2345

100100101001



# Exemple des nombre entiers

## Nous comptons en Base 10

Systeme décimal

Nous pouvons exprimer tous les nombres à l'aide de 10 symboles « chiffres »

actuellement 0 1 2 3 4 5 6 7 8 9

**Lorsque le nombre dépasse 9**

→ On les combine : règle de positionnement

10 11 12 13 14 15 16 17 18 19

0 unité et une fois un paquet de 10

3 unités et une fois un paquet de 10

Une manière simple  
d'exprimer des  
« choses  
complexes »

ou 2345

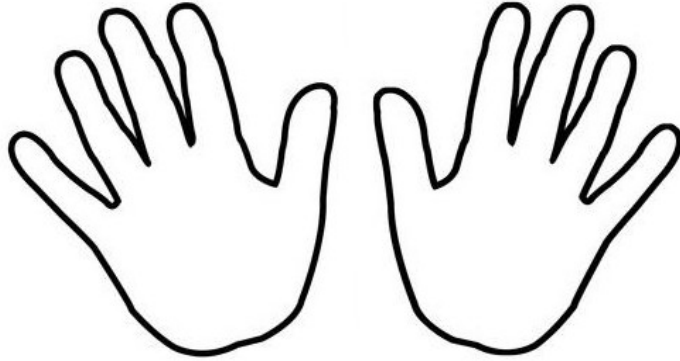
**codage**

5 unités et 4 fois un paquet de 10 et 3 fois un  
paquet de 100 et deux fois un paquet de 1000

## ***Le coin des Curieux***

**Questions ?** *Pourquoi la base 10 : système décimal ?*

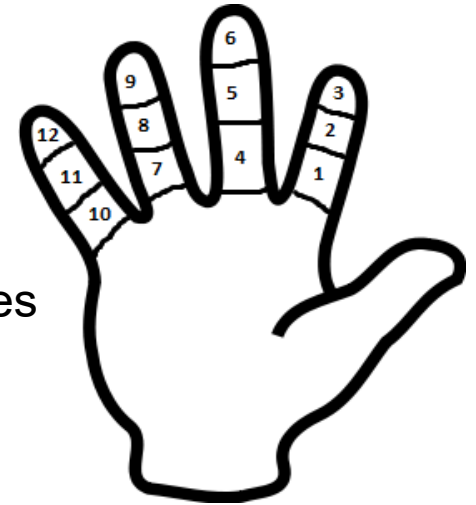
*Parce que :*



*D'autres systèmes de bases ont existé ...*

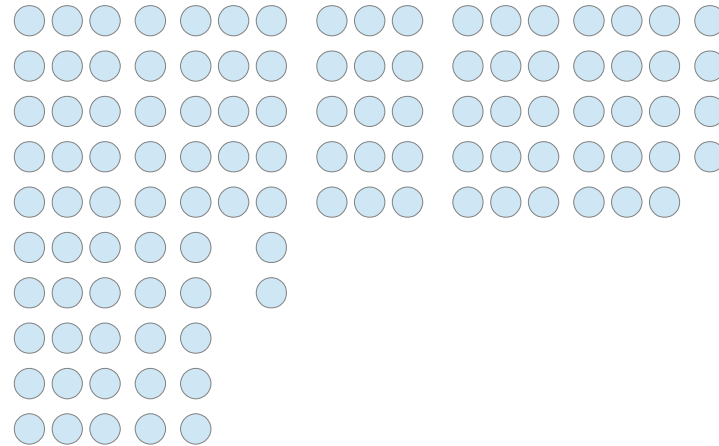
Base duodécimale (12)

Reste : 12 mois de l'année, douzaine d'oeufs, 12 signes astrologiques



## Le principe du Comptage en base 10

Exprimer le nombre de « ronds » présents sur la figure suivante

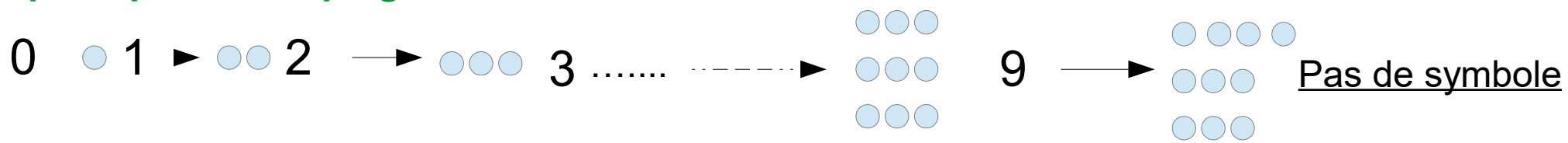


Avec mes dix chiffres 0 1 2 3 4 5 6 7 8 9

Avec le principe de positionnement

*pas évident : On se met à la place d'une personne qui ne connaît pas le principe et qui doit le découvrir*

# Le principe du Comptage en base 10

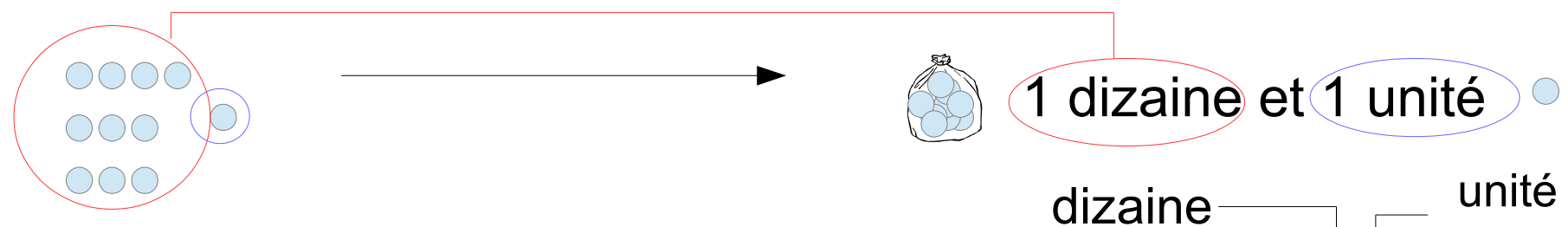


Comment faire ?

On a introduit la dizaine →  Paquet = dizaine *Et 0 unité*

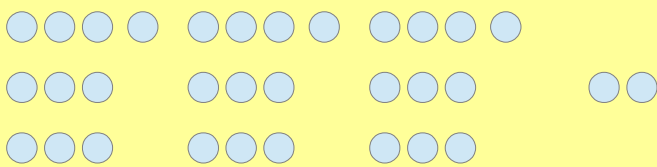

Et on écrit grâce à la règle du positionnement dizaine ———— | ———— unité  
**10**

Un « rond » de plus



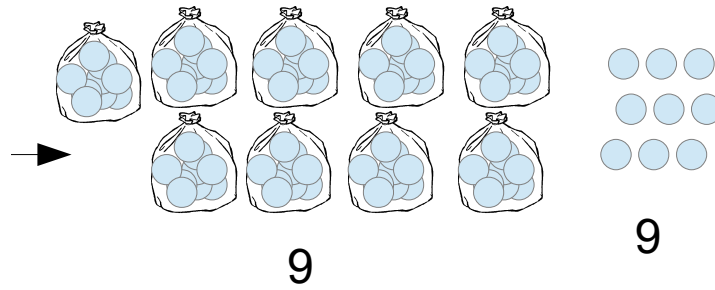
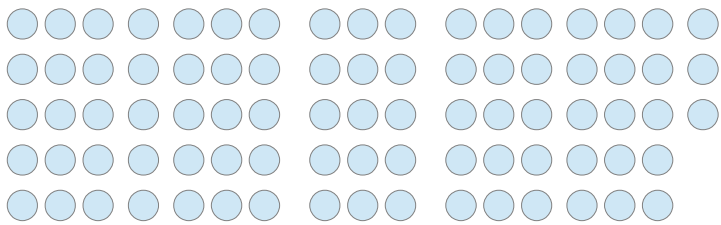
dizaine ———— | ———— unité  
*S'écrit grâce à la règle du positionnement* **11**

..... ► Autre exemple

  **3 dizaines et 2 unités**

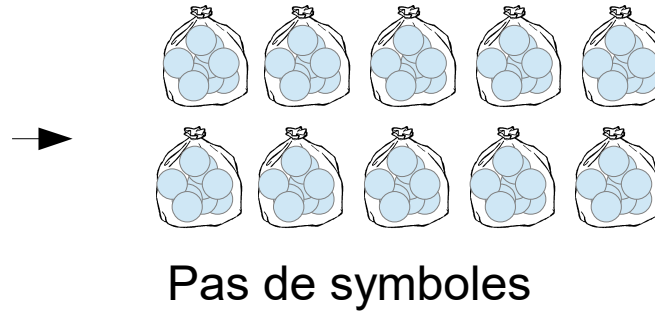
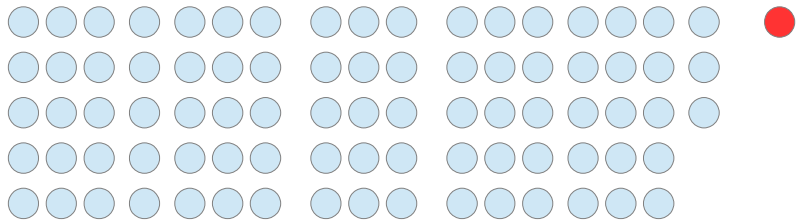
dizaine ———— | ———— unité  
*S'écrit grâce à la règle du positionnement* **32**

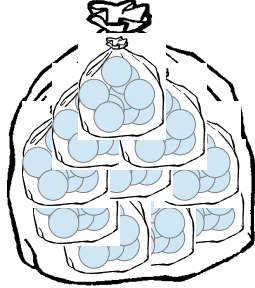
..... ►



dizaine    unité  
**99**

Et après un rond de plus    *Plus de chiffre*

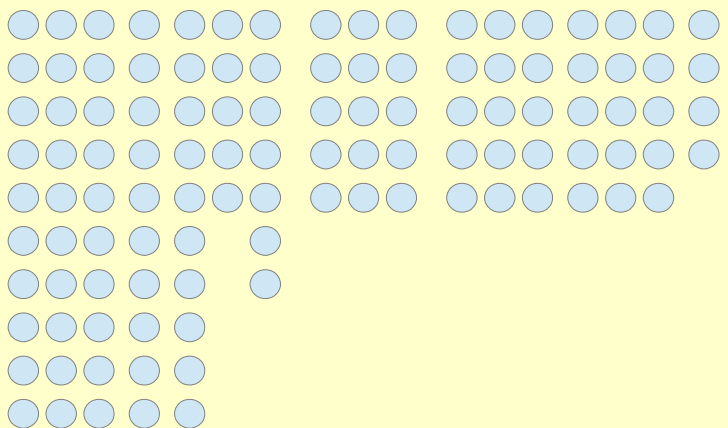


On introduit  
  
= 1 centaine

dizaine    unité  
centaine    **100**

*S'écrit grâce à la règle du positionnement*

Ainsi



L'ordinateur n'a pas **dix symboles** mais que **deux** symboles pour exprimer tous ces nombres

## PAR COMMODITE on les note : 0 et 1

*En fait on n'aurait pu prendre d'autres symboles pour les exprimer comme ■ ●  
Pour l'ordinateur ces symboles n'ont pas de sens...*

Pour lui : les deux « symboles » expriment le fait qu'un courant électrique passe ou pas

**Un ordinateur est un appareil électrique**

Problème : comment exprimer tous les nombres entiers à l'aide de 2 symboles (0 et 1) ?

**Application :**

- Exprimer le nombre 2
- Exprimer le nombre 4
- Exprimer le nombre 17

*Indice : s'appuyer sur le principe de comptage en base 10*

## Le principe du Comptage en base 2 ou binaire

0   ●   1                    ●●   *Pas de symbole...*

*On introduit le « doublet »*  *Paquet de* ●●

*Comme on a du introduire la dizaine en système décimale*

*Et on écrit en suivant la règle du  
positionnement*

« doublet » ——— | ——— unité  
**10**

Ainsi le 2 (décimal) s'écrit dans le « simple » langage de l'ordinateur : 10

et ●● ?  
●

et ●● ?  
●●  
●



0   ● 1   ●● *Plus de symbole...*

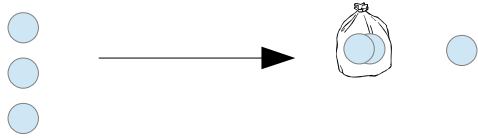
On introduit le « doublet »  Paquet de ●●


*Et grâce à la règle du positionnement*

« doublet » ———— unité  
**10**

Ainsi le 2 (décimal) s'écrit dans le « simple » langage de l'ordinateur : 10

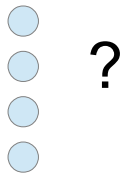
et



 « doublet » ———— unité ●  
**11**

Ainsi le 3 (décimal) s'écrit dans le « simple » langage de l'ordinateur : 11

et



0 1 *Plus de symbole...*

On introduit le « doublet »  Paquet de 




*Et grâce à la règle du positionnement*

« doublet »  unité  
**10**

Ainsi le 2 (décimal) s'écrit dans le « simple » langage de l'ordinateur : 10

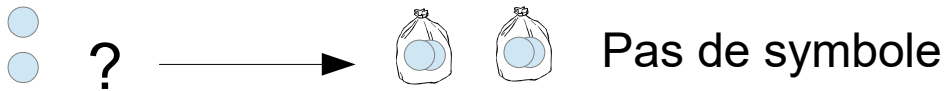
et



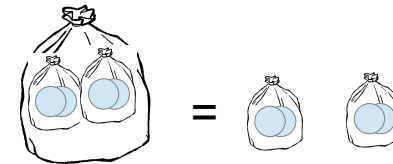
 « doublet »  unité   
**11**

Ainsi le 3 (décimal) s'écrit dans le « simple » langage de l'ordinateur : 11


et



On introduit le « quadruplet »

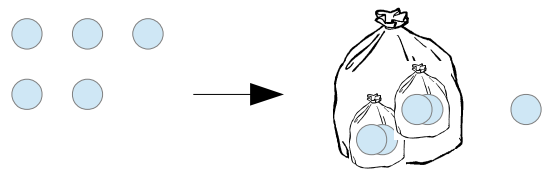


*Et grâce à la règle du positionnement*

quadruplet  doublet  
**100** unité

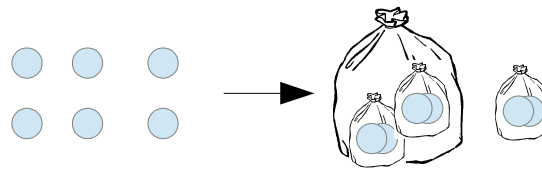
Ainsi le 4 (décimal) s'écrit dans le langage binaire de l'ordinateur : 100

Et ainsi de suite...



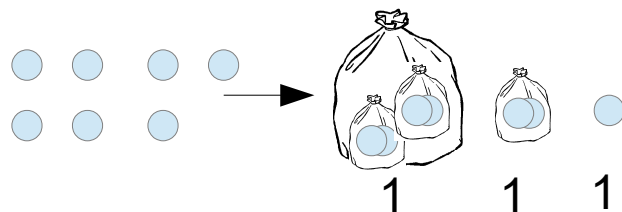
S'écrit

0 doublet ——— 1 unité  
1 quadruplet ——— **101**



S'écrit

1 doublet ——— 0 unité  
1 quadruplet ——— **110**

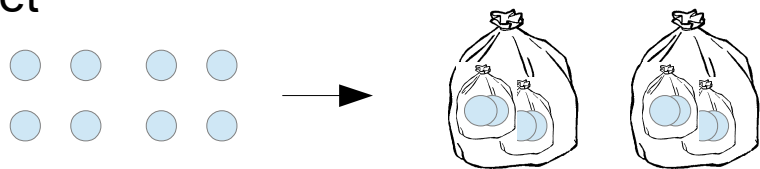


S'écrit

1 doublet ——— 1 unité  
1 quadruplet ——— **111**

Ainsi le 7 (décimal) s'écrit dans le langage binaire : 111

et



On introduit un « huitet »



=2  
paquets  
de 4

« **Dés qu'on doit faire apparaître le chiffre 2 on doit introduire un paquet** »

Et grâce à la règle du positionnement

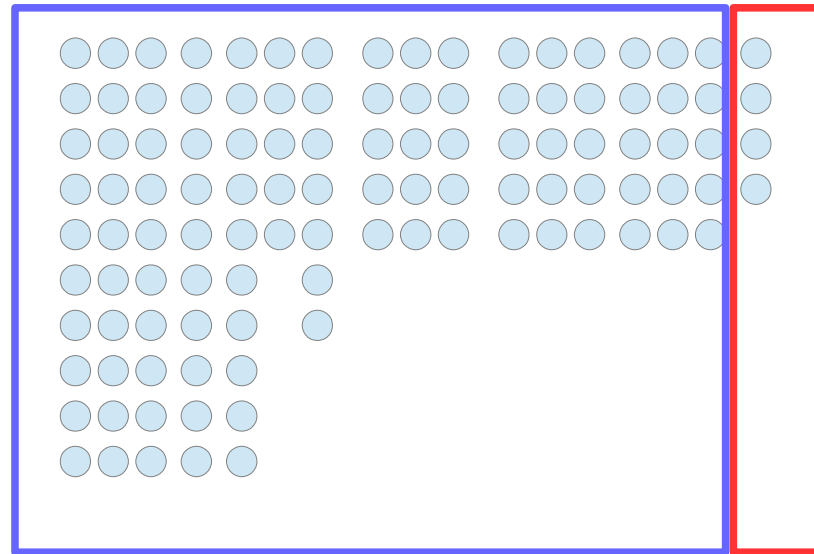
Ainsi le 8 (décimal) s'écrit dans  
le langage binaire : 1000

0 quadruplet 0 doublet  
1 huitet ——— **1000** ——— 0 unité

Exprimer le nombre de « ronds » présent sur la figure suivante

Avec les deux symboles 0 1

Avec le principe de positionnement



*néologismes  
barbarismes*

**1 «centvingthuitet»**

$$128 = 2 \times 64 = 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2$$

0 «soixantequatret»

$$64 = 2 \times 32 = 2 \times 2 \times 2 \times 2 \times 2 \times 2$$

0 «trentedeuxet»

$$32 = 2 \times 16 = 2 \times 2 \times 2 \times 2 \times 2$$

0 «seizet»

$$16 = 2 \times 8 = 2 \times 2 \times 2 \times 2$$

0 «huitet»

$$8 = 2 \times 4 = 2 \times 2 \times 2$$

0 unité

0 « doublet »

**1 «quadruplet»**

$$4 = 2 \times 2$$

# Ainsi

## Humain

10 chiffres pour exprimer tous les nombres...

0 1 2 3 4 5 6 7 8 9

## Système Décimal

0

1

2

3

4

17

Système décimal

1234

## Ordinateurs

2 « chiffres » pour exprimer tous les nombres...

0 1 **Système Binaire**

0

1

10

11

100

10001

Système binaire

10011010010

Codage

1 unité et 1 fois un paquet de 2

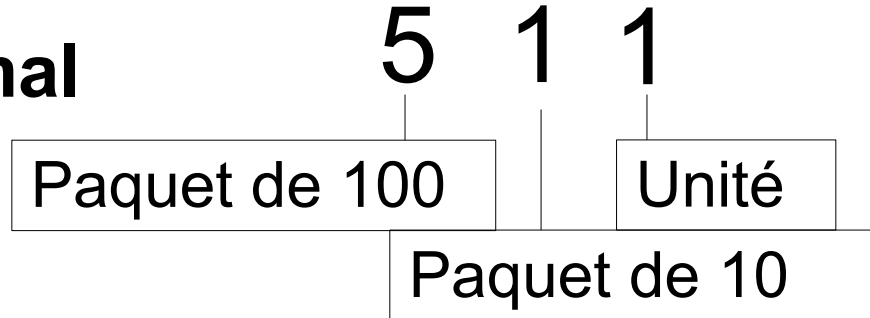
1 unité et 0 fois un paquet de 2 et 0 fois un paquet de 4 et 0 fois un paquet de 8 et 1 fois un paquet de 16

7 unités et 1 fois un paquet de 10

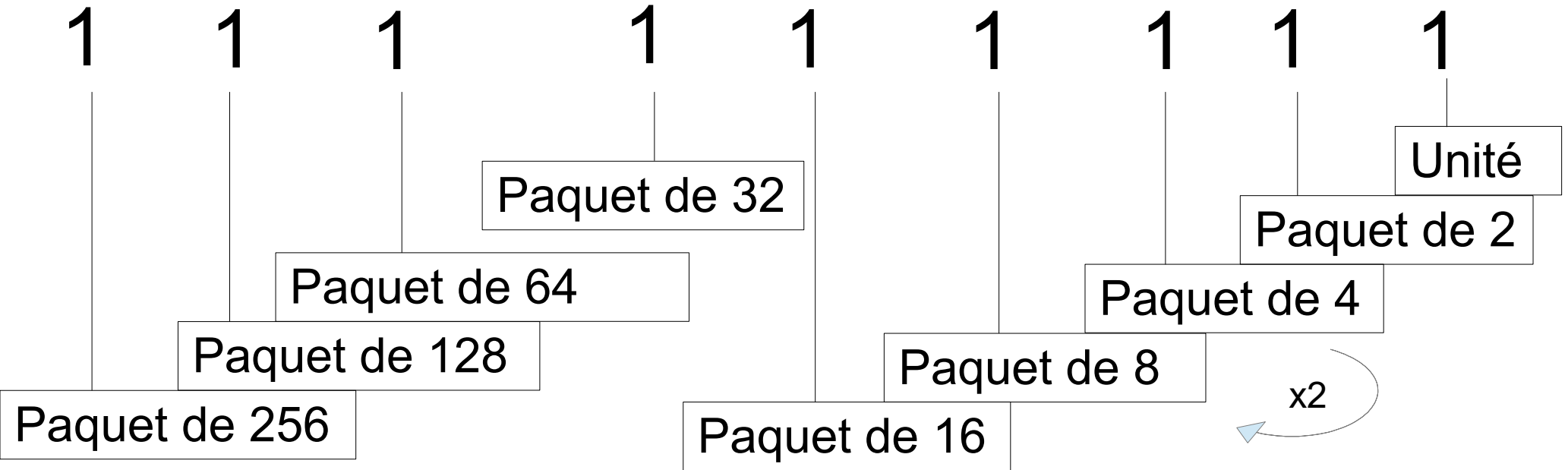
« En binaire l'écriture des nombres se rallonge vite »

Par exemple 511 en décimal

3 chiffres



Alors qu'en binaire



9 « chiffres » ou BIT

27333

110101011000101

Lecture et interprétation

3 (unité)

Et

3 paquets de 10

Et

3 paquets de 100

Et

7 paquets de 1000

Et

2 paquets de

10000

1 unité

Et

0 paquet de 2

Et

1 paquet de 4

Et

0 paquet de 8

Et

0 paquet de 16

Et

0 paquet de 32

Et

1 paquet de 64

Et

1 paquet de 128

Et

0 paquet de 256

Et

1 paquet de 512

Et

0 paquet de 1024

Et

1 paquet de 2048

Et

0 paquet de 4096

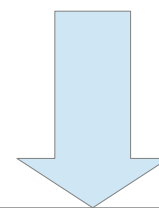
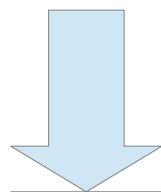
Et

1 paquet de 8192

Et

1 paquet de 16384

*Pas évident et long pour un humain..mais l'ordinateur est beaucoup plus rapide*



27333 unités

Plus petite unité d'information

**binary Unit**

*Chiffre binaire*

« Une lettre de l'ordinateur » : **1 Bit**

0110101011000101

Ce nombre est codé sur 16 Bit

***Le coin des Curieux***

**Questions ?**

Cela complique énormément l'écriture des nombres.....pourquoi est ce comme cela ?

Le principe du positionnement permet de simplifier les opérations de bases  
( additions et multiplications)

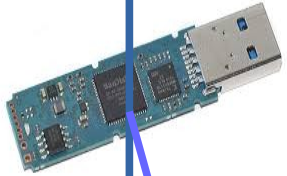
**Cf calculs binaires**



# Lien avec le stockage des fichiers

**Périphérique de stockage**

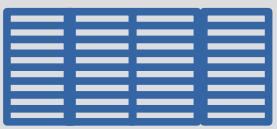
disque dur  
Clef USB  
Mémoire vive



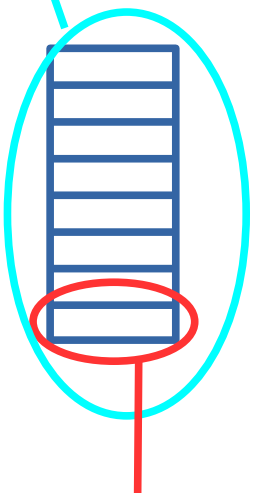
unité de mémoire : **Octet** = 8 bits  
*C'est historique*

Dans 4 Go =  
4000000000 octets  
= 32000000000 bit

Clef USB



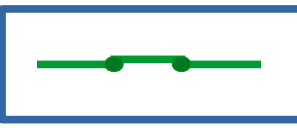
*Des milliards de « cases »*



Vocabulaire :  
**1 case : 1 bit**

Dans un bit soit  
1 soit 0

*Comme un interrupteur*



1



0

- Soit fermé :  
« courant électrique » ~1

Soit ouvert : « pas de  
courant électrique » ~ 0

Enregistrer un fichier c'est comme ouvrir ou fermer des interrupteurs des différents « bits » qui composent l'unité de stockage.

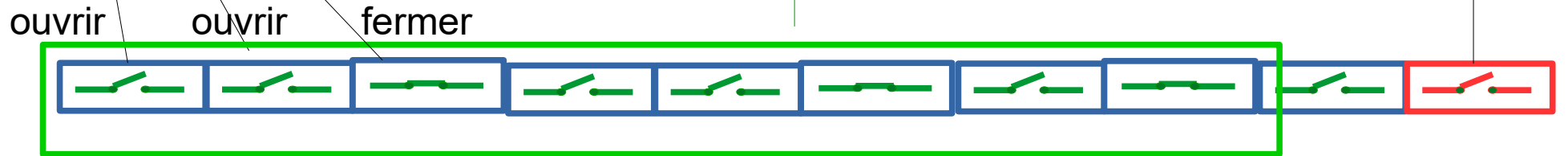
**Cf : Petite anatomie d'unités de stockage**

# Enregistrer un un fichier informatique

001001010100101011100011111...

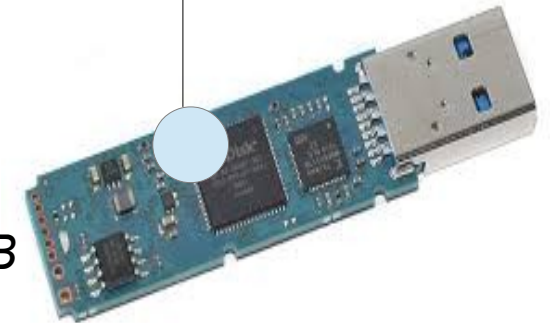
8 BIT = 1 Octet

1 BIT



X 1000000000

*En fait, il n'y a pas vraiment d'interrupteurs, comme ceux que l'on voit sur les murs.. il s'agit d'une image pour comprendre le principe d'enregistrement des fichiers...*



**Cf : Petite anatomie d'unités de stockage**

Capacité clef USB

4 Go = 4000000000 octets  
= 32000000000 bit

Exemple :

Si on enregistre une suite de 35 nombres entiers compris entre 0 et 30000

Il faut 16 bit pour distinguer chacun des nombres 364121993...



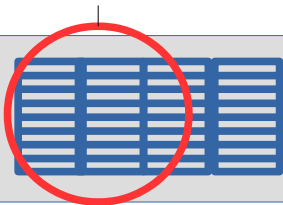
Permet de coder les nombres entre 0 et 32767

Une information (32 bit) codée au début du fichier indique qu'il s'agit de nombres et qu'ils sont codés sur 16 bits

```
0010010101001010111000111111
1110000111000111001010101
0111101001000110001111100011
1110001110000111000111
```

Puis 560 Bits (35 nombres de 16 bits chacun) se succèdent pour indiquer les valeurs... 364121993...

Clef USB



Des milliards de « cases »

Ce fichier numérique prendra une place de 592 bits dans l'unité de stockage

Unité de la taille des fichiers : l'octet = 8 bits

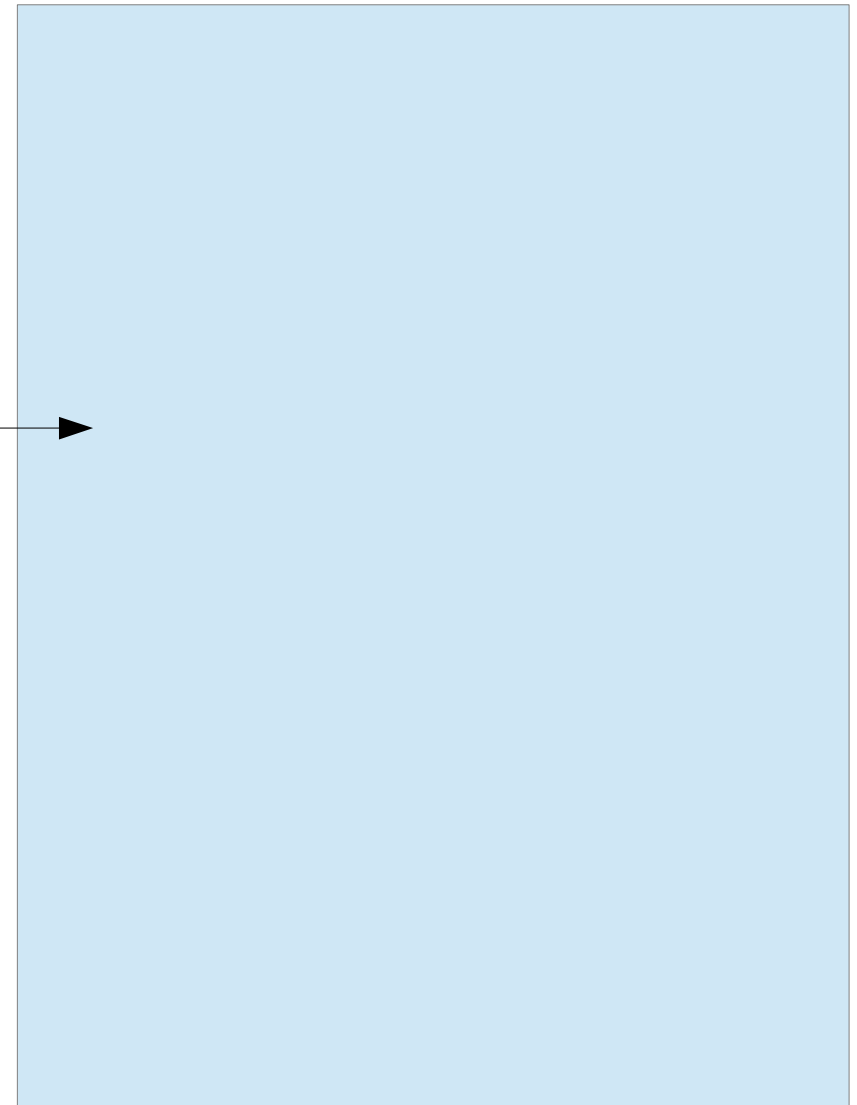
Le fichier occupera 74 octets

# Question bilan 1

## Feuille du classeur

<i>Culture informatique</i>	
<u>Qu'est ce que le Codage binaire ?</u>	
Nom prénom cl :	
<b>A) principe du codage binaire</b>	
Le langage de l'ordinateur ne comprend que deux symboles que l'on note habituellement <u>  </u> et <u>  </u> . Toutes les informations sont donc codées à l'aide de ces deux symboles. On dit que le langage est <u>                  </u>	
<b>B) Exemple des nombres entiers</b>	
Par exemple le nombre 4 ( en système décimal) en langage binaire s'exprime par :	
Par exemple le nombre 10 ( en système décimal) en langage binaire s'exprime par :	
Par exemple le nombre 11 ( en système décimal) en langage binaire s'exprime par :	
<b>C) Taille des fichiers</b>	
Le nombre 27333 s'exprime en langage binaire sous la forme : 110101011000101. On dit qu'il est codé avec <u>        </u> Bits.	
Un octet est une unité qui caractérise la taille d'un fichier informatique il correspond à un «mot» de <u>        </u> BIT(s)	

coller



## Question bilan 1

### **A) principe du codage binaire**

Le langage de l'ordinateur ne comprend que deux symboles que l'on note habituellement \_\_\_ **et** \_\_\_. Toutes les informations sont donc codées à l'aide de ces deux symboles. On dit que le **langage est** \_\_\_\_\_.

### **B) Exemple des nombres entiers**

Par exemple le nombre 4 ( en système décimal) en langage binaire s'exprime par : \_\_\_\_\_

Par exemple le nombre 10 ( en système décimal) en langage binaire s'exprime par : \_\_\_\_\_

Par exemple le nombre 11 (en système décimal) en langage binaire s'exprime par : \_\_\_\_\_

### **C) Taille des fichiers**

Le nombre 27333 s'exprime en langage binaire sous la forme : 110101011000101. On dit qu'il est codé avec \_\_\_\_\_ **BIT(s)**.

Un octet est une unité qui caractérise la taille d'un fichier informatique il correspond à un « mot » de \_\_\_\_\_ BIT(s)

## Question bilan : réponses

### **A) principe du codage binaire**

Le langage de l'ordinateur ne comprend que deux symboles que l'on note habituellement 0 et 1. Toutes les informations sont donc codées à l'aide de ces deux symboles. On dit que le **langage est binaire**

### **B) Exemple des nombres entiers**

Par exemple le nombre 4 ( en système décimal) en langage binaire s'exprime par : **100**

Par exemple le nombre 10 ( en système décimal) en langage binaire s'exprime par : **1010**

Par exemple le nombre 11 (en système décimal) en langage binaire s'exprime par : **1011**

### **C) Taille des fichiers**

Le nombre 27333 s'exprime en langage binaire sous la forme : 110101011000101. On dit qu'il est codé avec 15 **BIT(s)**.

Un octet est une unité qui caractérise la taille d'un fichier informatique il correspond à un « mot » de **\_8\_ BIT(s)**