

# Nombres binaires et systèmes de numération

Présentation est en partie issue de la version du Professeur  
**Ralph Morelli**, réalisée dans le cadre du cours mobile CSP

**Pierre Huguet**

1/11/2018

Pierre Huguet : <http://onvaessayer.org>



# Bases décimales et binaires

- Le système binaire (0,1) est utilisé par les ordinateurs pour représenter toutes les informations
  - nombres
  - textes
  - images
  - vidéos
  - sons
  - instructions des programmes

# Bases décimales et binaires

- Le nombre de chiffres est différent
  - 0,1,2,3,4,5,6,7,8,9 en décimal (10 chiffres)
  - 0,1,2,3,4,5,6,7 en octal (8 chiffres)
  - 0,1 en binaire (2 chiffres)

# Bases décimales et binaires

- Le nombre de chiffre est différent
  - 0,1,2,3,4,5,6,7,8,9 en décimal (10 chiffres)
  - 0,1,2,3,4,5,6,7 en octal (8 chiffres)
  - 0,1 en binaire (2 chiffres)
- Ce sont des systèmes positionnels
  - $132 = 100 + 30 + 2$
  - valeurs en puissances de 10 selon la position
    - ✓ 1, 10, 100, 1000, ... de droite à gauche

# Bases décimales et binaires

- On peut compter dans n'importe quelle base

base 10
0
1
2
3
4
5
6
7
8
9
10
11

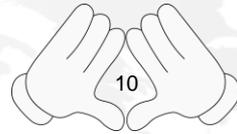
# Bases décimales et binaires

- On peut compter dans n'importe quelle base

base 10	base 8
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	10
9	11
10	12
11	13

# Bases décimales et binaires

- On peut compter dans n'importe quelle base
- La valeur de 10 dépend de la base utilisée



base 10	base 8
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	10
9	11
10	12
11	13

# Bases décimales et binaires

- On peut compter dans n'importe quelle base
- La valeur de 10 dépend de la base utilisée

base 10	base 8	base 5
0	0	0
1	1	1
2	2	2
3	3	3
4	4	4
5	5	10
6	6	11
7	7	12
8	10	13
9	11	14
10	12	20
11	13	21

# Bases décimales et binaires

- On peut compter dans n'importe quelle base
- La valeur de 10 dépend de la base utilisée

base 10	base 8	base 5	base 2
0	0	0	0
1	1	1	1
2	2	2	10
3	3	3	11
4	4	4	100
5	5	10	101
6	6	11	110
7	7	12	111
8	10	13	1000
9	11	14	1001
10	12	20	1010
11	13	21	1011

# Bases décimales et binaires

- On peut compter dans n'importe quelle base
- La valeur de 10 dépend de la base utilisée

10 en base 10 = 10	en décimal
10 en base 8 = 8	en décimal
10 en base 5 = 5	en décimal
10 en base 2 = 2	en décimal

base 10	base 8	base 5	base 2
0	0	0	0
1	1	1	1
2	2	2	10
3	3	3	11
4	4	4	100
5	5	10	101
6	6	11	110
7	7	12	111
8	10	13	1000
9	11	14	1001
10	12	20	1010
11	13	21	1011

# Bases décimales et binaires

- On peut compter dans n'importe quelle base
- La valeur de 10 dépend de la base utilisée

10 en base 10 = 10 en décimal
10 en base 8 = 8 en décimal
10 en base 5 = 5 en décimal
10 en base 2 = 2 en décimal

Nombre binaire	1	10	100	1 000	10 000
Valeur décimale	1	2	4	8	16
Ou	$2^0$	$2^1$	$2^2$	$2^3$	$2^4$

base 10	base 8	base 5	base 2
0	0	0	0
1	1	1	1
2	2	2	10
3	3	3	11
4	4	4	100
5	5	10	101
6	6	11	110
7	7	12	111
8	10	13	1000
9	11	14	1001
10	12	20	1010
11	13	21	1011

# Le binaire pour les ordinateurs

- En mécanique, on peut fabriquer des composants qui ont 10 états différents
  - La première machine à calculer de Blaise Pascal vers 1642-45 utilise le système décimal



# Le binaire pour les ordinateurs

- En mécanique, on peut fabriquer des composants qui ont 10 états différents
  - La première machine à calculer de Blaise Pascal vers 1642-45 utilise le système décimal
- En électronique c'est plus facile de savoir si un circuit est ouvert ou fermé, allumé ou éteint, on ou off, 1 ou 0
  - Les ordinateurs utilisent le système binaire



# Le binaire pour les ordinateurs

- Mais ...  
dès 1679, Leibniz considérait que la numération binaire serait mieux adaptée aux machines à calculer



<https://archive.org/details/69LeibnizDiadica>

# Bases couramment utilisées

0 et 1 sont des digits binaires ou bits  
un octet (ou byte) comprend 8 bits : 0 à 255

- Décimal 0,1,2,3,4,5,6,7,8,9
- Binaire 0,1
- Hexadécimal 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

La base 16 - hexadécimale permet de représenter  
256 valeurs (0 à 255) sur 2 caractères (00 à FF).

*Il n'y a que 10 types de personnes dans le monde.*

*Ceux qui comprennent le binaire  
et ceux qui ne le comprennent pas!*

