
Représentation d'un entier en base b

13 octobre 2012

1 Prérequis.

Les bases de la programmation en langage Python sont supposées avoir été travaillées. L'écriture en base b d'un entier est ainsi défini à partir de l'algorithme des divisions en cascade et sa traduction en langage Python permet aux élèves d'auto-corriger leurs exercices de passage d'une base à l'autre.

2 Objectifs.

Thème de la représentation de l'information : introduire et travailler l'écriture en base b . Un objectif parallèle : travailler sur de courts algorithmes et programmes python.

Un mini-projet sera évalué pour la synthèse de ce thème : programmation (en langage python) du passage de la base shadock à la base dix et vice versa et de la base shadock à la base deux et vice-versa.

Résumé

Comprendre l'humour geek : « Dans le monde, il y a 10 sortes de personnes : ceux qui comprennent le binaire, et les autres ».

3 0 et 1

La mémoire d'un ordinateur est constituée d'une multitude de petits circuits électroniques. Chacun de ces circuits ne peut prendre que deux états. On associe traditionnellement l'un des états à 0 et l'autre à 1. De ce fait toute information doit être traduite dans un ordinateur uniquement par des 0 et des 1.

bit

bit est l'abréviation de BInary Digit (chiffre binaire).

octet

Un octet (en anglais : byte) est une suite de 8 bits.

mot

L'état d'un circuit, composé de plusieurs circuits mémoire-un-bit, se décrit par une suite finie de 0 et de 1 qu'on appelle mot.

Exercice 1

Si la mémoire d'un ordinateur était constituée de dix circuits à mémoire-un-bit (autrement dit : par des mots de 10 bits), quel serait le nombre d'états possibles de la mémoire de cet ordinateur ? Et avec 1 milliard de circuits ?

Exercice 2

On veut représenter les 7 couleurs de l'arc en ciel par un mot, les sept mots devant être distincts et de même longueur (en bits). Quelle est la longueur minimale de ces mots ?

Une résolution

Avec un bit, on peut enregistrer deux informations différentes : 0 ou 1. Avec deux bits, on forme 4 mots différents : 00, 01, 10, 11. Avec trois bits, on forme 2^3 mots différents : 000, 001, 010, 011, 100, 101, 110, 111.

Il faut donc au moins une longueur 3. Mais on doit aussi pouvoir coder la nature de l'information transmise pour que le logiciel utilisant cette information sache ici que les mots concernés représentent des couleurs...

4 Division euclidienne

A savoir

Pour tout couple d'entiers naturels $(a; b)$ avec $a \in \mathbb{N}$ et $b \in \mathbb{N} - \{0\}$, il existe un unique couple $(q; r)$ d'entiers tels $a = bq + r$ et $0 \leq r < b$. q est appelé quotient de la division euclidienne (on parle aussi de division entière) de a par b et r reste de cette division.

Exemple : $34 = 9 \times 3 + 7$, le reste de la division euclidienne de 34 par 9 est donc 7. Par contre le reste de la division euclidienne de 34 par 3 n'est pas 7 (car le reste dans une division euclidienne par 3 ne peut valoir que 0, 1 ou 2).

On rappelle qu'en python le reste de la division euclidienne de a (de type int) par b (de type int) est obtenu par l'instruction $a \% b$ et le quotient de a par b par $a // b$ ou $\text{int}(a / b)$.

5 Les chiffres en base b .

5.1 Les chiffres d'un entier écrit en base 10.

Exercice 3

On considère l'entier $n = 5489$.

1. On a entré une série d'instructions dans ipython. Compléter les sorties successives.

Python

```
1 In [1]: n=5489
2 In [2]: n%10
3 Out[2]:
4 In [3]: n=n//10
5 In [4]: n
6 Out[4]:
7 In [5]: n%10
8 Out[5]:
9 In [6]: n=n//10
10 In [7]: n
11 Out[7]:
12 In [8]: n%10
13 Out[8]:
14 In [9]: n=n//10
15 In [10]: n
16 Out[10]:
17 In [11]: n%10
18 Out[11]:
19 In [12]: n=n//10
20 In [13]: n
21 Out[13]:
```

2. En déduire un programme python tel que :

Entrée : un entier naturel n donné en écriture décimale.

Sortie : les chiffres de l'entier naturel n sous forme d'une liste, dans l'ordre usuel de lecture.

En d'autres termes, votre programme devra avoir le même effet que le suivant mais l'utilisation des fonctions `str()` et `list()` doit être remplacée par les opérations arithmétiques de division euclidienne par 10 :

Python

```
1 def chiffre (n):
2     """n est un entier naturel donné par son écriture décimale"""
3     return list (str(n))
4     #####
5 print chiffre (4587)
```

Une résolution

Le programme attendu :

Python

```
1 def Chiffre (n):
2     """n est un entier naturel donné par son écriture décimale"""
3     c=[]
4     while n!=0:
5         c.append(n%10)
6         n//=10
7     c.reverse ()
8     return c
9
10 #####
11 print Chiffre (4587)
```

Si la programmation récursive a été abordée, on peut aussi utiliser :

Python

```
1 def CHIFFRE(n,L=[]):
2     if n==0 :
3         L.reverse()
4         return L
5     else :
6         return CHIFFRE(n//10,L+[n%10])
7
8 #####
9 print CHIFFRE(4587)
```

□

5.2 Écriture binaire d'un entier

Écriture binaire d'un entier

L'écriture binaire d'un entier (ou écriture en base deux) est obtenue en remplaçant 10 par 2 dans l'algorithme du paragraphe précédent qui donnait les chiffres d'un entier en base 10.

En d'autres termes, vous obtiendrez l'écriture binaire de l'entier naturel n par l'algorithme suivant (appelé algorithme des divisions en cascade) :

Python

```
1 def ChiffreBinaire(n):
2     """n est un entier naturel donné par son écriture décimale"""
3     c=[]
4     while n!=0:
5         c.append(n%2)
6         n//=2
7     c.reverse()
8     return c
9
10 #####
11 print Chiffre(4587)
```

Savoir Faire

Vous devrez savoir passer de l'écriture décimale à l'écriture binaire d'un entier (sans machine).

Exercice 4

Donner, sans machine, l'écriture binaires des entiers ci-dessous. Vous vérifierez ensuite l'exactitude de votre réponse à l'aide de l'algorithme précédent ou à l'aide de la fonction python `bin()` qui prend en paramètre un entier naturel (donné par son écriture décimale) et retourne l'écriture binaire de l'entier (précédée de 0b pour signaler qu'il s'agit d'une écriture binaire). $N_1 = 15$, $N_2 = 32$, $N_3 = 7$, $N_4 = 8$.

Exercice 5

Justifier que tout entier naturel possède une écriture binaire, c'est à dire que pour tout entier naturel donné en entrée de l'algorithme des divisions en cascade avec la base deux, l'algorithme se terminera.

Une résolution

Lorsqu'on divise un entier non nul n par 2, on obtient un entier strictement plus petit. La suite des quotients dans l'algorithme serait donc une suite infinie d'entiers positifs distincts si n ne prenait jamais la valeur 0. Comme une suite d'entiers positifs distincts, inférieurs à un entier n donné, ne contient qu'au plus n termes, l'algorithme doit se terminer. \square

5.3 Écriture en base 8.

Écriture octale

L'écriture en base huit d'un entier est obtenue en remplaçant 10 par 8 dans l'algorithme de division en cascade.

En d'autres termes, vous obtiendrez l'écriture octale de l'entier naturel n par l'algorithme suivant :

Python

```
1 def ChiffreOctal(n):
2     """n est un entier naturel donné par son écriture décimale"""
3     c=[]
4     while n!=0:
5         c.append(n%8)
6         n//=8
7     c.reverse()
8     return c
9
10 #####
11 print ChiffreOctal(65)
```

Exercice 6

Donner (sans machine) l'écriture en base huit des entiers ci-dessous.

$N_1 = 8$, $N_2 = 11$, $N_3 = 16$, $N_4 = 67$, $N_5 = 158$, $N_6 = 8$.

Vous vérifierez l'exactitude de vos réponses à l'aide de l'algorithme en cascade exécuté par python ou à l'aide de la fonction `oct()` du langage python.

Savoir Faire

Vous devrez savoir passer de l'écriture décimale à l'écriture octale d'un entier (sans machine).

5.4 Écriture hexadécimale d'un entier

Définition

L'écriture hexadécimale (écriture en base 16) d'un entier naturel est obtenue par l'algorithme des divisions en cascade avec la base 16.

Python

```
1 def Chiffrehexa(n):
2     """n est un entier naturel donné par son écriture décimale"""
3     c=[]
4     while n!=0:
5         c.append(n%16)
6         n//=16
7     c.reverse()
8     return c
9     #####
10    print Chiffrehexa(47)
```

Exercice 7

Pour écrire un nombre en base 16, on utilise les 10 chiffres usuels et les lettres a, b, c, d, e, f. Pourquoi ?

Exercice 8

Modifier le programme précédent de façon à ce que 10 soit remplacé par la lettre A, 11 par la lettre B, 12 par la lettre C, 13 par la lettre D, 14 par la lettre E, 15 par la lettre F.

Une résolution

Par exemple :

Python

```
1 def Chiffrehexa(n):
2     """n est un entier naturel donné par son écriture décimale"""
3     ch=dict()
4     for i in range(10):ch[i]=i
5     for i in range(10,16):ch[i]=chr(i+55)
6     c=[]
7     while n!=0:
8         c.append(ch[n%16])
9         n//=16
10    c.reverse()
11    return c
12    #####
13    print Chiffrehexa(47)
```

Exercice 9

Donner, sans machine, l'écriture hexadécimale des entiers $N_1 = 17$, $N_2 = 155$, $N_3 = 16$. Vous vérifierez l'exactitude de vos calculs à l'aide du programme des divisions en cascade précédent ou à l'aide de la fonction hex() du langage python.

Savoir Faire

Vous devrez savoir passer de l'écriture décimale à l'écriture hexadécimale d'un entier (sans machine).

5.5 L'algorithme des divisions en cascade : les chiffres en base b

Pour écrire un entier en base b (où b est un entier au moins égal à 2), on utilise l'algorithme des divisions en cascade :

Python

```
1 def cascade(b,n):
2     """ b entier au moins égal à 2, donné en base 10
3     n entier naturel donné en base 10.
4     renvoie l'écriture en base b de n par divisions en cascade"""
5     c=[]
6     while n!=0:
7         c.append(n%b)
8         n//=b
9     c.reverse()
10    return c
11 #####
12 print cascade(16,47)
```

Toutefois, lorsque b est supérieur ou égal à 10, de nouveaux symboles devront être introduits pour les restes obtenus supérieurs ou égaux à 10.

6 Combinaisons linéaires des puissances de la base.

6.1 Cas de la base 10.

Considérons l'entier $N = 4567$.

Les chiffres de cette écriture décimale peuvent être obtenus par l'algorithme des divisions en cascade.

Appliquons le étape par étape.

- $N = 10 \times 456 + 7$.
- $456 = 10 \times 45 + 6$, d'où $N = 10 \times (10 \times 45 + 6) + 7$, soit $N = 10^2 \times 45 + 10 \times 6 + 7$.
- $45 = 10 \times 4 + 5$ d'où $N = 10^2 \times (10 \times 4 + 5) + 10 \times 6 + 7$ et $N = 10^3 \times 4 + 10^2 \times 5 + 10 \times 6 + 7$.

On retrouve ainsi les notions d'unité, dizaine, centaine, millier ...

6.2 Du binaire au décimal

En procédant de la même façon que ci-dessus, on obtient les notions d'unité, deuzaine, quatraine, huitaine... pour la base deux.

L'écriture $N = 1101_{\text{deux}}$ donne par exemple $N = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1$.

Exercice 10

Donner l'écriture en base dix des entiers suivants (sans machine) : $N_1 = 1101_{\text{deux}}$, $N_2 = 10100_{\text{deux}}$. Vous vérifierez la validité de vos réponses à l'aide de python en entrant dans un terminal : `int('1101',2)`, `int('10100',2)`.

Savoir Faire

Savoir passer du binaire au décimal.

6.3 De l'octal au décimal

De la même façon que précédemment, l'écriture 457_{huit} désigne en fait l'entier $4 \times 8^2 + 5 \times 8 + 7$.

Exercice 11

Donner, sans machine, les écritures décimales de $N_1 = 77_{\text{huit}}$ et $N_2 = 145_{\text{huit}}$. Vous vérifierez ensuite vos calculs à l'aide de `int('77',8)`, `int('145',8)`.

Savoir Faire

Savoir passer de l'octal au décimal.

6.4 De l'hexadécimal au décimal

Exercice 12 ✍

Donner, sans machine, les écritures décimales de $N_1 = AA_{\text{seize}}$ et $N_2 = B7A_{\text{seize}}$. Vous vérifierez ensuite vos calculs à l'aide de `int('aa',16)`, `int('b7a',16)`.

Savoir Faire

Savoir passer de l'hexadécimal au décimal.

7 Plus grand entier représentable en binaire sur n bits.

Somme de termes en progression géométrique

Soit q un réel distinct de 1. $1 + q + q^2 + \dots + q^{n-1} = \frac{q^n - 1}{q - 1}$.

En particulier $1 + 2 + 2^2 + 2^3 + \dots + 2^{n-1} = 2^n - 1$.

Preuve. On note $S = 1 + q + q^2 + \dots + q^{n-1}$. Alors $qS - S = q(1 + q + q^2 + \dots + q^{n-1}) - (1 + q + q^2 + \dots + q^{n-1})$, soit $S(q - 1) = q + q^2 + \dots + q^{n-1} + q^n - 1 - q - q^2 - \dots - q^{n-1}$ et $S(q - 1) = q^n - 1$. \square

Corollaire

Sur un mot de n bits, on peut représenter en binaire les entiers compris entre 0 et $2^n - 1$.

Exercice 13 ✍

1. Quels entiers naturels peut-on écrire en binaire sur un octet ?
2. Quels entiers naturels peut-on écrire en binaire sur un mot de 32 bits ?
3. Quels entiers naturels peut-on écrire en binaire sur un mot de 64 bits ?

Savoir Faire

Savoir donner le nombre d'entiers qu'il est possible de coder en binaire sur une longueur de n bits.

8 Lien entre le binaire et l'hexadécimal

8.1 Du binaire à l'hexadécimal

Comme $16 = 2^4$, le passage de l'écriture en base 2 à l'écriture en base 16 peut se faire de la façon suivante.

- On regroupe les bits de l'écriture en base deux par paquets de quatre en commençant par le chiffre des unités et en ajoutant des zéros à gauche si nécessaire. Par exemple, pour $N = 110110_{\text{deux}}$, on regroupe ainsi : $N = 0011\ 0110_{\text{deux}}$.
- On traduit chaque paquet de quatre en base seize. Avec l'exemple, $0110_{\text{deux}} = 6_{\text{seize}}$ et $0011_{\text{deux}} = 3_{\text{seize}}$.

– On a alors obtenu l'écriture hexadécimale de l'entier qui était donné en binaire : $N = 36_{\text{seize}}$.

Exercice 14

Les entiers suivants sont donnés sous leur forme binaire. Donner leur écriture hexadécimale (sans machine).
 $N_1 = 101010$, $N_2 = 111111111$. Vous vérifierez vos calculs en utilisant dans un terminal : `hex(int('101010',2))`.

8.2 De l'hexadécimal au binaire

Pour passer de la base 16 à la base 2, on traduit chaque chiffre de l'écriture hexadécimale en base deux, en octroyant systématiquement quatre bits à chacun des chiffres de l'écriture hexadécimale (quitte à ajouter des 0 sur la gauche).

Exemple avec $N = 215_{\text{seize}}$. $5_{\text{seize}} = 101_{\text{deux}} = 0101_{\text{deux}}$, $1_{\text{seize}} = 1_{\text{deux}} = 0001_{\text{deux}}$ et $2_{\text{seize}} = 10_{\text{deux}} = 0010_{\text{deux}}$.
L'écriture binaire de $N = 215_{\text{seize}}$ est donc 0010 0001 0101 (soit 1000010101).

Exercice 15

Les deux entiers suivants sont donnés sous leur forme hexadécimale : $N_1 = 1A5$, $N_2 = BAC$. Donner leur écriture binaire. Vérifier avec `bin(int('bac',16))`.

Savoir Faire

Savoir passer d'une écriture binaire à une écriture hexadécimale et vice-versa.

Il résulte des deux paragraphes précédents que la forme hexadécimale est une sorte de forme "compacte" de l'écriture binaire et par là, plus facile à lire pour l'être humain. C'est la raison pour laquelle elle est souvent utilisée en informatique.

9 Opérations en binaire.

Exercice 16

Si l'on dispose de l'écriture d'un entier n en base dix, il suffit d'ajouter un 0 à sa droite pour avoir l'écriture de l'entier $10n$.
On dispose de l'écriture d'un entier n en base b , on ajoute un 0 à la droite de cette écriture. A quelle opération cela correspond-il ?

Une résolution

Si $n = x_k b^k + x_{k-1} b^{k-1} + \dots + x_1 b + x_0$ alors $bn = x_k b^{k+1} + x_{k-1} b^k + \dots + x_1 b^2 + x_0 b + 0$. L'ajout d'un 0 à droite de l'écriture revient donc à multiplier par la base b .

En python, une opération de décalage des bits. Tester :

 Python

```
1 a=0b101
2 a=a<<1
3 bin(a)
```

□

Exercice 17

Pour ajouter des entiers écrits en binaire, on procède comme en base 10, en n'oubliant pas les retenues.

Il suffit de se rappeler le résultat des additions suivantes pour pouvoir poser toutes les additions : $0 + 0 = 0$, $0 + 1 = 1 + 0 = 1$, $1 + 1 = 10$, $1 + 1 + 1 = 11$. Poser les additions suivantes (comme à l'école élémentaire!), sans traduire les entiers dans la base dix.

$S_1 = 1011 + 111011$, $S_2 = 110111 + 111000$.

Vérifier avec une instruction telle que `bin(int('1011',2)+int('111011',2))`.

Savoir Faire

Savoir additionner deux entiers sous leurs formes binaires.

10 Mini-projet.