Première NSI
Chapitre IV - Les flottants

I. Les nombres binaires à virgule

On rappelle que le système décimal(base 10) est un système de position. Ainsi chaque chiffre qui compose un nombre a une certaine valeur.

Par exemple, dans le nombre $123,456$, le nombre 1 a une valeur de 100 et le nombre 5 a une valeur de $0,05$.
On peut écrire :

$$123,456 = 1 \times 100 + 2 \times 10 + 3 \times 1 + 4 \times 0,1 + 5 \times 0,01 + 6 \times 0,001$$

Soit :

$$123,456 = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 4 \times 10^{-1} + 5 \times 10^{-2} + 6 \times 10^{-3}$$

I.1. du système binaire vers le sytème décimal

Il en va de même en binaire (base 2).
Ainsi le nombre $110101,01101$ a pour valeur :
$110101,011_2 = 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3}$
$110101,011_2 = 32 + 16 + 0 + 4 + 0 + 1 + 0 + 0,25 + 0,125 = 53,375$

I.2. du sytème décimal vers le système binaire

Lorsqu'on veut convertir un nombre du système décimal vers le système binaire, on ne peut qu'avoir une approximation. Ce sera alors à nous de déterminer la précision que l'on veut obtenir. Les différentes étapes de la conversion sont :

- $1^{\text{ère}}$ étape : écrire la partie entière du nombre décimal en binaire (cf chapitre sur les nombres entiers)
- $2^{\text{ème}}$ étape : écrire la partie décimale en binaire (voir algorithme ci-dessous)
- $3^{\text{ème}}$ étape : assembler les deux.

Pour déterminer l'écriture décimale d'un nombre positif inférieur à 1, on exécute l'algorithme suivant :

- on multiplie le nombre par 2.
  - Si le résultat est supérieur ou égale à 1, on ajoute le chiffre 1 à la partie décimale de la réponse
  - sinon, on ajoute le chiffre 0 à la partie décimale de la réponse
- on garde la partie décimale du résultat.
- on réitère les deux premières étapes avec cette partie décimale.

Exemple

On veut écrire en bianire le nombre $12,348$ du système décimal.
$1^{\text{ère}}$ étape :
$12_{10} = 1100_2$
$2^{\text{ème}}$ et $3^{\text{ème}}$ étapes :
$0,348 \times 2 = 0,696$ donc $12,348_{10} = 1100,0$ avec une précision de 1 chiffre après la virgule.
$0,696 \times 2 = 1,392$ donc $12,348_{10} = 1100,01$ avec une précision de 2 chiffre après la virgule.
$0,392 \times 2 = 0,784$ donc $12,348_{10} = 1100,010$ avec une précision de 3 chiffre après la virgule.
$0,784 \times 2 = 1,568$ donc $12,348_{10} = 1100,0101$ avec une précision de 4 chiffre après la virgule.
$0,568 \times 2 = 1,136$ donc $12,348_{10} = 1100,01011$ avec une précision de 5 chiffre après la virgule.
$\vdots$

II. Représentation approximative des nombres réels

II.1. Généralités

On utilise le format IEEE-754. Il existe plusieurs formats binaires à virgule flottante : à 32 bits (simple précision) ou 64 bits (double précision). (4 ou 8 octets).

Format	signe	exposant	mantisse
32 bits	1 bit	8 bits	23 bits
64 bits	1 bit	11 bits	52 bits

Remarque

- le signe est représenté sur le bit de poids fort.
- *exposant* est la représentation binaire d'un entier signé (sur 8 ou 11 bits)
- Les formats à 32 ou 64 bits ignorent la partie entière de la mantisse.

II.2. De la représentation binaire vers l'écriture décimale

Un nombre réel en virgule flottante est le nombre :

$$(-1)^s \times 2^{exposant-biais} \times 1, mantisse$$

Remarque

- *exposant* est à convertir dans le système décimal.
- *biais* est égal à $2^{n-1} - 1$ où n est le nombre de bits sur lequel est codé l'*exposant*.
- Multiplier par $2^{exposant}$ revient à décaler la virgule de $1, mantisse$ de *exposant* chiffres vers la droite si *exposant* est positif ou vers la gauche si *exposant* est négatif.

Exemple

Soit le nombre réel suivant en format IEEE -754 simple précision écrit en binaire : 1100 0010 1111 0100 1010 0110 0110 0110

Ecrivons le différemment : 1 10000101 11101001010011001100110

- Le signe : $(-1)^1 = -1$
- L'exposant : $10000101 = 133$
- Le biais : $2^{8-1} - 1 = 127$
- $1, mantisse$: $1, 11101001010011001100110$

Le calcul :
$-1 \times 2^{133-127} \times 1, 11101001010011001100110_2$
$= -1 \times 2^6 \times 1, 11101001010011001100110_2$
$= -1 \times 1111010, 01010011001100110_2$
$= -1 \times (2^6 + 2^5 + 2^4 + 2^3 + 2^1 + 2^{-2} + 2^{-4} + 2^{-7} + 2^{-8} + 2^{-11} + 2^{-12} + 2^{-15} + 2^{-16})$
$= -122.3249969482421875$

II.3. De l'écriture décimale vers la représentation binaire

Les différentes étapes pour trouver la représentation binaire d'un nombre dans la norme IEEE-754 sont :

- 1^{re} étape : déterminer le signe.
- 2^{me} étape : écrire la valeur absolue du nombre décimal en binaire.
- 3^{me} étape : trouver le décalage et la mantisse.
- 4^{me} étape : déterminer le biais et le codage de l'exposant.
- 5^{me} étape : donner la valeur en virgule flottante.

Exemple

Soit le nombre réel 12,531. Donner sa représentation binaire en format IEEE-754 simple précision puis sa représentation binaire.
Les étapes dans l'ordre :

- 1^{re} étape
  Le signe : positif donc 0
- 2^{me} étape
  $12 = 1100_2$
  $0,53110 = 0,10000111111011111001_2$
  Donc $12,531 = 1100,10000111111011111001_2 = 1,10010000111111011111001 \times 2^3$
- 3^{me} étape
  La mantisse vaut 10010000111111011111001 et $decalage = 3$
- 4^{me} étape
  $biais = 2^{8-1} - 1 = 127$
  Donc $exposant = 3 + biais = 3 + 127 = 130 = 10000010_2$
- 5^{me} étape
  Ainsi $12,531$ est codé sur 32 bits par 0 10000010 10010000111111011111001
  soit 0100 0001 0100 1000 0111 1110 1111 1001.

III. Exercices

III.1. Exercice 1

1. Déterminer la valeur du nombre binaire $1100110,01101$.
2. Déterminer l'écriture binaire du nombre $57,853$ avec une précision de 7 chiffres après la virgule.

III.2. Exercice 2

Donner l'écriture décimale des nombres dont la représentation binaire dans la norme IEEE-754 est :

- 1011 1110 1111 0001 1001 1001 1000 0100
- 0100 0000 1111 0000 0000 0000 0000 0000

III.3. Exercice 3

Donner la représentation binaire des nombres suivants au format IEEE-754 simple précision :

- $1,625$
- 47
- $\frac{1}{3}$

III.4. Exercice 4

- Créer une fonction `repr_bin(nb: float) -> str` qui renvoie la représentation binaire sur 32 bits dans la norme IEEE-754 d'un nombre `nb` placé en paramètre.
- Créer une fonction `ecriture_dec(chaine: str) -> float` qui renvoie l'écriture décimale dont la représentation binaire dans la norme IEEE-754 simple précision est `chaine`.