

REPRÉSENTATION DES NOMBRES RÉELS - EXERCICES

Exercice 1 : codage en virgule fixe

- Encoder en base 2 le nombre $(4,125)_{10}$.
- Calculer la représentation décimale de $(100,0101)_2$.
- Encoder maintenant le nombre $(0,1)_{10}$ en base 2. Que remarquez vous ?
- Créer une fonction **partieDecimaleEnBinaire**(*valeur,bits*) prend en argument un nombre *valeur* (en base 10) correspondant à la partie décimale d'un nombre et un nombre *bits* correspondant au nombre de bits sur lesquels on donnera la partie décimale. La fonction retournera suite des nombres de sa partie décimale entre 0 et 1 en binaire sur le nombre de bits demandé en argument soit sous forme d'une chaîne de caractère, soit sous forme d'un tableau.
- En utilisant la fonction de conversion des entiers positifs de la base 10 vers la base 2 de la feuille d'exercices du chapitre 1 (exercice 7 ; vous pourrez la recopier ou l'importer...rappelez vous!!), créer un programme Python avec une troisième fonction **reelPositifEnBinaire**(*valeur,bits*) qui retourne le nombre en base 2 sous la forme d'une chaîne de caractère avec une virgule ou sous la forme d'une liste de deux listes contenant la valeur en binaire (partie entière puis décimale). Vérifiez alors vos résultats !

Exercice 2 : codage en virgule flottante - norme IEEE-754 -simple précision

- Convertir en base 10 les nombres ci-dessous codés selon la norme IEEE-754 en simple précision :
 - 1 0100 0110 1001 0000 0000 0000 0000 000
 - 0 0111 1100 0100 0000 0000 0000 0000 000
- Convertir les nombres suivants en base 2 selon la norme IEEE-754 en simple précision :
 - $(10,59375)_{10}$
 - $(-32,75)_{10}$

Exercice 3

On tape en Python l'expression `>>>(1e25+16) - 1e25`

Quel est le résultat attendu ? Quel est le résultat obtenu ? Pourquoi ?

REPRÉSENTATION DES NOMBRES RÉELS - EXERCICES

Exercice 1 : codage en virgule fixe

- Encoder en base 2 le nombre $(4,125)_{10}$.
- Calculer la représentation décimale de $(100,0101)_2$.
- Encoder maintenant le nombre $(0,1)_{10}$ en base 2. Que remarquez vous ?
- Créer une fonction **partieDecimaleEnBinaire**(*valeur,bits*) prend en argument un nombre *valeur* (en base 10) correspondant à la partie décimale d'un nombre et un nombre *bits* correspondant au nombre de bits sur lesquels on donnera la partie décimale. La fonction retournera suite des nombres de sa partie décimale entre 0 et 1 en binaire sur le nombre de bits demandé en argument soit sous forme d'une chaîne de caractère, soit sous forme d'un tableau.
- En utilisant la fonction de conversion des entiers positifs de la base 10 vers la base 2 de la feuille d'exercices du chapitre 1 (exercice 7 ; vous pourrez la recopier ou l'importer...rappelez vous!!), créer un programme Python avec une troisième fonction **reelPositifEnBinaire**(*valeur,bits*) qui retourne le nombre en base 2 sous la forme d'une chaîne de caractère avec une virgule ou sous la forme d'une liste de deux listes contenant la valeur en binaire (partie entière puis décimale). Vérifiez alors vos résultats !

Exercice 2 : codage en virgule flottante - norme IEEE-754 -simple précision

- Convertir en base 10 les nombres ci-dessous codés selon la norme IEEE-754 en simple précision :
 - 1 0100 0110 1001 0000 0000 0000 0000 000
 - 0 0111 1100 0100 0000 0000 0000 0000 000
- Convertir les nombres suivants en base 2 selon la norme IEEE-754 en simple précision :
 - $(10,59375)_{10}$
 - $(-32,75)_{10}$

Exercice 3

On tape en Python l'expression `>>>(1e25+16) - 1e25`

Quel est le résultat attendu ? Quel est le résultat obtenu ? Pourquoi ?