Travaux Pratiques d'algorithmique n°2

Cours d'Informatique de Deuxième Année

—DEUG Sciences S3-MI—

Codage IEEE

L'objet de cette séance est l'étude et l'implémentation du standard IEEE (ici sur 32 bits) pour le codage des nombres réels.
Toutes les fonctions programmées devront être commentées.

▸ **Exercice 1.** ***(Rappels)*** *Quelles sont les valeurs possibles pour un bit ? Combien de valeurs différentes peut prendre un nombre codé sur un octet (8 bits) ? En C, sur combien d'octets sont codés les entiers (*`int` *et* `long int`*), les réels (*`float`*,* `double` *et* `long double`*) ? Consulter le fichier* `/usr/include/float.h`*. En déduire comment tester l'égalité de deux réels ?*

▸ **Exercice 2.** *Rappelons tout d'abord qu'un réel se décompose dans le standard IEEE (simple précision, i.e. 32 bits) sous la forme suivante :*

$$X = (-1)^S \times 2^{E-127} \times (1 + M)$$

où S est le bit de signe ;
E est l'exposant (codé sur 8 bits) ;
M est la mantisse (codée sur 23 bits).

Le but de cet exercice est de manipuler la ***mantisse****, c'est-à-dire de représenter un réel appartenant à $[0, 1[$. Cette mantisse sera stockée dans un tableau de 23 "bits".*

- *Rappeler sur le papier le détail du calcul pour 0,8 ;*
- *écrire la fonction de prototype* `float Mantisse2Reel(int *mantisse)` *qui retourne le réel codé par la mantisse passée en argument ;*
- *écrire la fonction de prototype* `int Reel2Mantisse(float reel, int *mantisse)` *qui code la mantisse du nombre réel donné en argument. On rappelle que ce nombre doit nécessairement appartenir à $[0, 1[$.*

▸ **Exercice 3.** ***(IEEE)*** *La représentation IEEE d'un réel est donnée par : un "bit" pour le signe, un tableau de 8 "bits" pour l'exposant et enfin un tableau de 23 "bits" pour la mantisse (voir exercice précédent).*

Écrire la fonction qui renvoie le réel associé à une représentation IEEE. Écrire également la fonction qui code dans le standard IEEE un nombre réel quelconque.