

## T.P. 2

# Conversion de données

Ce TP a pour but de :

1. convertir des entiers naturels en différentes bases
2. afficher le code ASCII d'un caractère

## 1 Conversion de base

**Exercice 1.** L'objectif de cet exercice est d'écrire un programme qui lit un entier positif ou nul, puis l'affiche en base binaire.

Ecrivez un programme qui fait ce qui suit :

1. demande à l'utilisateur de saisir un entier naturel  $n$  (utiliser la fonction *scanf*)
2. convertit l'entier naturel  $n$  saisi par l'utilisateur en base binaire (écrivez une fonction *int binaire(int n)* qui prend l'entier  $n$  en argument et retourne sa valeur en base binaire)
3. affiche la valeur binaire de l'entier saisi par l'utilisateur tout en expliquant les différentes étapes de la conversion (utiliser des commentaires que vous affichez à l'écran au fur et à mesure de la conversion en se servant de la fonction *printf*)

Exemple de ce que vous devez avoir à l'écran :

Veillez saisir l'entier à convertir :

9

```
**** Début de la fonction binaire(9) ****
```

Les principales étapes de la conversion sont :

Etape 1. Division euclidienne de 9 par 2 :  $9 = 4 * 2 + 1$

Etape 2. Division euclidienne de 4 par 2 :  $4 = 2 * 2 + 0$

Etape 3. Division euclidienne de 2 par 2 :  $2 = 1 * 2 + 0$

Etape 4. Division euclidienne de 1 par 2 :  $1 = 0 * 2 + 1$

```
**** Fin de la fonction binaire(9) ****
```

La valeur binaire équivalente à cet entier est :

1001

**Exercice 2.** A l'image de l'exercice précédent, convertissez un entier positif, saisi par l'utilisateur, de la base décimale à la base hexadécimale

## 2 Affichage du code ASCII d'un caractère

**Exercice 3.** Ecrivez un programme qui lit un caractère saisi par l'utilisateur et qui affiche son code ASCII.

**Exercice 4.** Utilisez l'exercice 3. pour écrire un programme qui convertit les lettres majuscules (respectivement minuscules) en lettres minuscules (respectivement majuscules), puis afficher la table des caractères ASCII.