

## L2 Programmation Impérative

### TP5

## 1 Structures

Nous voulons définir une structure composé *Joueur* qui comprend 3 champs : *nom* est une chaîne de caractères ; *nbmj* (nombre de matchs déjà joués), *nbmrj* (nombre de matchs qui restent à jouer) sont des entiers.

1. Définissez la structure *Joueur*.
2. Ecrivez une fonction **Saisie** qui demande à l'utilisateur de saisir le nombre de joueurs et les champs *nom*, *nbmj* concernant ces joueurs.
3. Ecrivez une fonction **Affiche** pour afficher les données dans un tableau de joueurs.
4. Ecrivez une fonction **Calcul** pour calculer le nombre de matchs à jouer (le champ *nbmrj*) à partir du nombre de matchs déjà joués (le champ *nbmj*) et le nombre de matchs que chaque joueur doit jouer au total (ce nombre *nbm* peut être passé en paramètre de la fonction). Le calcul à effectuer pour chaque joueur est  $nbmrj = nbm - nbmj$ .
5. Soit *TabJ* un tableau de joueurs. Ecrivez le programme principal qui fait appel aux fonctions dans l'ordre **Saisie**, **Calcul** et **Affiche**.

## 2 Liste

On considère la définition suivante :

```
typedef struct maillon
{
    int donnee;
    struct maillon * suivant;
} maillon;
```

En utilisant cette structure, définissez une liste chaînée d'entiers (en ordre croissant dans cet exercice), n'oubliez pas de définir la fin de la liste. Ecrivez les fonction **Saisie** et **Affiche** respectivement pour saisir et afficher une telle liste. Finalement, écrivez une fonction **supprime** :

- d'en-tête : *maillon \* supprime (maillon \* deb, int cle)*
- pour laquelle :
  - deb* est l'adresse du début d'une liste chaînée de maillon(s) qui contient des entiers tous distincts classés par ordre croissant
  - cle* est un entier
- qui supprime de la liste chaînée la valeur *cle* si elle existe et qui retourne le début de la liste ainsi obtenue.