Université de Marne la Vallée
Licence EEA
Année 2001-2002

Langage C

TD N° 5 : Structurations

Exercice 1
Ecrire une fonction récursive (c'est à dire qui s'appelle elle-même) de façon à calculer le n-ème terme de la suite définie par :

$$\begin{cases} u_n = u_{n-1}^2 + 1 & \text{si} \quad n > 0 \\ u_0 = 1. \end{cases}$$

(Cette suite donne le nombre d'arbres binaires possibles ayant une profondeur au plus égale à n).

Exercice 2
Ecrire une fonction
`void mult (float a[ ], float b[ ], float c[ ], int n)`
effectuant le produit terme à terme des composantes des vecteurs a et b de dimension n. Les n valeurs résultantes seront stockées dans le vecteur c.

Exercice 3
On définit une fiche signalétique par un nom et un surnom regroupés au sein d'une structure :

```
struct fiche {
        char nom[80], surnom[80];}
```

1. Ecrire une fonction
   `void saisie(struct fiche *pf)`
   qui effectue la saisie de la fiche pointée par pf.
2. Ecrire une fonction
   `void affiche(struct fiche f)`
   qui affiche le contenu de la fiche f.

Exercice 4
Le tri à bulle est un algorithme de tri très simple. Pour trier un tableau tab de n valeurs par ordre croissant, l'algorithme procède de la façon suivante. On prend le premier élément du tableau tab[0] et on le compare à tous les autres éléments du tableau. Si tab[0] > tab[i], on échange les 2 élements sinon on ne fait rien. A la fin de cette première passe, le plus petit élément se trouve dans tab[0]. On répète alors la procédure en partant de l'élément tab[1], puis de tab[2] et ainsi de suite jusqu'à tab[n-1]. Ecrire une fonction
`void bulle(float *tab,int n)`
réalisant le tri à bulle.