



Licence d'informatique
Module de Pratique du C

Travaux dirigés

Manipulation de structures

Philippe MARQUET

Octobre 2004

Les exercices proposés illustrent l'utilisation simple de structures. Une structure, ou enregistrement, est un agrégat de données. Une structure définit un nouveau type de donnée dont les éléments, ou champs, sont éventuellement hétérogènes (de types différents). Ces champs sont accessibles par leur nom.

On traite aussi d'unions et de types énumérés.

Nous reviendrons sur des utilisations plus avancées des structures et de leur représentation en mémoire dans d'autres TD.

Exercice 1 (Jouons aux cartes)

On désire gérer des cartes à jouer.

- On joue au bridge avec un jeu de 4 (trèfle, carreau, coeur, pique) \times 13 (as à 10 + valet, dame, roi) = 52 cartes.
- Au tarot il faut aussi considérer un cavalier dans chaque couleur, les atouts (de 1 à 21), et l'excuse.

Question 1.1 Proposez des structures de données pour représenter une carte d'un jeu de bridge, puis une carte d'un jeu de tarot.

Question 1.2 Au tarot, on identifie le 1 et le 21 d'atout ainsi que l'excuse comme des bouts. Proposez une fonction `est_un_bout` qui retourne vrai si et seulement si la valeur d'une carte passée en paramètre est un bout.

La belote se joue à quatre joueurs avec un jeu de 32 cartes (on exclut les cartes 2 à 6). Une des couleurs est désignée comme atout. L'ordre des cartes est le suivant :

- pour l'atout : valet, 9, as, 10, roi, dame, 8, 7 ;
- pour les autres couleurs : as, 10, roi, dame, valet, 9, 8, 7.

À chaque tour de jeu, un premier joueur joue la carte de son choix. Cette carte indique la couleur demandée. Chacun des autres joueurs joue une carte. La levée est emportée par le plus gros atout, ou, s'il n'y en a pas, pas la plus forte carte de la couleur demandée.

Question 1.3 Proposez une fonction qui retourne la carte emportant la levée identifiée par les paramètres : quatre cartes, l'atout, la couleur demandée.

Question 1.4 Proposez la définition d'un type de données pour mémoriser les cartes gagnées par un joueur lors des différentes levées de la partie.

Exercice 2 (Grands entiers positifs, inspiré d'un énoncé de Christian QUEINNEC, LIP6)

On propose de manipuler de grands entiers positifs. Un grand entier positif est représenté par un vecteur de `GEP_SIZE` chiffres qui sont des `unsigned long`.

Question 2.1 Proposez la définition d'un type de données pour mémoriser un grand entier positif.

Question 2.2 Donnez le prototype d'une fonction qui retourne la somme de deux GEP et l'éventuelle retenue.

Question 2.3 Proposez une définition de cette fonction.

Exercice 3 (Manipulation de formules)

On désire manipuler des formules caractérisées par :

- le résultat d'une formule est une valeur flottante ;
- une formule peut être un terme ;
- une formule peut être la somme de deux termes ;
- une formule peut être le produit (ou le quotient) d'un terme et d'un coefficient entier ;
- un terme est soit une valeur flottante immédiate, soit un nom de variable ;
- une variable est un identificateur auquel est associée une valeur flottante.

Ainsi les formules suivantes sont valides :

5.0 a $a + 4.0$ $a + b$ $b \times 3$

On propose les définitions et déclarations suivantes pour les variables et les termes :

```
/* Les variables */
#define NAME_LG      16          /* longueur d'un id de variable */
#define NVARs        32          /* nombre de variables */

struct var_s {
    char   v_name[NAME_LG];      /* nom de la variable */
    float  v_val;                /* valeur de cette variable */
};

static struct var_s vars[NVARs]; /* "table des symboles" */
static unsigned int nvars=0;

/* Les termes */
enum term_type_e {VAL_T, ID_T}; /* deux types de termes */

struct term_s {
    enum term_type_e    t_type;
    union {
        float t_val;
        char   t_var[NAME_LG];
    }
    t_term;
};
```

Question 3.1 Proposez une fonction `value_of_term()` qui retourne la valeur d'un terme.

Question 3.2 Proposez une définition de type pour représenter une formule.

Question 3.3 Proposez une fonction `value_of_formula()` qui retourne la valeur de l'évaluation d'une formule passée en paramètre.