Stage d'algorithmique n°3

Cours d'Informatique de Deuxième Année

—DEUG Sciences M3-MI—

Adresses, pointeurs, appels par adresse et tableaux

De même qu'il est possible de modifier la valeur d'un paramètre en *Pascal* en utilisant le mot-clef `VAR`, une telle opération est souvent nécessaire dans les programmes *C*. La nuance terminologique entre *C* et *Pascal* est que *C* travaille directement (au niveau syntaxique), non pas sur la valeur de la variable, mais sur le contenu de la cellule mémoire réservée pour cette variable. En fait, on parle plutôt de l'*adresse de la variable* pour désigner cet emplacement en mémoire. Une variable qui contient elle-même une adresse sera appelée un *pointeur*.
Nous allons voir ensuite que le mot *tableau* a, en *C*, une signification bien particulière, proche de la notion de pointeur...
Cette séance est donc consacrée à :

- l'accès et la modification d'une cellule en mémoire ;
- l'introduction de la notion de pointeur ;
- l'introduction de la notion de tableau ;
- la réalisation d'un programme de tri d'un tableau d'entiers par un algorithme simple : le tri par recherche du minimum.

▸ **Exercice 1.** *Soit le programme C suivant :*

```
#include <stdio.h>
void main(void)
{ int a, b, *c;
  a=2; b=a; c=&a;
  printf("%d %d %d\n",a,b,*c);
  a=3;
  printf("%d %d %d\n",a,b,*c);
}
```

Commentez l'affichage du programme et rajoutez une instruction qui affiche l'adresse de `a` *et la valeur de* `c`*.*

▸ **Exercice 2. (réalisation d'un swap de deux variables)** *Vérifions nos connaissances en matière de passage de paramètres à une fonction, en commentant le résultat de l'exécution du programme suivant :*

```c
#include <stdio.h>
void swap(int a, int b)
{ int tmp=a;
  a=b; b=tmp;
  printf("Dans la fonction : a = %d et b = %d\n", a, b);
}

void main(void)
{ int a=5,b=3;
  printf("Avant de rentrer dans la fonction  : a = %d et b = %d\n", a, b);
  swap(a, b);
  printf("En sortant de la fonction : a = %d et b = %d\n", a, b);
}
```

Comment modifier ce programme afin qu'il réalise vraiment l'échange des valeurs des deux variables ?

▸ **Exercice 3.** *Définissez une constante `MAXTAB` au moyen de la macro `#define`, puis déclarez dans un programme un tableau de `MAXTAB` entiers que l'utilisateur de votre programme devra remplir au clavier (fonction `LireTab`). Le programme affichera enfin le tableau au moyen d'une fonction `AffTab`. Modifiez ensuite votre programme pour qu'il traite des tableaux de taille variable.*

▸ **Exercice 4. (tri par recherche du minimum)** *On se propose dans cet exercice de réaliser une fonction de tri en ordre croissant d'un tableau d'entiers. On divise le problème en plusieurs étapes.*

1. *Ecrire une fonction de nom `MinTab` prenant en paramètres un tableau d'entiers ainsi que sa taille, et retournant la position du minimum des valeurs du tableau ;*
2. *Maintenant que l'on sait trouver la position du minimum des n valeurs du tableau, on peut donner l'algorithme de tri par recherche du minimum. En effet, le minimum occupera la première case du tableau trié. Ayant amené la dite valeur à sa place, il restera à trier la fin du tableau, c'est-à-dire $n-1$ valeurs. Notre algorithme s'arrêtera dès lors qu'il n'y aura plus qu'une valeur à trier.*
   (a) *donner en pseudo-langage, l'algorithme de tri par recherche du minimum d'un tableau de n éléments ;*
   (b) *donner une fonction de nom `TriMin` implémentant cet algorithme.*