

PROG2 - Programmation impérative

TP 03 – Les procédures et les fonctions

Julien Sopena

Février 2008

Exercice 1 : Sous-programmes

Question 1

Dans cet exercice, on veut utiliser un tableau pour stoker jusqu'à 100 entiers positifs. Quel que soit leur nombre, ils seront toujours suivis d'un entier négatif qui jouera le rôle de "bouchon". Donner le type `t_tab` correspondant à un tel tableau.

```
const
  TAILLE_TAB = 100;
type
  T_TAB = array [1..TAILLE_TAB+1] of integer ;
```

Question 2

Écrire un sous-programme `saisir` qui permette de saisir au clavier les différentes valeurs d'un tableau `t_tab`. La saisie s'arrête si le tableau est plein ou si l'utilisateur entre un nombre négatif.

```
procedure saisir(var tab : T_TAB) : integer ;
var
  i : word ;
begin
  i := 0 ;
  repeat
    i := i+1 ;
    write ('entrez un nombre : ');
    readln (tab[i]);
  until (tab[i] < 0) or (i = TAILLE_TAB) ;
  if (tab[i] >= 0) then
    tab[i+1] := -1;
end ;
```

Question 3

Écrire un sous-programme **taille** qui retourne le nombre d'entier contenu dans un tableau `t_tab`.

```
function taille(tab : T_TAB) : integer ;
var
  t : integer ;
begin
  t := 1 ;
  while (tab[t] >= 0) do
    inc(t) ;
  taille := t - 1 ;
end ;
```

Question 4

Écrire un sous-programme **max** qui retourne le plus grand des entiers contenus dans un tableau `t_tab`.

```
function max(tab : T_TAB) : integer ;
var
  i,m : word ;
begin
  m := tab[1];
  i := 1 ;
  while (tab[i]>=0) do
    begin
      if m < tab[i] then
        m := tab[i];
      inc(i);
    end ;
  max := m;
end ;
```

Question 5

Écrire un sous-programme **affiche** qui affiche toute les valeurs contenues dans un tableau `t_tab`. Le format de sortie sera le suivant : $[a_1, a_2, \dots, a_n]$. Faites attention aux virgules.

```
procedure affiche(tab : T_TAB) ;
var
  i : integer ;
begin
  write ('[');
  if tab[1] >= 0 then
    begin
      write (tab[1]);
      i := 2 ;
      while (tab[i] >= 0) do
        begin
          write (', ', tab[i]);
          i := i+1 ;
        end;
    end;
```

```

end ;
writeln (']');
end ;

```

Question 6

Écrire un sous-programme **échange** qui inverse deux éléments d'un tableau `t_tab`.

```

procedure échange(var tab : T_TAB ; i,j :word) ;
var
    tmp : integer ;
begin
    tmp := tab[i] ;
    tab[i] := tab[j] ;
    tab[j] := tmp ;
end ;

```

Question 7

Écrire un sous-programme **tri** qui utilise un tri à bulle pour ordonner toutes les valeurs entières contenues dans un tableau `t_tab`. On rappelle que le tri à bulle consiste à faire remonter le plus grand élément du tableau (comme une bulle d'air remonte à la surface) en comparant (et permutant si nécessaire) les éléments successifs du tableau. On réitère alors l'opération sur un tableau plus petit, jusqu'à obtenir un tableau trié (i.e., il n'y a pas eu besoin de permutation).

```

procedure tri(var tab : T_TAB) ;
var i,j,n : Integer ;
    enDesordre : Boolean ;
begin
    n := taille(tab) ;
    i :=1 ;
    enDesordre :=True ;
    while (i<n) and (enDesordre) do
    begin
        enDesordre :=False ;
        For j := n downto i+1 do
            if tab[j] < tab[j-1] then
                begin
                    échange(tab,j,j-1) ;
                    enDesordre :=True ;
                end ;
            inc(i) ;
        end ;
    end ;
end ;

```

Question 8

Écrire un sous-programme **moyennage** qui lisse les valeurs entières contenues dans un tableau `t_tab`. Cette opération de moyennage se fait en remplaçant le terme d'indice `j` par la moyenne de trois termes : la valeur de l'élément d'indice `j` du tableau et les valeurs des termes immédiatement avant et après dans le

tableau ($j - 1$ et $j + 1$). Les moyennes se calculent à partir des valeurs initiales du tableau. La première et la dernière valeur du tableau ne sont pas modifiées.

```

procedure moyennage(var tab : T_TAB) ;
var
  oldPred,tmp,i : integer ;
begin
  if (tab[1] >= 0) and (tab[2] >= 0) then
  begin
    oldPred := tab[1];
    i := 2 ;
    while (tab[i+1] >= 0) do
    begin
      tmp := tab[i];
      tab[i] := (oldPred + tab[i] + tab[i+1]) div 3;
      oldPred := tmp;
      inc(i);
    end;
  end;
end;

```

Question 9

Écrire un programme qui teste les sous-programmes et les déclarations précédents.

```

var
  monTab : T_TAB ;
  longueur : integer ; begin
  saisir(monTab);
  write('Tableau saisi : '); affiche(monTab);
  writeln ('Taille du tableau : ',taille(monTab));
  writeln ('Max dans le tableau : ',max(monTab));
  tri(monTab);
  write('Après tri : '); affiche(monTab);
  moyennage(monTab);
  write('Après moyennage : '); affiche(monTab);
  readln;
end.

```