

PROG2 - Programmation impérative

TP 11 – Les fichiers de caractères (suite)

Julien Sopena

Février 2008

Exercice 1 : Sous-programmes

Question 1

Dans cet exercice, on considère des fichiers de caractères dont chaque ligne comporte un nom. Définissez un type `t_liste` qui permette de stocker en mémoire une de ces listes de noms.

```
const
  TAILLE_MAX = 100 ;
type
  t_indice = 1..TAILLE_MAX ;
  t_taille = 0..TAILLE_MAX ;
  t_tab = array[t_indice] of string ;
  t_listeNoms = record
    taille : t_taille ;
    tab : t_tab ;
  end ;
```

Question 2

Écrire un sous programme `afficherFichier` qui affiche à l'écran la liste des noms contenus dans un fichier donc le nom sera passé en paramètre.

```
procedure afficherFichier (nomFichier : string);
var
  f : text;
  s : string;
begin
  assign(f, nomFichier) ;
  reset(f);
  while not eof(f) do
```

```

begin
  readln(f,s);
  writeln(s);
end;
close(f);
end;

```

Question 3

Écrire un sous-programme **lireFichier** qui enregistre dans une variable de type **t_liste** la liste des noms contenus dans un fichier donc le nom sera passé en paramètre.

```

procedure lireFichier (nomFichier : string;var l : t_listeNoms);
var
  f : text;
  s : string;
begin
  assign(f, nomFichier) ;
  reset(f);
  l.taille := 0;
  while not eof(f) do
  begin
    readln(f,s);
    inc(l.taille);
    l.tab[l.taille] :=s;
  end;
  close(f);
end;

```

Question 4

Écrire un sous-programme **ecrireFichier** qui enregistre une liste de noms de type **t_liste** dans un fichier dont le nom est passé en argument.

```

procedure ecrireFichier (fichier : string;liste : t_listeNoms);
var
  f : text;
  i : t_indice;
begin
  assign(f, fichier) ;
  rewrite(f);
  for i := 1 to liste.taille do
    writeln(f,liste.tab[i]);
  close(f);
end;

```

Question 5

Écrire un sous-programme **nouveauFichier** qui permet d'enregistrer une liste de noms dans un fichier dont le nom est passé en argument. Ces noms seront saisis par l'utilisateur, jusqu'à ce qu'il entre une chaîne vide. Le fichier sera écrasé s'il existe déjà.

```
procedure nouveauFichier (fichier : string);
var
  f : text;
  s : string;
begin
  assign(f, fichier) ;
  rewrite(f);
  write('Entrer un nom : ');
  readln(s);
  while length(s)>0 do
  begin
    writeln(f,s);
    write('Entrer un nom : ');
    readln(s);
  end;
  close(f);
end;
```

Question 6

Écrire un sous-programme **ajouterNom** qui ajoute un nom saisi au clavier à la fin d'un fichier dont le nom est passé en argument.

```
procedure ajouterNom(fichier : string);
var
  f : text;
  s : string;
begin
  assign(f, fichier) ;
  append(f);
  repeat
    write('Entrer un nom : ');
    readln(s);
  until length(s)>0 ;
  writeln(f,s);
  close(f);
end;
```

Question 7

Écrire un sous-programme **permuterNom** qui permute deux noms dans une liste de type **t_liste**.

```
procedure permuterNom (var tab : t_tab; i,j : t_indice) ;
var
  tmp : string ;
begin
  tmp := tab[i] ;
  tab[i] := tab[j] ;
  tab[j] := tmp ;
end ;
```

Question 8

Écrire un sous-programme **trierListe** qui trie les noms d'une liste dans l'ordre alphabétique. Pour faire ce tri, on pourra utiliser la méthode du tri à bulle vue dans le TD3.

```

procedure trierListe(var l : t_listeNoms) ;
var i,j : t_indice;
    enDesordre : Boolean;
begin
    i :=1;
    enDesordre :=True;
    while (i<l.taille) and (enDesordre) do
    begin
        enDesordre :=False;
        For j := l.taille downto i+1 do
            if l.tab[j] < l.tab[j-1] then
                begin
                    permuteNom(l.tab,j,j-1);
                    enDesordre :=True;
                end;
            inc(i);
        end;
    end;

```

Question 9

Écrire un sous-programme **trierFichier** qui, après l'avoir trié, enregistre la liste des noms d'un fichier dans un autre fichier.

```

procedure trierFichier(srcNom, destNom : string) ;
var
    liste : t_listeNoms ;
begin
    lireFichier(srcNom,liste);
    trierListe(liste);
    ecrireFichier(destNom,liste);
end;

```

Question 10

Écrire un programme qui teste les sous-programmes et les déclarations précédents. Ce programme pourra avoir l'interface suivante :

```
q->quitter n->nouveau e->ajouter a->afficher t->trier :
```

```

var
    c : char ;
    s,d : string ; begin
    repeat
        write ('q->quitter n->nouveau e->ajouter a->afficher t->tri : ');
        readln (c);
        case c of
            'e' : begin

```

```
    write ('Entrer le nom du fichier à modifier : ');
    readln (s);
    ajouterNom(s);
end;
'n' : begin
    write ('Entrer nom du nouveau fichier : ');
    readln (s);
    nouveauFichier(s);
end;
'a' : begin
    write ('Entrer le nom du fichier à ouvrir : ');
    readln (s);
    afficherFichier(s);
end;
't' : begin
    write ('Entrer le nom du fichier à trier : ');
    readln (s);
    write ('Entrer un nom de fichier pour enregistrer la liste triée : ');
    readln (d);
    trierFichier(s,d);
end;
end;
until (c = 'q');
end.
```