

## Consignes

Vous composerez ce devoir par binôme. Tous les documents, qu'ils soient de cours, de td ou de tp sont autorisés. Vous rendrez votre copie sous forme d'un fichier source ADA commenté dont le nom doit être : *AP2\_Nom1\_Nom2.adb*. La procédure principale du programme devra avoir le même nom.

Vous écrirez au début du fichier source ADA les mentions suivantes :

```
-----  
-- AP2 - GROUPE ??? - Vendredi 8 Décembre 2006 - 10h30/12h30  
-- NOM1, prénom1, No carte étudiant  
-- NOM2, prénom2, No carte étudiant  
-----
```

Vous enverrez sous forme électronique non compressée votre copie à l'adresse de votre encadrant de TP : {dupas,sylvie.alayrangues}@sic.univ-poitiers.fr.

## 1 Tri par dénombrement

Nous souhaitons réaliser le tri d'un tableau de caractères. Au fur et à mesure des questions, vous écrirez la procédure principale de votre programme permettant de tester toutes les fonctions et procédures demandées. Vous commenterez particulièrement les choix que vous faites sur le mode de passage des paramètres des procédures que vous écrivez

### 1.1 Opérations de bases

Nous souhaitons que les tableaux de caractères ne puissent contenir que les lettres minuscules allant de 'a' à 'z'.

**Question 1 :** Définir le type *T\_Car\_Min*.

**Question 2 :** Définir une constante *N* comme étant le nombre de caractères des tableaux que nous manipulons. A l'aide de cette valeur, définir le type *T\_Tab\_Car\_Min*.

Pour faciliter la saisie et l'affichage d'un tableau de caractères, nous allons définir deux procédures, respectivement nommées *Get* et *Put*.

**Question 3 :** Écrivez la procédure *Get*. Cette procédure ne doit faire aucun affichage et ne doit pas se servir de la procédure *Skip\_Line*. Elle a pour objectif de demander *N* caractères à l'utilisateur et de les ranger dans le tableau.

**Question 4 :** Écrivez la procédure *Put*. Cette procédure doit afficher les lettres en majuscule en les séparant par une virgule. Ainsi le tableau contenant 

a	b	c
---	---	---

 s'affiche sous la forme : *A, B, C*.

Cette procédure ne s'occupe pas de la saisie des caractères du tableau

### 1.2 Opérations sur le tableau de caractères

**Question 5 :** Écrivez une fonction *Min* et une fonction *Max* qui renvoient respectivement la lettre minimale et maximale d'un tableau de caractères minuscules. L'ordre à utiliser est l'ordre lexicographique.

**Question 6 :** Écrivez une fonction qui renvoie la longueur du plus grand plateau dans un tableau de caractères. Un plateau est une suite de caractères identiques.

### 1.3 Processus de tri

Nous allons maintenant nous intéresser plus particulièrement à l'algorithme de tri.

*Principe :*

Pour trier un tableau de caractères, nous comptons le nombre d'occurrences de chacun des caractères et mémorisons ces renseignements dans un tableau  $C$ . A partir de ce tableau, nous pouvons construire le tableau trié en y insérant  $C('a')$  'a', puis  $C('b')$  'b', puis  $C('c')$  'c' et ainsi de suite jusqu'à  $C('z')$  'z'.

**Exemple:**

Tableau de caractères minuscules : 

a	c	a	b	d	e	d
---	---	---	---	---	---	---

Tableau compteur : 

Indice :	a	b	c	d	e	f	...	y	z
Compte :	2	1	1	2	1	0	...	0	0

Nous pouvons maintenant construire le tableau trié en le remplissant par deux 'a', puis un 'b', un 'c', deux 'd' et enfin un 'e'.

Tableau trié : 

a	a	b	c	d	d	e
---	---	---	---	---	---	---

Ce processus nécessite un tableau comptant le nombre d'occurrences d'un caractère dans le tableau. Notre tableau de comptage est un tableau d'entiers indexé par notre type de caractère minuscule.

**Question 7 :** Définissez le type  $T\_Compt\_Car\_Min$ .

**Question 8 :** Écrivez une procédure *Compter* qui remplit un tableau de type  $T\_Compt\_Car\_Min$  avec le nombre d'occurrences de chaque caractère. Donnez en commentaire une idée de la complexité de cette opération en terme de temps.

Nous disposons maintenant de toutes les informations nécessaires afin de construire un tableau de caractères trié.

**Question 9 :** Écrivez la procédure de tri :  $Tri\_Denombrement$ .