

## Les chaînes de caractères

**Exercice 1.** Écrire une fonction `Nettoyage` qui prend en argument une chaîne de caractères et rend comme résultat la même chaîne privée des caractères d'espace.

**Exercice 2.** Écrire une fonction `Alphabetique` qui renvoie le plus grand dans l'ordre alphabétique de deux mots.

**Exercice 3.** Écrire une fonction `Palindrome` qui teste si un mot est un palindrome. Pour mémoire, un palindrome est une expression qui peut se lire indifféremment de gauche à droite, et de droite à gauche : *ressasser*.

**Exercice 4.** Écrire une fonction `Phrase_palindrome` qui teste si une phrase, avec espaces et ponctuation, est un palindrome : *Madam, I'm Adam !*

**Exercice 5.** Écrire une fonction `Anagramme` qui indique si deux mots, sur l'alphabet  $a, \dots, z$ , sont anagrammes l'un de l'autre. Cette fonction devra s'exécuter en temps linéaire.

**Exercice 6.** Écrire une fonction `Reformater` qui prend en argument un texte sous forme de chaînes de caractères et le met en forme suivant les règles

- le texte commence par une lettre en majuscule,
- tout point est précédé d'une lettre, puis suivi d'un espace et d'une lettre en majuscule,
- toute virgule est précédée d'une lettre, puis suivie d'un espace et d'une lettre,
- il n'y a pas d'espaces consécutifs.

Par exemple, le texte

```
le_loup_est_dans_la_foret. Ce_soir, faites_attention.
```

devient

```
Le_loup_est_dans_la_foret. Ce_soir, faites_attention.
```

Les textes qui ne peuvent pas être mis en forme correctement devront déclencher une exception. Une solution simple pour ce problème est d'utiliser un automate déterministe.