

## TD6 : éléments de correction

### Exercice 4 bis : Palindrome

Fonction : **Palindrome**

Entrée : - chaîne : chaîne de caractères

Local : - i : entier

Sortie : - estPalindrome : booléen

Début

Soit estPalindrome = vrai

Soit i = 0

**Tant Que** i < Taille(chaîne) / 2 ET estPalindrome == vrai **Faire**

| **Si** chaîne[i] != chaîne[Taille(chaîne) - 1 - i] **Alors**

| | Soit estPalindrome = faux

| **Fin Si**

| Soit i = i + 1

**Fin Pour**

**Retourner** estPalindrome

Fin

Procédure : **Main**

Entrée : void

Local : - chaîneU : chaîne de caractères

- chaîneEstPalindrome : booléen

Sortie : void

Début

Soit chaîneU = SaisirChaîne()

Soit chaîneEstPalindrome = **Palindrome**(chaîneU)

**Si** chaîneEstPalindrome == vrai **Alors**

| Afficher("La chaîne {chaîneU} est un palindrome.")

**Sinon**

| Afficher("La chaîne {chaîneU} n'est pas un palindrome.")

**Fin Si**

Fin

## Exercice 4 bis : Comparaison de chaînes de même longueur

Fonction : **ComparerChainesMemeLongueur**

Précondition : Longueur(chaine1) == Longueur(chaine2)

Entrée :  
- chaine1 : chaîne de caractères  
- chaine2 : chaîne de caractères

Local :  
- i : entier

Sortie :  
- identiques : booléen

Début

Soit identiques = vrai

Soit i = 0

**Tant Que** i < Longueur(chaine1) OU identiques == vrai **Faire**

| **Si** chaine1[i] != chaine2[i] **Alors**

| | Soit identiques = faux

| **Fin Si**

| Soit i = i + 1

**Fin Pour**

Retourner identiques

Fin

Procédure : Main

Entrée :  
void

Local :  
- chaineU1 : chaîne de caractères  
- chaineU2 : chaîne de caractères  
- comparerU12 : booléen

Sortie :  
void

Début

Soit chaineU1 = SaisirChaîne()

Soit chaineU2 = SaisirChaîne()

// vérification de la précondition avant l'appel

**Si** Longueur(chaineU1) == Longueur(chaineU2) **Alors**

| Soit comparerU12 = **ComparerChainesMemeLongueur**(chaineU1, chaineU2)

| **Si** comparerU12 == vrai **Alors**

| | Afficher("Les chaînes {chaineU1} et {chaineU1} sont identiques.")

| **Sinon**

| | Afficher("Les chaînes {chaineU1} et {chaineU1} sont différentes.")

| **Fin Si**

**Sinon**

| Afficher("Les chaînes {chaineU1} et {chaineU1} sont de longueurs différentes !")

**Fin Si**

Fin

## Exercice 4 bis : Comparaison de chaînes de longueurs différentes

Fonction : **ComparerChaines**

Entrée :  
- chaine1 : chaîne de caractères  
- chaine2 : chaîne de caractères

Local :  
- i : entier

Sortie :  
- n : entier

Début

```
    Soit identiques = vrai
    Soit borneSup = Maximum(Longueur(chaine1), Longueur(chaine2))
    Soit i = 0
    Tant Que i < borneSup OU identiques == vrai Faire
        Si chaine1[i] != chaine2[i] Alors
            Soit identiques = faux
        Fin Si
        Soit i = i + 1
    Fin Pour
    Retourner identiques
```

Fin

Procédure : Main

Entrée :  
void

Local :  
- chaineU1 : chaîne de caractères  
- chaineU2 : chaîne de caractères  
- comparerU12 : booléen

Sortie :  
void

Début

```
    Soit chaineU1 = SaisirChaîne()
    Soit chaineU2 = SaisirChaîne()
    Soit comparerU12 = ComparerChaines(chaineU1, chaineU2)
    Si comparerU12 == vrai Alors
        Afficher("Les chaînes {chaineU1} et {chaineU1} sont identiques.")
    Sinon
        Afficher("Les chaînes {chaineU1} et {chaineU1} sont différentes.")
    Fin Si
```

Fin