Travaux Dirigés n°6
DEUG SCIENCES S2
Chaînes de caractères

Une chaîne de caractères en C est un tableau de `char` particulier. Les caractères à prendre en compte dans le tableau sont en effet suivis par un caractère spécial, utilisé comme *marqueur de fin de mot* : il s'agit du caractère de code ASCII 0, noté '`\0`'.
Dans toute la suite de cette feuille, on travaillera sur des tableaux contenant au plus `MAXLETTRES` caractères, donc sur des chaînes contenant au plus `MAXLETTRES`-1 caractères.

Rappels : Format pour lecture et écriture

- Le format chaîne de caractères est indiqué aux fonctions `scanf` et `printf` par `%s`.
- La fonction `scanf` remplace le premier séparateur *(espace, tabulation ou passage à la ligne)* par '`\0`'.
- La fonction `printf` n'affiche pas '`\0`'.
- Aucune vérification sur la validité des données n'est faite par ces fonctions : pas de vérification que la place réservée est suffisante, ni que le symbole '`\0`' est présent.

On supposera dans la suite que les chaînes manipulées sont correctes.

1. **Saisie et affichage d'une chaîne de caractères**
   - Écrire un programme qui lit une chaîne de caractères entrée par l'utilisateur et qui l'affiche ensuite à l'écran. La chaîne saisie sera stockée dans un tableau. Utiliser le format `%s` de la fonction `scanf`.
   - Écrire une fonction `SaisieControlee` qui prend en argument un tableau de caractères `t` et un entier `n`, et qui effectue la saisie de `n` caractères entrés au clavier, puis le stockage (sous forme d'une chaîne) de ces caractères dans le tableau passé en paramètre. On pourra utiliser la valeur de retour pour indiquer si les `n` caractères ont effectivement été stockés.
2. **Longueur d'une chaîne**
   Écrire une fonction `Longueur` qui reçoit une chaîne de caractères et renvoie le nombre de *caractères* qu'elle contient (c'est-à-dire les caractères placés avant le premier '`\0`').
3. **Nombre de chiffres**
   Écrire une fonction `Nombre_Chiffres` qui reçoit une chaîne de caractères et renvoie le nombre de *chiffres* qu'elle contient.
4. **Comparaison**
   Écrire une fonction `Comparaison` qui reçoit deux chaînes de caractères et qui renvoie 1 si elles sont différentes et 0 sinon.

5. **Copie**
   Écrire une fonction `Copie` qui reçoit deux chaînes de caractères et qui copie la première dans la seconde.

6. **Mise en majuscule**
   Écrire une fonction `Met_En_Majuscule` qui reçoit une chaîne de caractères et qui met en majuscule les lettres minuscules contenues dans cette chaîne.

7. **Supression des caractères non alphabétiques**
   Écrire une fonction `SupprimeNonAlphabetique` qui reçoit une chaîne de caractères et qui supprime de cette chaîne les caractères qui ne sont pas alphabétiques. La fonction retournera la longueur de la chaine obtenue après suppression.

8. **Palindromes**
   Un mot est un *palindrome* s'il peut être lu indifféremment de droite à gauche ou de gauche à droite (comme *été, laval,* ...). On étend cette définition aux phrases en ignorant les séparateurs, la ponctuation et la différence entre minuscules et majuscules (ex : *Esope reste et se repose*). Le but de l'exercice est de déterminer si une phrase est un palindrome. Comment procéder sans modifier la phrase d'origine ?

   - Première méthode :
     Écrire une fonction `Miroir` qui reçoit une chaîne de caractères et la remplace par la chaîne écrite en sens inverse.
     Utiliser cette fonction pour déterminer si une chaîne est un palindrome.
   - Deuxième méthode :
     Procéder directement sur la seule chaîne. On pourra même écrire une fonction ne nécessitant pas de "nettoyer" la chaîne.