

1. Implémentez la fonction **longueur_chaine**, qui reçoit une chaîne de caractères **str** valide en paramètre et doit renvoyer la taille de la chaîne **str**.
Rappel : la taille d'une chaîne est le nombre de caractères avant le `'\0'`.
2. Implémentez la fonction **est_palindrome**, qui reçoit une chaîne de caractères **str** valide en paramètre et doit renvoyer vrai si et seulement si **str** est un palindrome.
Une chaîne est un palindrome si elle se peut se lire indifféremment de gauche à droite, et de droite à gauche. Par exemple, "tnt", "kayak", "essayasse" ou "avid diva" sont des palindromes.
3. Implémentez la fonction **comparaison**, dont le prototype est donné ci-dessous.

```
int comparaison(const char * str1, const char * str2);
```

Cette fonction reçoit deux chaînes valides (**str1** et **str2**) et renvoie :

- un résultat négatif si $str1 < str2$,
- un résultat nul si $str1 = str2$,
- un résultat positif si $str1 > str2$.

La comparaison entre ces deux chaînes se fait selon l'ordre lexicographique. De manière informelle, c'est l'*ordre du dictionnaire*. Par exemple, "bouh" < "bwah" car 'o' < 'w'. De plus, "bouh" > "bo" car 'u' > '\0'.

4. Implémentez la fonction **index_of**, dont le prototype est donné ci-dessous.

```
int index_of(const char * big_str, const char * little_str);
```

Cette fonction reçoit deux chaînes valides et renvoie :

- -1 si **big_str** ne contient pas **little_str**,
 - L'indice où commence la première occurrence de **little_str** dans **big_str** sinon.
- Par exemple, `index_of("Richard Stallman", "man")` doit renvoyer 13.

5. Implémentez la fonction **occurrences**, dont le prototype est donné ci-dessous.

```
int occurrences(const char * str, char value, int * result);
```

Cette fonction a pour paramètres d'entrée une chaîne valide **str** et un caractère **value**. Elle a également un paramètre de sortie : un tableau d'entiers **results** de taille supérieure ou égale à la longueur de la chaîne.

Cette fonction doit renvoyer le nombre d'occurrences de **value** dans **str** et mettre les indices de ces occurrences dans **result**.

Par exemple, `occurrences("solaire of astora", 'a', res)` doit renvoyer 3 et placer [3, 11, 16] au début du tableau `res`.