

TP Informatique 10

⚠ On rappelle qu'un script (fichier *.py) doit être enregistré et exécuté (touche F5) pour que les fonctions saisies dans le script soient utilisables dans la console.

Dans ce TP, on s'intéresse à la recherche d'un mot dans un texte.

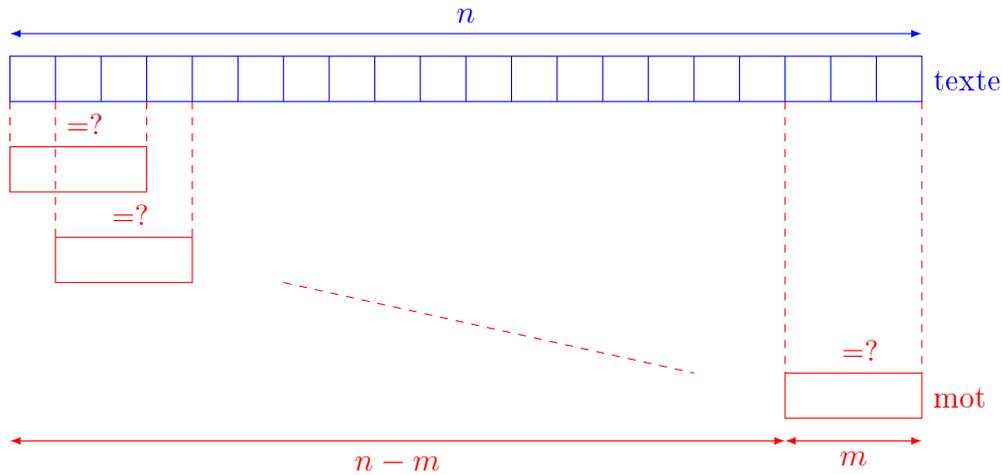


FIGURE 1 – Recherche d'un mot dans un texte

La mention « égalité sur des chaînes » concerne des chaînes de taille > 1 .

Exercice 1

Chaque réponse fera l'objet d'une vérification par expérimentation. Les saisies seront effectuées dans la console.

1. Saisir `texte="bonjour"`, `n=len(texte)`, `mot="jour"`, `m=len(mot)`. Que contiennent les variables `n` et `m` ?
2. (a) Saisir `k=0`. Que renvoie `texte[k:k+m]` ? Que renvoie `texte[k:k+m]==mot` ?
(b) Reprendre la question précédente avec `k=1`, `k=2`, `k=3` et `k=4`.
(c) Jusqu'à quelle valeur de `k` fonction de `n` et `m` le test `texte[k:k+m]==mot` est-il pertinent ?

Exercice 2

Dans cet exercice, on autorise l'usage du slicing et les tests d'égalité sur des chaînes.

1. Écrire une fonction `rech1(mot, texte)` d'arguments `mot` et `texte` des chaînes qui renvoie `True` si `mot` est présent dans `texte` et `False` sinon.
2. Écrire une fonction `rech2(mot, texte)` avec boucle `while` d'arguments `mot` et `texte` des chaînes qui renvoie `True` si `mot` est présent dans `texte` et `False` sinon.

Exercice 3

Dans cet exercice, l'usage du slicing n'est pas autorisé.

1. Écrire une fonction `detect3(elt,T,pos)` d'arguments `elt` et `T` des chaînes et `pos` un entier, qui renvoie `True` si la chaîne `elt` coïncide avec `T[pos:pos+m]` et `False` sinon avec `m=len(elt)`.
2. Écrire une fonction `rech3(mot,texte)` d'arguments `mot` et `texte` des chaînes qui renvoie `True` si `mot` est présent dans `texte` et `False` sinon.
3. Quel est l'intérêt de procéder sans slicing ?

Exercice 4

Dans cet exercice, l'usage du slicing est autorisé mais pas les tests d'égalité entre chaînes.

1. Écrire une fonction `str_egal(x,y)` d'arguments `x` et `y` des chaînes de même taille qui renvoie `True` si elles sont égales et `False` sinon.
2. Écrire une fonction `rech4(mot,texte)` d'arguments `mot` et `texte` des chaînes qui renvoie `True` si `mot` est présent dans `texte` et `False` sinon.

Exercice 5

Dans cet exercice, l'usage du slicing n'est pas autorisé.

1. Écrire une fonction `detect5(elt,T,pos)` avec boucle `while` d'arguments `elt` et `T` des chaînes et `pos` un entier, qui renvoie `True` si la chaîne `elt` coïncide avec `T[pos:pos+m]` et `False` sinon avec `m=len(elt)`.
2. Écrire une fonction `rech5(mot,texte)` avec boucle `while` d'arguments `mot` et `texte` des chaînes qui renvoie `True` si `mot` est présent dans `texte` et `False` sinon.

Exercice 6

Dans cet exercice, l'usage du slicing n'est pas autorisé.

Écrire une fonction `remp(mot1,mot2,texte)` qui remplace dans `texte` toutes les occurrences de `mot1` par `mot2`. On pourra faire appel à la fonction `detect3` précédemment définie.

Exercice 7

Dans cet exercice, les tests d'égalité entre caractères sont autorisés mais pas les tests d'égalité entre chaînes contenant plus de un caractère.

Écrire une fonction `pos(mot1,mot2)` d'arguments `mot1` et `mot2` des chaînes qui renvoie 0 si `mot1` et `mot2` sont égaux, 1 si `mot1` est après `mot2` dans le dictionnaire et `-1` sinon. On testera la fonction sur les couples de mots suivants en échangeant les rôles pour chaque couple :

("a", "b") ("aa", "b"), ("aa", "ab") ("ab", "abc") ("abc", "abc")