

**PROGRAMMATION AVANCÉE ET ALGORITHMIQUE**

*-Travaux Pratique 5-*

***Réalisé par :***

Chao PAN

***Encadrée par :***

M. Renaud Marlet et M. Pascal Monasse

***Octobre 2019***

TP5

Levenshtein distance :

Où a et b signifient deux strings, i, j représentent leur position dans un string

Quand min (i, j) =0, au moins d’un string fait égal vide, donc, une distance entre a et b est max (i, j).

Si , Supprimer

Si , Insérer à la position

Si , remplacer avec

Damerau-Levenshtein :

Ou a et b signifient deux strings, i, j représentent leur position dans un string

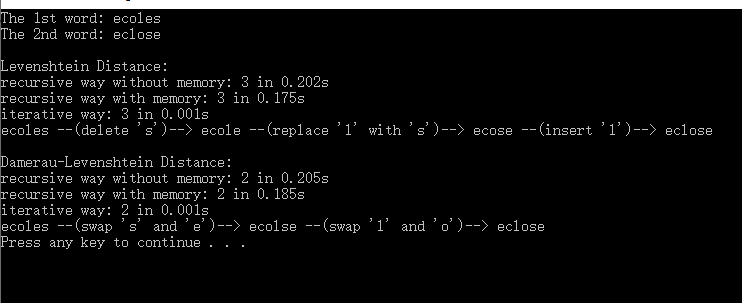
Quand min (i, j) =0, au moins d’un string fait égal vide, donc, une distance entre a et b est max (i, j).

Si , Supprimer

Si , Insérer à la position

Si , remplacer avec

Si , échanger avec



4.

À priori, la longueur de string a est m et celle de string b est n. Un tableau ci-après représente la complexité en espace et celle-ci en temps pour deux types de méthodes récursives.

|  |  |  |
| --- | --- | --- |
| Type | Complexité en espace | Complexité en temps |
| Sans mémoire | O (1) | Exponentielle |
| Avec mémoire | O (mn) | Exponentielle |

Pour la méthode récursive sans mémoire, la complexité en espace est O (1) parce qu’il n’y a pas de variables pour stocker des valeurs intermédiaires. Pour la méthode récursive avec mémoire, la complexité en espace est O (mn) parce qu’il y a une matrice en 2 dimensions. Puis, la complexité en temps pour les deux types est exponentielle.

5.

Pour deux types (sans mémoire et avec mémoire), leur temps d’exécution est plus ou moins identique parce qu’ils ont pris la façon récursive.

7.

Pour un algorithme itératif, sa complexité en espace est O(mn) parce qu’il y a une matrice en 2 dimensions et sa complexité en temps est O(mn) parce qu’il faut parcourir cette matrice pour obtenir une distance la plus petite.

8.

Pour un algorithme itératif, son temps d’exécution est beaucoup moins que celui d’un algorithme récursif, parce que la complexité en temps pour un algorithme itératif est O(mn). En revanche, celle-ci pour un algorithme récursif est exponentielle.

11.

Pour une version itérative, on prend une matrice pour stocker des valeurs intermédiaires.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | e | c | l | o | s | e |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| e | 1 | 0 | 1 | 2 | 3 | 4 | 5 |
| c | 2 | 1 | 0 | 1 | 2 | 3 | 4 |
| o | 3 | 2 | 1 | 1 | 1 | 2 | 3 |
| l | 4 | 3 | 2 | 1 | 2 | 2 | 3 |
| e | 5 | 4 | 3 | 2 | 2 | 3 | 2 |
| S | 6 | 5 | 4 | 3 | 3 | 2 | 1 |

Mais, pour chaque calcul, il n’a que besoin de trois valeurs, c’est-à-dire, il suffit de uniquement garder deux lignes pour stocker ces valeurs. Dans ce cas-là, notre algorithme est linéaire en espace O(n), où n est la longueur de string s2.