

TP Informatique 10

Exercice 1

Étant données deux chaînes de caractères non vides $X = "x_0 \dots x_{n-1}"$ et $Y = "y_0 \dots y_{m-1}"$, on appelle *sous-suite commune* à X et Y une chaîne de caractères $Z = "z_0 \dots z_{p-1}"$ telle qu'il existe $\varphi : \llbracket 0; p-1 \rrbracket \rightarrow \llbracket 0; n-1 \rrbracket$ et $\psi : \llbracket 0; p-1 \rrbracket \rightarrow \llbracket 0; m-1 \rrbracket$ strictement croissantes et vérifiant

$$"z_0 \dots z_{p-1}" = "x_{\varphi(0)} \dots x_{\varphi(p-1)}" = "y_{\psi(0)} \dots y_{\psi(p-1)}"$$

La chaîne vide "" est considérée sous-suite commune de X et Y , en particulier si X ou Y est une chaîne vide. On appelle *plus longue sous-suite commune* à deux chaînes de caractères une sous-suite commune dont la taille est maximale.

On rappelle que pour une chaîne de caractères S , l'instruction $S[:k]$ renvoie la chaîne extraite d'indice 0 jusqu'à $k-1$ pour k entier non nul (chaîne vide si $k=0$).

On a :

- Si $X[n-1] = Y[m-1]$, alors $Z[p-1] = X[n-1] = Y[m-1]$ et $Z[:p-1]$ est une plus longue sous-suite commune à $X[:n-1]$ et $Y[:m-1]$;
- Si $X[n-1] \neq Y[m-1]$ et $Z[p-1] \neq X[n-1]$, alors Z est une plus longue sous-suite commune à $X[:n-1]$ et Y ;
- Si $X[n-1] \neq Y[m-1]$ et $Z[p-1] \neq Y[m-1]$, alors Z est une plus longue sous-suite commune à X et $Y[:m-1]$.

On note $L(X, Y)$ la taille d'une plus longue sous-suite commune à X et Y et on définit

$$\forall (i, j) \in \llbracket 0; n \rrbracket \times \llbracket 0; m \rrbracket \quad c(i, j) = L(X[:i], Y[:j])$$

Les fonctions seront testées au fur et à mesure de leur saisie. Les fonctions à compléter sont dans le fichier TP10_EX01.py.

1. Écrire une fonction `PLSC1(X, Y)` d'arguments X, Y des chaînes de caractères et qui renvoie une plus longue sous-suite commune à X et Y en réalisant le calcul de manière descendante sans mémorisation.
2. Écrire une fonction `PLSC2(X, Y)` d'arguments X, Y des chaînes de caractères et qui renvoie une plus longue sous-suite commune à X et Y en réalisant le calcul de manière descendante avec mémorisation.
3. Déterminer $c(i, 0)$ pour $i \in \llbracket 0; n \rrbracket$ et $c(0, j)$ pour $j \in \llbracket 0; m \rrbracket$.
4. Pour $(i, j) \in \llbracket 1; n \rrbracket \times \llbracket 1; m \rrbracket$, déterminer une expression de $c(i, j)$ en fonction de $c(i-1, j-1)$, $c(i-1, j)$ et $c(i, j-1)$. On prendra soin de distinguer deux régimes.
5. Compléter la fonction `C_tab(X, Y)` d'arguments X, Y des chaînes de caractères pour qu'elle renvoie la matrice $(C(i, j))_{(i,j) \in \llbracket 0; n \rrbracket \times \llbracket 0; m \rrbracket}$ calculée de manière ascendante.
6. Écrire une fonction `PLSC3(X, Y)` d'arguments X, Y des chaînes de caractères et qui renvoie une plus longue sous-suite commune à X et Y en utilisant le résultat de `C_tab(X, Y)`.
7. Compléter la fonction `PLSC_tab_all(X, Y)` d'arguments X, Y des chaînes de caractères et qui renvoie la liste de toutes les plus longues sous-suites communes à X et Y .