Esiee-Paris - cours d'algorithmique - Feuille d'exercices numéro 4

R. Natowicz, I. Alame, A. Çela, X. Hilaire, T. Wu, W. Xu

Exercice 8. Plus longue séquence commune à deux séquences. Une séquence $T = t_0 t_1 ... t_{m-1}$ est une suite de symboles. En supprimant des symboles de T on obtient une sous-séquence S. On peut supprimer un nombre quelconque de symboles, au moins 0 et au plus m. En particulier si 0 symbole supprimé alors $S = T$, si m symboles supprimés alors S est la séquence vide.

Soient T et U deux séquences. Une séquence S qui est sous-séquence de T et sous-séquence de U est une séquence commune à T et U.

On veut connaître la longueur d'une plus longue séquence commune (plsc) aux séquences T et U puis en construire une.

Exemples :

- `algorithme = plsc(aligatorithme, algorithmique);`
- `algo = plsc(uvuavvzzlwxyyvgzvowx, tsbakghlpgmolibbdène).`

Il s'agit d'un problème de base en bioinformatique : le calcul d'une plus longue séquence commune à deux séquences de gènes ou autres. Il est également à la base de l'outil `diff` des systèmes Unix et Linux qui affiche les différences entre deux fichiers. La commande `diff` voit un fichier comme une suite de lignes, chaque ligne est un symbole. Elle calcule une plsc aux deux fichiers puis elle affiche les lignes des fichiers qui ne sont pas dans la plsc.

Soient m et n les longueurs des séquences T et U et soit $l(m,n)$ la longueur s'une plsc à T et U.
Supposons le problème résolu. Comment la valeur $l(m,n)$ a-t-elle été obtenue ? Il n'y a que quatre possibilités.

1. les deux derniers symboles de T et U sont égaux ($t_{m-1} = u_{n-1}$). Alors $l(m,n) = 1 + l(m-1,n-1)$.
   Démonstration : soit $S = s_0 ... s_{p-1}$ une plsc à T et U.
   a) Le dernier symbole de S est t_{m-1} ($s_{p-1} = t_{m-1}$). Si ça n'était pas le cas il suffirait d'ajouter t_{m-1} à la fin de S pour obtenir une sous-séquence commune (ssc) à T et U plus lonque que S. Donc S ne serait pas une plsc.
   b) Soit S' la séquence S privée de son dernier symbole et de même pour les séquences T' et U'. La séquence S' est une plsc à T' et U'. Si ce n'était pas le cas, il existerait une ssc à T' et U' plus longue que S'. Soit S'' cette séquence. Alors, S'' prolongée par t_{m-1} serait une ssc à T et U, plus longue que S. Donc S ne serait pas une plsc à T et U.
   La séquence S' étant une plsc à T' et U', sa longueur est $l(m-1,n-1)$ (par définition de $l(.,.)$).
   Enfin, la plsc S étant la séquence S' prolongée par le symbole t_{m-1}, sa longueur est $1 + l(m-1,n-1)$.
2. les deux derniers symboles de T et U sont différents et aucun des deux n'est le dernier symbole de S. Alors S est une plsc de T privée de son dernier symbole et de U privée de son dernier symbole. Donc $l(m,n) = l(m-1,n-1)$.
3. les deux derniers symboles de T et U sont différents et le dernier symbole de T n'est pas le dernier de S. Alors $l(m,n) = l(m-1,n)$.
4. les deux derniers symboles de T et U sont différents et le dernier symbole de U n'est pas le dernier de S. Alors $l(m,n) = l(m,n-1)$.

La longueur $l(m,n)$ est donc :

- si les deux derniers symboles de T et U sont égaux, $l(m,n) = 1 + l(m-1,n-1)$.
- s'ils sont différents, alors $l(m,n) = \max\{l(m-1,n-1), l(m-1,n), l(m,n-1)\}$

On peut remarquer que la deuxième situation ci-dessus est un cas particulier de la troisième : il s'agit de la troisième situation dans laquelle, en plus, le dernier symbole de U n'est pas le dernier symbole de S. Donc $l(m-1,n-1) \leq l(m-1,n)$. Pour la même raison, elle est un cas particulier de la quatrième situation. Donc $l(m-1,n-1) \leq l(m,n-1)$. Donc $\max\{l(m-1,n-1), l(m-1,n), l(m,n-1)\} = \max\{l(m-1,n), l(m,n-1)\}$. Le 2e cas ci-dessus est inutile.

Ainsi, lorsque les deux derniers symboles de T et U sont différents nous avons simplement

$$l(m,n) = \max\{l(m-1,n), l(m,n-1)\}$$

Généralisons ce résultat. Le p-préfixe d'une séquence T de longueur m est la séquence de ses p premiers symboles. En particulier son 0-préfixe est la séquence vide et son m-préfixe est la séquence T. La valeur $l(m,n)$ est donc la longueur d'une plsc à T et U. Nous généralisons aux longueurs $l(p,q)$ des plsc aux p et q préfixes de T et U.

1. Base : $l(0,q) = 0, \forall q, 0 \leq q < n+1$ et $l(p,0) = 0, \forall p, 0 \leq p < m+1$ car les 0-préfixes sont vides.
2. Hérédité, $1 \leq p < m+1, 1 \leq q < n+1$:
   - Si $t_{p-1} = u_{q-1}$: $l(p,q) = 1 + l(p-1,q-1)$
   - si $t_{p-1} \neq u_{q-1}$: $l(p,q) = \max\{l(m-1,n), l(m,n-1)\}$.

On représente les séquences par des chaînes (`String`). On rappelle deux méthodes de la classe `String` utiles pour la suite.

- `S.length()` retourne la longueur de la chaîne,
- `S.carAt(i)` retourne le caractère d'indice i,
- `S.substring(i, j)` retourne la chaîne "s_i,...,s_{j-1}", sous-chaîne de longueur $j-i$ commençant à l'indice i. La sous-chaîne `S.substring(0, j)` est le j-préfixe de longueur j. En particulier `S.substring(0, 0)` est la chaîne vide,
- `S1+S2` est la concaténation des chaînes $S1$ et $S2$. Exemple : ``bonjou'' + ``bonsoir''.substring(6,1)` est la chaîne ``bonjour''.

Question 1. Écrire une fonction `int[][] calculerL(String T, String U)` qui calcule et retourne le tableau $L[0:m+1][0:n+1]$ de terme général $L[p][q] = l(p,q)$.
Bien comprendre que la ligne $L[p]$ contient les valeurs $l(p,q)$, $0 \leq q < n+1$, c'est-à-dire les longueurs des plsc au p-préfixe de T et aux q-préfixes de U.

Question 2. Écrire une fonction `String plsc(String T, String U, int[][] L, int p, int q)` qui retourne une plsc aux p et q préfixes des chaînes T et U.
L'appel principal de cette fonction est `String S = plsc(T, U, L, m, n)` où m et n sont les longueurs des chaînes T et U.