$Z_3 = \langle B, B \rangle = X_{4-5}$ sottostringa, e sottosequenza rispetto agli indici $i_1 = 2$ $i_2 = 5$

Quante sono le sottosequenze di una stringa?

$$\sum_{k=0}^{m} \binom{m}{k} = 2^m$$

→ spazio delle sottosequenze è enorme

Determinazione della sottosequenza comune di massima lunghezza di due stringhe ("<u>Longest Common Subsequence</u>" LCS):
date X, Y determina Z tale che
1) Z è sottosequenza di X e di Y (cioè sottoseq. comune)
2) Z è la più lunga tra tutte le sottosequenze comuni

Applicazioni: bioinformatica, word processing, ...

Esempio:

Provo per bruteforce tutte le possibili idee/soluzioni. L'idea di usare un approccio esponenziale non è indicato, dovuto al suo costo.

$X = \langle A, B, C, B, B, D \rangle$
$Y = \langle A, D, C, C, B, D \rangle$

$Z = \langle A, C, B, D \rangle$ è una LCS (qui anche l'unica)
↳ $i_1 = 1$ $i_2 = 3$ $i_3 = 4 \vee 5$ $i_4 = 6$
↳ $j_1 = 1$ $j_2 = 3 \vee 4$ $j_3 = 5$ $j_4 = 6$

Alg. esaustivo: comprende e usa tutte le soluzioni.
In questo caso, anche aggiungendo A, rimane comunque LCS, per semplice proprietà associativa (somma di sottostringhe).

Complessità dell'algoritmo esaustivo?

$|X| = m$ $|Y| = n$ → $\Omega(2^m \times 2^n) = \Omega(2^{m+n})$ <u>esponenziale!</u>

cerco allora di individuare una struttura ricorsiva

$X = \langle X', a \rangle$

$Y = \langle Y', a \rangle$

$\Rightarrow$ ho che $Z = LCS(X, Y) = \langle Z', a \rangle$, dove

$Z' = LCS(X', Y')$

Z' è la soluzione ottima su un pezzo di input più piccolo. Nel caso non finisca con "a", andrò a prendere Z' per ricorsione. Questo perché, rimpicciolendo una delle due stringhe di input, riduco la dimensione del problema sapendo che solo una parte è comune ("a").
Ciò nel caso X; analogamente, nel caso di Y
Z quindi ragiona per sottoistanze, per DP.

l'altro caso:

$X = \langle X', a \rangle$

$Y = \langle Y', b \rangle$

$\Rightarrow$ Z è la stringa più lunga tra $LCS(X', Y)$ e $LCS(X, Y')$

spazio sottoproblemi: $S = \{ LCS(X_i, Y_j) : 0 \leq i \leq m, 0 \leq j \leq n \}$

prefissi

$|S| = (m+1)(n+1)$

Proprietà di sottostruttura ottima per il sottoproblema $LCS(X_i, Y_j)$

$X_i = \langle x_1, x_2, \dots, x_i \rangle$

$Y_j = \langle y_1, y_2, \dots, y_j \rangle$

sia $Z = \langle z_1, z_2, \dots, z_k \rangle = LCS(X_i, Y_j)$

0) (caso base) se $i = 0$ o $j = 0$, allora $Z = \varepsilon$

1) $(i, j > 0)$

se $(x_i = y_j)$ allora

a) $z_k = x_i \ (= y_j)$

b) $Z_{k-1} = LCS(X_{i-1}, Y_{j-1})$

Del problema considerato, al di là dei casi base, consideriamo che:
-ci può essere un caso fortunato (quindi x(i) =y(j), quindi stessi caratteri alla fine.
La notazione usata riporta questo.

2) $(i, j > 0)$

se $(x_i \neq y_j)$ allora

Z è la stringa di lunghezza massima tra $LCS(X_i, Y_{j-1})$ e $LCS(X_{i-1}, Y_j)$

Dimostrazione della proprietà di sottostruttura

0) banale

1) $LCS(X_i, Y_j) = Z = \langle z_1, \dots, z_k \rangle$

$= \langle x_{i_1}, \dots, x_{i_k} \rangle$

$= \langle y_{j_1}, \dots, y_{j_k} \rangle$

dove $1 \leq i_1 < i_2 < \dots < i_k \leq i$

$1 \leq j_1 < j_2 \dots < j_k \leq j$

Stiamo lavorando sui due prefissi (i, parte prima e j, parte dopo)
La dimostrazione viene fatta per assurdo, dicendo che l'indice dell'ultimo carattere della soluzione non è l'indice dell'ultimo carattere del mio input; stessa cosa per j, simmetricamente.

a) per assurdo, supponiamo $z_k \neq (x_i = y_j)$

$z_k = x_{i_k} = y_{j_k} \Rightarrow i_k < i$

$j_k < j$

La dimostrazione sta affermando banalmente che la sottolunghezza più lunga sarà essa stessa più lunga di quella maggiore su tutto l'array: assurdo!
Per la proprietà degli indici considerati, mettiamo l'indice (k+1) come maggiore dell'indice attuale; è una supposizione sbagliata. Questo perché la subsequence non può essere sottosequenza comune, rispetto invece alla vera sequenza maggiore.

Considera $Z' = \langle Z, x_i \rangle$

$|Z'| = k+1 > |Z|$

dimostra che Z' è sottosequenza comune di X_i e Y_j

$\Rightarrow$ assurdo perché e per ipotesi $Z = LCS(X_i, Y_j)$

$1 \leq i_1 < i_2 < \dots < i_k < i_{k+1} = i$

$1 \leq j_1 < j_2 < \dots < j_k < j_{k+1} = j$

b) $Z_{k-1} = \langle X_{i_1}, X_{i_2}, \dots, X_{i_{k-1}} \rangle$

$= \langle Y_{j_1}, Y_{j_2}, \dots, Y_{j_{k-1}} \rangle$

$i_{k-1} \leq i-1$
$j_{k-1} \leq j-1$ $\Rightarrow$ Z_{k-1} è sottosequenza di X_{i-1} e Y_{j-1}

ora dimostro che $Z_{k-1} = LCS(X_{i-1}, Y_{j-1})$

suppongo non vero, per assurdo: allora $\exists$ un'altra di lunghezza $\geq k$; a questa aggiungo in coda x_i, ottenendo una CS (X_i, Y_j) di lunghezza $\geq k+1$: assurdo.

2) $x_i \neq y_j$ $(i, j > 0)$

basta dimostrare che $Z = LCS(X_i, Y_{j-1})$ oppure $Z = LCS(X_{i-1}, Y_j)$

SC=Sequenza comune (common sequence)

a) $i_k = i$

$\Rightarrow j_k < j \Rightarrow Z$ è $SC(X_i, Y_{j-1})$

Se l'ultimo carattere fosse dato dall'i-esimo di x, non può essere l'ultimo sia in x che in y, perché se è l'ultimo da una parte non può esserlo anche dall'altra. Aggiungendo l'ultimo carattere di j, comunque ottengo una stringa più lunga di quella base; assurdo pure qui.

$x_i \neq y_j$ Z è anche $LCS(X_i, Y_{j-1})$ $\rightarrow$ per assurdo, come prima

b) $i_k < i$

Non finisce con l'ultimo carattere in x è l'idea è identica a prima.

$\Rightarrow Z$ è $SC(X_{i-1}, Y_j)$

Z è anche $LCS(X_{i-1}, Y_j)$ $\rightarrow$ per assurdo, come prima

Fine dim.

Passo 2: ricorrenza sui costi

chiamo $\ell(i,j) = |LCS(X_i, Y_j)|$

$\hookrightarrow$ funz. di costo