

Zad 5.

wtorek, 9 maja 2023 13:14

5. (2pkt) Ułóż algorytmy, które dla danych podciągów x i y rozwiązują następujące wersje problemu znajdowania najdłuższego wspólnego podciągu:

- znajdowanie najdłuższego wspólnego podciągu zawierającego podciąg "matma",
- znajdowanie najdłuższego wspólnego podciągu nie zawierającego podciągu "matma",
- znajdowanie najdłuższego wspólnego podciągu zawierającego pod słowo "matma",
- znajdowanie najdłuższego wspólnego podciągu nie zawierającego pod słowa "matma".

1) dodajemy dodatkowy wymiar do dp

$s_1[1..N], s_2[1..M]$ - wejściowe napisy

$m = [m, a, t, m, a]$

$dp[n][m][L]$ - długość najdłuższego wspólnego podciągu $s_1[1..n]$ i $s_2[1..m]$ zawierającego L -elementowy prefiks słowa "matma"

$dp[n][m][0] = \text{tak jak zwykły lcs}$

$$dp[n][m][L] = \begin{cases} 0 & n < L \vee m < L \vee L > r \\ \max(dp[n-1][m][L], dp[n][m-1][L]) & s_1[n] \neq s_2[m] \\ 1 + dp[n-1][m-1][L-1] & s_1[n] = s_2[m] \text{ wpp} \end{cases}$$

odpowiedź: $dp[N][M][5]$

2) tak jak w (1)

odpowiedź: $\max_{0 \leq L \leq 4} (dp[N][M][L])$

3)

$dp[n][m][L]$ - najdłuższy wspólny podciąg zawierający podstawo będące L -elementowym prefiksem słowa "matma"

$dp[n][m][0] = \text{jak zwykły lcs}$

$$dp[n][m][L] = \begin{cases} 0 & n < L \vee m < L \vee L > r \\ \max(dp[n-1][m][L], dp[n][m-1][L]) & s_1[n] \neq s_2[m] \\ 1 + dp[n-1][m-1][L] & (*) \text{ wpp} \end{cases}$$

(*) jeśli

4) analogicznie do (2)