

ALGORYTMY I STRUKTURY DANYCH

PROGRAMOWANIE DYNAMICZNE

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

1. [*] Udowodnij, że algorytm wyznaczający współczynnik dwumianowy $\binom{n}{k}$ oparty na metodzie “dziel i zwyciężaj” oblicza w sumie $2\binom{n}{k} - 1$ współczynników.
2. [**] Pokaż, jak obliczyć współczynnik dwumianowy $\binom{n}{k}$ używając tablicy pomocniczej $L[0 \dots k]$ oraz $O(1)$ dodatkowych komórek pamięci. Następnie pokaż, jak rozwiązać to zadanie używając jeszcze mniejszej tablicy pomocniczej $L[0 \dots \min(k, n - k)]$ oraz $O(1)$ dodatkowych komórek pamięci.
3. [*] Wyznacz optymalną kolejność mnożenia macierzy $A_0 \times A_1 \times A_2 \times A_3 \times A_4$, gdy macierze te mają rozmiary odpowiednio 10×4 , 4×5 , 5×20 , 20×2 , 2×50 .
4. [*] Pokaż, że w celu pełnego opatrzenia nawiasami iloczynu n macierzy potrzebnych jest $n - 1$ par nawiasów.
5. [***] Rozważmy dwa algorytmy obliczania optymalnego nawiasowania iloczynu n macierzy:
 - (a) Algorytm naiwny, polegający na wygenerowaniu wszystkich poprawnych nawiasowań i obliczenie liczby mnożeń skalarów w każdym z nich.
 - (b) Algorytm rekurencyjny, oparty na metodzie “dziel i zwyciężaj”.Który z tych sposobów jest efektywniejszy? Odpowiedź uzasadnij obliczeniami.
6. [**] Zmodyfikuj podany na wykładzie algorytm dla problemu *MCM* (ang. *matrix-chain multiplication*), tak aby wypisywał on ciąg n macierzy wraz z poprawnie rozstawionymi nawiasami na podstawie wyliczonej w trakcie działania algorytmu tablicy pomocniczej $L[0 \dots n - 1][0 \dots n - 1]$. Samo wypisanie wspomnianego ciągu macierzy wraz z nawiasami powinno zająć $O(n)$ czasu.
7. [*] Wyznacz najdłuższy wspólny podciąg dla ciągów *ABBACCBA* i *BABCACBA*.
8. [**] Zmodyfikuj podany na wykładzie algorytm dla problemu *LCS* (ang. *longest common subsequence*), tak aby zrekonstruować najdłuższy wspólny podciąg ciągów $X[1 \dots n]$ i $Y[1 \dots m]$ na podstawie wyliczonej w trakcie działania algorytmu tablicy pomocniczej $L[0 \dots n][0 \dots m]$. Sama rekonstrukcja wspomnianego podciągu powinna zająć $O(m + n)$ czasu.

9. [**] Pokaż, jak obliczyć długość największego wspólnego podciągu używając tablicy pomocniczej $L[0 \dots 2 \min(n, m)]$ oraz $O(1)$ dodatkowych komórek pamięci. Następnie pokaż, jak rozwiązać to zadanie używając jeszcze mniejszej tablicy pomocniczej $L[0 \dots \min(n, m)]$ oraz $O(1)$ dodatkowych komórek pamięci.
10. [****] Podaj algorytm znajdujący w czasie $O(n \log n)$ i dodatkowej pamięci $O(n)$ najdłuższy niemalejący podciąg w zadanym ciągu n liczb.