

ALGORYTMY I STRUKTURY DANYCH

PROGRAMOWANIE DYNAMICZNE

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

1. [*] Przedstaw implementację rekurencyjnego algorytmu obliczającego n -tą liczbę Fibonacciego, która wykorzystuje spamiętywanie (memoization).
2. [*] Udowodnij, że rekurencyjna procedura obliczająca współczynnik dwumianowy $\binom{n}{k}$ będzie w sumie wywoływana $2\binom{n}{k} - 1$ razy.
3. **[**] 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.
4. **[**] 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.
5. **[**] Pokaż, jak zrekonstruować największy wspólny podciąg tylko na podstawie tablicy kosztów C i ciągów wejściowych X i Y . Twój algorytm powinien działać w czasie $O(m + n)$, gdzie $m = |X|$ i $n = |Y|$.
6. **[**] Zmodyfikuj podany na wykładzie algorytm dla problemu optymalnego mnożenia macierzy, tak aby wypisywał on ciąg n macierzy wraz z poprawnie roztawionymi nawiasami na podstawie wyliczonej w trakcie działania algorytmu tablicy pomocniczej $L[1 \dots n][1 \dots n]$, zawierającej informacje o najmniej kosztownych podziałach. Samo wypisanie wspomnianego ciągu macierzy wraz z nawiasami powinno posiadać złożoność czasową rzędu $O(n)$.
7. **[**] Rozważmy dwa algorytmy obliczania optymalnego nawiasowania iloczynu n macierzy:
 - (a) algorytm naiwny, polegający na wygenerowaniu wszystkich poprawnych nawiasowań i obliczeniu 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.
8. **[**] Dany jest n -elementowy słownik statyczny z wyrazami $w_1, w_2, \dots, w_n$. Znamy też prawdopodobieństwa zapytań o poszczególne wyrazy $p_1, p_2, \dots, p_n$, gdzie $\sum_{i=1}^n p_i = 1$ oraz $p_i > 0$. Wyrazy te należy umieścić w drzewie BST w taki sposób, aby zminimalizować koszt wyszukiwania wyrazów z tego słownika, przy czym koszt wyszukiwania to długość ścieżki poszukiwań rozpoczynającej się od korzenia. Należy więc zbudować takie drzewo BST, które minimalizuje sumę $\sum_{i=1}^n d_i p_i$, gdzie d_i to głębokość węzła zawierającego wyraz w_i . Przedstaw algorytm rozwiązujący ten problem, uzasadnij jego poprawność i oszacuj złożoność obliczeniową.
9. **[**] Niech dany będzie graf skierowany silnie spójny z nieujemnymi wagami na krawędziach $D = (V, E, w : E \mapsto \mathbf{R}_+)$. Przeanalizuj algorytm Floyda wyznaczania długości najkrótszych ścieżek dla każdej pary wierzchołków w takim grafie. Uzasadnij jego poprawność i oszacuj złożoność obliczeniową. Jak należy ten algorytm zmodyfikować, aby można było później zrekonstruować najkrótszą ścieżkę pomiędzy każdą parą wskazanych wierzchołków?