
ALGORYTMY I STRUKTURY DANYCH

METODA REDUKCJI

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

-
-
1. [***] Podaj algorytm rozwiązujący *problem podziału* dla n -elementowej tablicy T względem zadanej wartości x . Algorytm ma tak przestawiać elementy w tablicy, aby podzielić ją na trzy fragmenty: na początku elementy mniejsze od x , potem elementy równe, a na końcu elementy większe od x . Podział tablicy będzie więc określony dwoma indeksami r i s ($0 \leq r \leq s \leq n-1$) w taki sposób, aby elementy mniejsze zostały wpisane do komórek $T[0 \dots r-1]$, elementy równe x do $T[r \dots s-1]$, a elementy większe do komórek $T[s \dots n-1]$. Twój algorytm powinien działać *w miejscu* (nie powinien używać żadnych tablic pomocniczych ani stosu), a jego czas działania być rzędu $\Theta(n)$. Uzasadnij jego poprawność. Przedstaw implementację tego algorytmu w pseudokodzie.
 2. [**] Podaj algorytm rozwiązujący *problem scalania* dwóch posortowanych n -elementowych tablic T_0 i T_1 . Ma on umieścić elementy z tych tablic w zewnętrznej $2n$ -elementowej tablicy w sposób posortowany. Czas działania algorytmu powinien być rzędu $\Theta(n)$. Uzasadnij jego poprawność. Przedstaw implementację tego algorytmu w pseudokodzie.
 3. [***] Dane są dwie posortowane n -elementowe tablice liczb oraz liczba całkowita k z zakresu $0 \leq k < 2n$. Opisz algorytm znajdowania k -tej co do wielkości liczby spośród liczb zapisanych w zadanych tablicach. Czas działania twojego algorytmu powinien być rzędu $O(\log n)$. Uzasadnij jego poprawność.
 4. [**] Niech dana będzie n -elementowa posortowana tablica liczb T oraz wartość liczbowa x . Zmodyfikuj algorytm wyszukiwania binarnego w taki sposób, aby wyznaczał on liczbę komórek w tablicy T , w których pamiętana jest wartość x .
 5. [****] Podaj algorytm wyznaczający jednocześnie element największy i drugi co do wielkości w zbiorze n -elementowym. Twój algorytm powinien wykonywać nie więcej niż $\lceil n + \log n - 2 \rceil$ porównań. Przedstaw jego implementację w pseudokodzie, zakładając że dane będą trzymane w tablicy. Twój algorytm nie powinien używać żadnych tablic pomocniczych, ale może wykorzystać stos na co najwyżej logarytmiczną liczbę rekurencyjnych wywołań jakiejś funkcji.
Wskazówka: Ideę działania algorytmu opisz na binarnym drzewie turniejowym.