

Algorytmy i Struktury Danych

egzamin poprawkowy, cz. II

wrzesień 2018

Zadanie 1. Mamy dostęp do tablicy $A[1 \dots n][1 \dots n]$, w której wszystkie wiersze i kolumny są posortowane niemalejąco, czyli $A[i][j] \leq A[i][j+1]$ dla każdego $i = 1, 2, \dots, n$ oraz $j = 1, 2, \dots, n-1$ oraz $A[i][j] \leq A[i+1][j]$ dla każdego $i = 1, 2, \dots, n-1$ oraz $j = 1, 2, \dots, n$. Konstruując odpowiedni algorytm, pokaż, że do znalezienia i, j , dla których $A[i][j] = x$ wystarczy odczytać $2n - 1$ pól tablicy. Podaj strategię adversarza, która wymusza odczytanie tylu pól.

Zadanie 2. Chcemy wykonać następujące operacje na zbiorze liczb:

- (a) utwórz zbiór zawierający podane n liczb,
- (b) dodaj x do aktualnego zbioru, oraz
- (c) usuń medianę/min/max aktualnego zbioru.

Podaj implementację, w której te operacje zajmują $\mathcal{O}(n)$ dla utworzenia nowego zbioru i $\mathcal{O}(\log n)$ dla pozostałych operacji, gdzie n jest rozmiarem aktualnego zbioru.

Zadanie 3. Dany jest ciąg liczb a_1, \dots, a_n , w którym występuje co najwyżej k różnych wartości. Pokaż, jak posortować go w czasie $\mathcal{O}(n \log k)$, używając tylko porównań dwóch liczb.

Zadanie 4 (dodatkowe). Ułóż algorytm, który dla danego ciągu $X = x_1, \dots, x_n$ oraz dla liczby naturalnej $k < 10$ znajduje jego podciąg $X' = x_{i_1}, \dots, x_{i_j}$ taki, że:

- (a) X' jest rosnący,
- (b) $i_{r+1} - i_r \leq k$ dla każdego $r = 1, \dots, j - 1$,
- (c) j jest możliwie największe.

Udowodnij poprawność swojego algorytmu. Podaj jego złożoność.