ASD 2020/2021 – Klasówka 1

(3 grudnia 2020)

Uwaga: uzasadnij poprawność swoich rozwiązań i dokonaj analizy złożoności obliczeniowej zaproponowanych algorytmów.

Zadanie 1 [8 punktów]

Na uporządkowanym rosnąco n-elementowym ciągu $x_1, x_2, \ldots, x_n$ dokonano dokładnie k zmian wartości elementów w tym ciągu, $0 \le k \le n$. Ciąg otrzymany w ten sposób nazywamy *k-zaburzonym*.

Przykład

Uporządkowany ciąg <2,6,8,12,21>. Po zmianach wartości elementów - 1-go na 7 i 4-go na 1 - dostajemy ciąg 2-zaburzony <7,6,8,1,21>.

Dana jest liczba całkowita k, $0 \le k \le n$, oraz k-zaburzony ciąg $x = <x_1, x_2, \ldots, x_n>$. Interesuje nas wydajne sortowanie ciągu x ze względu na porównania.

a) **[3 punkty]** Zaproponuj optymalne sortowanie ze względu na porównania dla k = 1 oraz
   a1) **[1 punkt]** n = 4
   a2) **[2 punkty]** n = 5
b) **[5 punktów]** Zaproponuj asymptotycznie optymalny algorytm sortowania k-zaburzonych ciągów n-elementowych. Pamiętaj o uwzględnieniu obu parametrów – n i k.

Zadanie 2 [4 punkty]

Niech A[1..4,1..n] będzie tablicą liczb całkowitych. Powiemy, że dwa elementy tablicy - A[i,j] oraz A[i',j'] - są *sąsiednie*, jeśli ich współrzędne różnią się o 1 na dokładnie jednej pozycji. Podzbiór elementów tablicy nazywamy *rozproszonym*, gdy nie ma w nim elementów sąsiednich. *Wagą* podzbioru elementów w tablicy nazywamy sumę zawartych w nim elementów. Zaprojektuj wydajny algorytm, który dla danej tablicy A znajdzie podzbiór rozproszony o największej wadze.

Przykład

Tablica z zaznaczonym pewnym zbiorem rozproszonym o wadze -2.

2	0	-10	2	7	6	6	4
3	3	5	-20	1	1	2	2
-1	-2	1	3	3	8	-5	3
6	4	2	4	5	7	9	7

Zadanie 3 [5 punktów]

Dany jest zbiór S zawierający n różnych punktów (x,y), gdzie $x \in \{0, 1, \ldots, n\}$, $y \in \{0, 1, \ldots, n\}$. Zaprojektuj wydajny algorytm znajdujący wszystkie punkty (a,b) w zbiorze S takie, że **każdy** punkt (c,d) z S różny od (a,b) ma mniejszą co najmniej jedną współrzędną, tzn. $a > c$ lub $b > d$.

Zadanie 4 [3 punkty]

Dany jest nieskierowany graf G = (V,E) z dodatnimi wagami na krawędziach i wyróżnionym wierzchołkiem źródłowym s. Zmodyfikuj algorytm Dijkstry w taki sposób, żeby dla każdego wierzchołka v wyznaczał wagę najlżejszej ścieżki z s do v oraz liczbę różnych ścieżek z s do v o tej wadze.