Examen - Structuri de Date
Seria 14

17 iunie 2021

In primul rand, va rog sa va scrieti NUMELE si GRUPA pe foaia de examen! Timpul de rezolvare este de 2 (doua) ore. Daca vom gasi asupra dumneavoastra telefoane mobile, laptopuri, tablete, fituici sau alte materiale ce contin informatii ajutatoare, veti fi scosi din sala de examinare. Daca aveti intrebari, ridicati mana si unul dintre instructori va veni la dumneavoastra in cel mai scurt timp.

Aveti 1 punct din oficiu :).

1 Exercitii foarte simple - (4 puncte)

1.1 1 punct (0,25 puncte pe exercitiu)

Exprimati functiile urmatoare in notatia Θ (scrieti doar raspunsul, fara demonstratii):

(a) $\lg(\sqrt{n})$

(b) $(n+2^2)^5$.

(c) $\lg n^4 + n^2$.

(d) $\lg n^4$.

1.2 1 punct

Construiti *suffix tree* si *suffix array* pentru urmatorul sir: *abracadabra*. Doar rezultatul final este suficient, fara pasi intermediari.

1.3 1 punct

Sa se deseneze arborele Huffman pentru literele urmatoare ce au frecventele: a = 5, b = 2, c = 4, d = 8, e = 16, f = 22, g = 40, h = 100.

Scrieti si codul optim (binar) pentru fiecare litera. Puteti desena pasii intermediari sau doar arborele final (cum doriti).

1.4 1 punct (0,5 puncte pe exercitiu)

Sa se construiasca arborele binar de cautare obtinut prin insertia urmatoarelor chei (doar arborele final, fara pasi intermediari). Apoi, sa se extraga radacina si sa se deseneze arborele rezultat: 7, 1, 2, 5, 8, 10, 12, 9, 22, 16, 4.

2 Exercitii simple - (2 puncte)

2.1 1 punct

Demonstrati (folosind definitiile) urmatoarea propozitie.

Daca $f(n) \in \Omega(g(n))$ si $g(n) \in \Omega(h(n))$, atunci $f(n) \in \Omega(h(n))$.

2.2 1 punct

Rezolvati recurenta $T(n) = 3T(n-1) + 5$. Demonstrati prin inductie ca rezultatul este corect. Arborele de recurenta nu se considera demonstratie.

3 Exercitiu usor - (3 puncte)

3.1 1.5 puncte

Se dau doua siruri t_1 si t_2 cu n, respectiv m caractere. Sa se gaseasca cel mai lung subsir comun al celor doua siruri (subsir inseamna caractere consecutive). Demonstrati corectitudinea si timpul de rulare al algoritmului. Un algoritm de complexitate $O(n \cdot m)$ va primi $0,75$ puncte, iar un algoritm de complexitate $O(n+m)$ va primi $1,5$ puncte. Bonus de $0,5$ puncte daca puteti generaliza algoritmul pentru k siruri de intrare $t_1, t_2, \ldots, t_k$.

Exemplu: $t_1 = abbbac$; $t_2 = babbac$. Cel mai lung subsir comun este *bbac* si are lungime 4.

3.2 1.5 puncte

k quantilele unui sir de n-elemente sunt cele k valori din sir, care impart sirul sortat in $k+1$ multimi egale (+/- un element). Construiti un algoritm care sa gaseasca k cuantilele unui sir de n elemente dat. Daca sunt mai multe raspunsuri valide, algoritmul poate returna pe oricare din ele. Demonstrati corectitudinea algoritmului. Pentru un algoritm de complexitate $O(n \log n)$ veti primi $0,25$ puncte. Pentru un algoritm de complexitate $O(n \cdot k)$ veti primi $0,75$ puncte, iar pentru un algoritm de complexitate $O(n \log k)$ veti primi punctajul intreg.

Exemplu: n = 7, k = 2 si sirul 2 1 3 6 5 8 4 7 9, solutii posibile ar fi (in sirul sortat quantilele sunt marcate cu bold):

$(3,6)$; 1 2 **3** 4 5 **6** 7 8 9
$(3,7)$; 1 2 **3** 4 5 6 **7** 8 9
$(4,7)$; 1 2 3 **4** 5 6 **7** 8 9