

Tema 8
27 noiembrie 2018

Probleme obligatorii

Termen de predare : Laboratorul din săptămâna 11 (10 - 15 decembrie 2018)

1. Arbori binari de căutare

(10 p) **1.** Să se implementeze un *arbore binar de căutare echilibrat AVL* cu următoarele operații (cu echilibrare după fiecare operație, acolo unde este necesar):

- (a) *adauga* (*t*, *x*) - insereaza cheia *x* în arborele de rădăcină *t*;
- (b) *cauta* (*t*, *x*) - întoarce 1 dacă elementul *a* se afla în arborele de rădăcină *t* și 0 în caz contrar;
- (c) *maxim* (*t*) - întoarce elementul maxim din arborele de rădăcină *t*, fără a-l șterge din arbore;
- (d) *sterge* (*t*, *x*) - șterge în arborele de rădăcină *t* nodul cu cheia *x* (păstrând proprietatea de arbore binar de căutare și, eventual, echilibrarea);
- (e) *afisează* (*t*) – afișează cheile din arborele de rădăcină *t*, în ordine crescătoare.

2. Sortare

(1 p) **2.** Să se implementeze metoda de ordonare Merge – Sort.

Probleme suplimentare

Termen de predare : Laboratorul din săptămâna 11 (10 - 15 decembrie 2018)

(1 p) **3.** Să se ordoneze descrescător un șir de cuvinte date de la tastatură, folosind un arbore binar de căutare.

(2 p) **4.** Să se optimizeze algoritmul de bază al metodei de *sortare prin interclasare* prin utilizarea inserției directe la sortarea subșirurilor mici (mai mici de 10 elemente).

(2 p) **5.** Fie două secvențe sortate care împart același tablou și sunt poziționate astfel: prima crescând urmată de cealaltă descrescând, sau prima descrescând urmată de cealaltă crescând (secvență bitonică). Se cere să se sorteze prin *interclasare* tabloul respectiv.

(3 p) **6. Bitone**

O secvență de numere întregi se numește **bitonă** dacă este crescătoare la început, iar apoi descrescătoare. Mai precis, o secvență a_1, a_2, \dots, a_n este bitonă dacă:

- este o secvență nedescrescătoare: $a_1 \leq a_2 \leq \dots \leq a_n$ **sau**
- este o secvență necrescătoare: $a_1 \geq a_2 \geq \dots \geq a_n$ **sau**
- exista un indice i pentru care $a_1 \leq a_2 \leq \dots \leq a_i \geq a_{i+1} \geq \dots \geq a_n$

Cerință

Dată o secvență de numere întregi a_1, a_2, \dots, a_n și niște întrebări de forma (i, j) să se răspundă pentru fiecare întrebare dacă subsecvența a_i, a_{i+1}, \dots, a_j este bitonă.

Date de intrare

Fișierul de intrare `bitone.in` conține pe prima linie numărul de numere din secvență, n . Pe a doua linie conține cele n numere ale secvenței, separate de spații. Pe a treia linie se află numărul de întrebări q . Pe următoarele q linii se vor găsi câte două numere i, j separate prin spațiu, reprezentând întrebările la care se cere răspuns.

Date de ieșire

Fișierul de ieșire `bitone.out` va conține o singură linie cu q caractere 0 sau 1, fără spații între ele, caractere ce reprezintă răspunsurile la întrebări. Pentru fiecare întrebare veți răspunde 1 dacă subsecvența este bitonă, sau 0 în caz contrar.

Restricții

- $1 \leq n \leq 1.000.000$
- $-2.000.000.000 \leq a_i \leq 2.000.000.000$
- $1 \leq q \leq 1.000.000$
- $1 \leq i \leq j \leq n$

Exemple

bitone.in	bitone.out	Explicație
10 10 19 19 18 18 21 21 11 11 13 6 9 10 6 10 4 8 8 10 1 7 3 3	101101	subsecvențele (6, 10) și (1, 7) nu sînt bitone. Toate celelalte sînt.

15 10 11 13 13 6 8 8 8 4 4 5 9 0 2 2 10 2 10 9 13 1 3 7 14 4 7 1 9 3 10 4 11 13 13 9 9	0110000011	subsecvențele (9, 13) (1, 3) (13, 13) și (9, 9) sînt bitone. Toate celelalte nu.
--	------------	--

Cerc informatică Vianu

Probleme facultative

Termen de predare : Laboratorul din săptămîna 10 (3 - 8 decembrie 2018)

(5 ps) 1. Fiind dată o tablă de șah de 8 x 8 pătrate, putem să o tăiem în două trapeze și două triunghiuri, ca în imaginea din stînga. O reasamblăm apoi după cum este indicat în figura din dreapta. Aria tablei din stînga este $8 \times 8 = 64$, pe cînd aria tablei din dreapta este $13 \times 5 = 65$. Explicați paradoxul.

