

# Structuri de date și algoritmi

## - examen scris -

### Notă

1. Subiectele se notează astfel: of - 1p; A – 2p; B - 1.5p; C1 - 1p; C2 – 1p; D - 3.5p.
2. Pentru cerința A, justificarea unei complexități presupune deducția acesteia.
3. Pentru cerințele B și C (C1, C2) se cer justificări, care vor fi punctate.
4. Problema de la D se va rezolva în Pseudocod. Se cer și se vor puncta: (1) descrierea ideii de rezolvare și comentarii despre soluția propusă; (2) scrierea reprezentării indicate în enunț; (3) (specificare și) implementare subalgoritm(i); (4) complexitate.

**Nu se acceptă cod C++. Nu se acceptă pseudocod fără comentarii despre soluția propusă.**

**A.** Scrieți un subalgoritm recursiv având complexitatea timp  $\theta(n^3)$ . Justificați complexitatea (prin deducția acesteia).

**B.** Este vectorul în care elementele se succed în ordine crescătoare un ansamblu? Justificați.

C. În implementarea Stivei folosind o listă înlănțuită, ce operații necesită timp liniar în cazul defavorabil? Justificați

a) vidă      b) accesare (operația element)      c) ștergere      d) adăugare      e) niciuna din operațiile menționate

dacă în reprezentare avem alocat un pointer pentru ultimul elem atunci toate op au complexitatea  $O(1)$ , altfel complex este  $O(n)$  (mai puțin pt vidă) pentru că trebuie parcursă lista până la final

C. Considerăm expresia în forma infixată:  $4+3*(6*3-12)$ . Presupunem că folosim o Stivă pentru a converti expresia din forma infixată în forma postfixată. Care este numărul maxim de simboluri care vor apărea în stivă la un moment dat de-a lungul conversiei? Justificați      a) 1      b) 2      c) 3      d) 4      **e) 5**

output: 4 3 6 3 12

stiva: + \* ( \* -

la întâlnirea parantezei ")" încep să se scoată elem => 4 3 6 3 12 - \* \* +

**D.** Descrieți operația de singură rotație spre dreapta pentru reechilibrare într-un Arbore Binar de Căutare. Arborele se reprezintă secvențial, pe vector, folosind ca schemă de memorare ansamblul. Indicați grafic situația de rotație, reprezentarea arborelui și descrieți în Pseudocod subalgoritmul. Precizați complexitatea operației. Folosiți comentarii pentru a ușura înțelegerea soluției.