

Structuri de date și algoritmi

- examen scris -

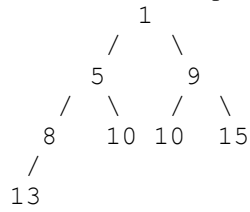
Notă

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.

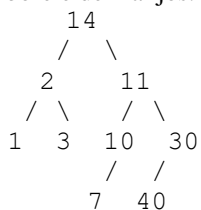
Nu se acceptă cod C++. Nu se acceptă pseudocod fără comentarii despre soluția propusă.

A. Scrieți un subalgoritm iterativ care să aibă timpul de execuție $\theta(n^2 \cdot \log_2 n)$. Deduceți complexitatea. Nu se va folosi funcția matematică **logarithm**.

B. Arătați ansamblul rezultat prin inserarea valorii 3 în următorul ansamblu. Justificati



C. Fie arborele de mai jos.



Care este inordinea arborelui? Justificati

a) 1 2 3 7 10 11 14 30 40

b) 1 2 3 14 7 10 11 40 30

c) 1 3 2 7 10 40 30 11 14

d) 14 2 1 3 11 10 7 30 40

C. Prespunem că T este un arbore binar cu 14 noduri. Care este adâncimea minimă a lui T ? Justificati

a) 0

b) 3

c) 4

d) 5

D. Fie containerul **CP2** similar cu **Coadă cu priorități**, exceptând faptul că vrem să accesăm și să ștergem **al doilea cel mai prioritar element** în raport cu o relație de ordine \mathcal{R} între priorități (dacă $\mathcal{R}=\leq$, atunci elementul cel mai prioritar este **minimul**). Specificați și implementați operația de **ștergere** din **CP3**, folosind un ansamblu *ternar* (în loc de 2 descendenți, vor fi 3) pentru reprezentarea containerului. Se va indica reprezentarea folosită și se va preciza complexitatea operației. Folosiți comentarii pentru a ușura înțelegerea soluției.