

Structuri de date și algoritmi

- examen scris -

Notă

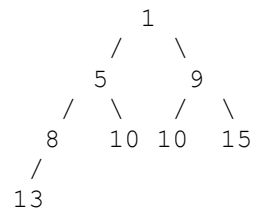
1. Subiectele se notează astfel: of - 1p; A - 2p; B - 1.5p; C1 - 1p; C2 - 1p; D - 3.5p.
2. Pentru cerința A, justificarea unei complexități presupune deducția acesteia.
3. Pentru cerințele B și C (C1, C2) se cer justificări, care vor fi punctate.
4. Problema de la D se va rezolva în Pseudocod. Se cer 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. Deduceți timpii mediu si defavorabil pentru următorul subalgoritm. Justificați rezultatul.

```
Funcția f(n, i) este { :Intreg }
|   { pre: n, i: Intreg }
|   dacă n > 1 atunci
|       m ← [n/2]; S ← 2 * f(m, i-1); j ← 1; gata ← adevărat
|       cattiimp j ≤ n * n și gata executa
|           dacă i mod 2 = 1 atunci gata ← fals
|           altfel j ← j + 1
|           sfdacă
|               sfcattiimp
|               f ← S + 2 * f(m, i+1)
|       altfel f ← 0
|   sfdaca
sff
```

B. Arătați ansamblul rezultat în urma aplicării operației de ștergere din următorul ansamblu. Justificați



- C. Presupunem o Stivă implementată secvențial pe un vector, având 10 elemente stocate de la `data[0]` până la `data[9]`. Capacitatea maximă a stivei este 42. Vrem sa adăugăm în stivă un nou element. Unde se va adăuga acest element în vector? Justificati
- a) `data[0]`
 - b) `data[1]`
 - c) `data[9]`
 - d) `data[10]`

C. Care dintre următoarele operații pe o Coadă pot cauza *depășire superioară*? Justificați

a) vidă

b) șterge

c) adaugă

D. Implementați în Pseudocod operația de **adăugare** într-un ABC. Arborele se reprezintă înlănțuit, cu înlănțuirile reprezentate pe tablou. Se va folosi o procedură nerecursivă. Se va indica reprezentarea și se va preciza complexitatea operației. Folosiți comentarii pentru a ușura înțelegerea soluției.