

Universitatea din București
Facultatea de Matematică-Informatică
Structuri de Date
Examen, 12 Iunie, Nivelul I, Subiecte: A

Nume: _____

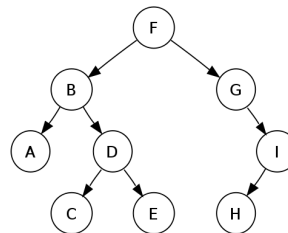
Puncte:25. Timp: 50 min

Data: 12-06-2023

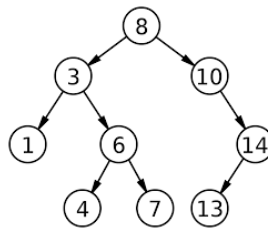
Instrucțiuni I. Incercuiți răspunsul/răspunsurile corecte. Toate întrebările contează în mod egal. Puteți folosi marginile hârtiei drept ciornă, **dar niciun alt material**. O întrebare poate avea mai multe răspunsuri corecte. Toate contează în mod egal. **Pe de altă parte, dacă alegeți un răspuns greșit, punctajul vostru la întrebare este zero.**

1. Care e înălțimea maximă a unui arbore AVL cu 4 noduri? Presupunem că înălțimea unui arbore cu un nod este 0.
 - (a) 1
 - (b) 2
 - (c) 3
 - (d) Răspunsul corect este altul.
2. Traversarea în preordine a unui arbore binar de căutare este 30, 20, 10, 15, 25, 23, 39, 35, 42. Care din următoarele este traversarea în postordine pentru același arbore ?
 - (a) 10, 20, 15, 23, 25, 35, 42, 39, 30
 - (b) 15, 10, 25, 23, 20, 42, 35, 39, 30
 - (c) 15, 20, 10, 23, 25, 42, 35, 39, 30
 - (d) 15, 10, 23, 25, 20, 35, 42, 39, 30
3. Pentru a efectua o ștergere prin copiere (delete by copy) într-un arbore binar de căutare trebuie să găsim succesorul (în inordine) al unui nod cu doi copii. Care din următoarele afirmații este adevărată ?
 - (a) Succesorul este întotdeauna un nod frunză.
 - (b) Succesor este întotdeauna fie un nod frunză, fie un nod fără copil stâng .
 - (c) Succesorul poate fi un strămoș al nodului.
 - (d) Succesor este întotdeauna fie un nod frunză, fie un nod fără copil drept.
4. Următoarele numere sunt inserate succesiv într-un arbore binar de căutare gol: 10, 1, 3, 5, 15, 12, 16. Care este înălțimea arborelui la final (the height is the maximum distance of a leaf node from the root)?
 - (a) 2
 - (b) 3.
 - (c) 6
 - (d) 4.

5. Să considerăm următoarele elemente: (4322, 1334, 1471, 9679, 1989, 6171, 6173, 4199) și funcția hash $h(x) = x \bmod 10$. Care din următoarele afirmații sunt adevărate ? (i) 1471, 6171 produc o coliziune (ii) 9679, 1989, 4199 se mapează la aceeași valoare. (iii). Toate elementele sunt mapate la aceeași valoare (iv.) Fiecare element este mapat pe o valoare distinctă.
- (a) doar (i).
- (b) doar (ii).
- (c) doar (i) și (ii).
- (d) (iii) sau (iv).
6. Care din următoarele secvențe **nu** este una din traversările în preordine, înordine, postordine ale arborelui din figura de mai jos ?

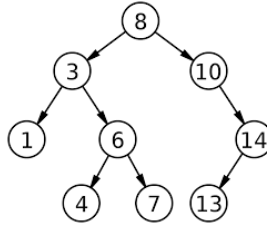


- (a) A,B,C,D,E,F,G,H,I
- (b) F,B,D,A,C,E,G,H,I
- (c) F,B,G,A,D,I,C,E,H
- (d) Niciun șir de mai sus nu este un răspuns corect
7. Care nod(uri) este/sunt în mod sigur colorat(e) în roșu în arborele de mai jos dacă știm că trebuie să fie un arbore red-black ? Nodurile sentinelă nu sunt reprezentate.



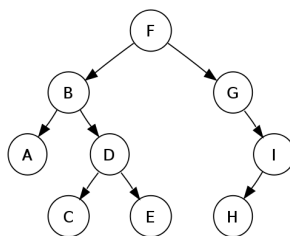
- (a) 1
- (b) 6
- (c) 8
- (d) 13
8. De ce adăugăm câmpul *tree-size* la un arbore "order-statistics" ?
- (a) Pentru a calcula în mod eficient numărul de noduri în arbore.
- (b) Pentru a face operațiile să ruleze în timp $O(\log n)$
- (c) Pentru că putem calcula *tree-size* în $O(1)$ din valorile câmpului *tree-size* al nodurilor copii.
- (d) Niciunul din celelalte răspunsuri nu este corect.

9. Să considerăm următorul arbore binar de căutare. După ce am rotit nodul 6 în jurul lui 3 ...



- (a) Nodul 1 este copil al nodului 3
 - (b) Node 4 este copil al nodului 3
 - (c) Mărimea subarborelui nu se schimbă, dar rădăcina sa da.
 - (d) Niciunul din celelalte răspunsuri nu este corect.
10. Care din următoarele afirmații sunt adevărate? Intr-o listă skip:
- (a) probabilitatea ca un nod să aibă cel puțin doi pointeri este exact $1/4$.
 - (b) Elementele sunt sortate în ordine crescătoare.
 - (c) Nivelele sunt spațiate în mod egal
 - (d) Niciunul din celelalte răspunsuri nu este corect.
11. Care din următoarele secvențe de operații este **imposibilă** într-o stivă cu trei elemente ?
- (a) PUSH, POP, POP, POP, POP, PUSH
 - (b) PUSH, POP, POP, POP, PUSH, POP, POP
 - (c) PUSH, POP, POP, POP, POP, POP, PUSH, POP
 - (d) PUSH, PUSH, PUSH, PUSH, PUSH, PUSH
 - (e) POP, POP, POP, POP, POP, POP, POP
12. Când numărul de slot-uri într-o tabelă de dispersie (hash table) se dublează, ce se întâmplă cu încărcarea tabelului (load) ?
- (a) se dublează
 - (b) se înjumătățește
 - (c) rămâne la fel.
 - (d) niciun alt răspuns nu este corect.

13. Presupunând că arborele de mai jos e unul de căutare, unde se află elementul median ?



- (a) F
 - (b) D
 - (c) E
 - (d) G
14. Când calculăm înfășurătoarea convexă a n puncte ...
- (a) folosim o coadă
 - (b) sortăm punctele după coordonata Oy
 - (c) sortăm punctele după coordonata Ox
 - (d) niciun alt răspuns nu este corect
15. Să presupunem că modificăm algoritmul de parcurgere în lățime (breadth-first search) a unui **arbore binar** în felul următor; în loc de o coadă folosim o coadă dublă (deque); când scoatem primul nod din coadă îl vizităm mai întâi, iar vecinii nodului scos îi adăugăm **în vârful cozii duble**. Modificarea astfel descrisă este echivalentă cu o parcurgere a arborelui
- (a) în preordine
 - (b) în inordine
 - (c) în postordine
 - (d) în adâncime
 - (e) niciunul din celelalte răspunsuri nu este corect.
16. Vrem să implementăm următoarea operație pe o mulțime dinamică de elemente: $QUERY(a,b)$ returnează numărul de elemente ale lui S în intervalul $[a, b]$. Vrem ca operația să aibă complexitate $O(\log(n))$. De la ce structură de date ar fi cel mai bine să pornim ?
- (a) Vector
 - (b) Hash table.
 - (c) splay tree
 - (d) order-statistics tree
 - (e) niciuna din alegerile de la celelalte puncte nu este bună.

17. Să presupunem că modificăm o listă skip ca să putem face salturi **înainte și înapoi**, folosind intuitiv liste dublu (în loc de simplu) înlanțuite pe fiecare nivel. Ne vom limita la a implementa o listă cu **patru nivele de pointeri**. Nivelul unui nod va fi ales exact ca la un skip list obișnuit. Numărul total mediu de pointeri în varianta noastră este ...
- (a) $\Theta(n)$.
 - (b) $\Theta(n \log(n))$.
 - (c) $\Theta(n^2)$.
 - (d) Niciunul din celelalte răspunsuri nu este corect.
18. Să considerăm o schemă de double hashing care mapează elementele unui univers U pe o mulțimea $0, 1, 2, \dots, m-1$ via funcția $h(x, i) = (h_1(x) + i \cdot h_2(x)) \pmod{m}$ unde m este mărimea tabelului iar h_1, h_2 funcții hash "simple". Să considerăm funcțiile $H_1(x, i) = (h_2(x) + i h_1(x)) \pmod{m}$, $H_2(x, i) = (h_2(x) - 1 + i(h_1(x) + 1)) \pmod{m}$. Care dintre funcțiile H_1, H_2 sunt potrivite, în principiu, în loc de h pentru double hashing ?
- (a) H_1 dar nu și H_2
 - (b) H_2 dar nu și H_1
 - (c) Și H_1 , și H_2 .
 - (d) Nici H_1 , nici H_2 .
19. Vrem să reprezentăm mulțimea $S = \{1, 2, 3\}$ cu un red-black tree. În câte moduri putem face acest lucru? Doi arbori pot diferi fie prin formă fie (dacă au aceeași formă) prin culoare.
- (a) 1
 - (b) 2
 - (c) 3
 - (d) 8
 - (e) Răspunsul corect nu se află printre alternativele de mai sus.
20. Vrem să reprezentăm mulțimea $S = \{1, 2, 3\}$ cu un arbore AVL. În câte moduri diferite putem face acest lucru ?
- (a) 1
 - (b) 2
 - (c) 3
 - (d) 8
 - (e) Răspunsul corect nu se află printre alternativele de mai sus.

21. O listă skip reprezintă mulțimea de elemente 1,2,3,5,8,13,21,44. În al câatâlea nod vom găsi elementul 5 ?
- (a) 1
 - (b) 2
 - (c) 4
 - (d) 8
 - (e) altă valoare.
22. Fie x o cheie într-un B -tree. Unde se poate afla succesorul lui x (presupunem că există) ?
- (a) În același nod cu cheia originală
 - (b) Într-un descendent al nodului în care se află cheia
 - (c) Într-un nod strămoș al nodului în care se află cheia
 - (d) Mai există și alte alternative neluate în calcul.
23. Când ruăm algoritmul heapsort, numărul maxim de swap-uri de elemente se atinge
- (a) când secvența inițială este sortată crescător.
 - (b) când secvența inițială este sortată descrescător
 - (c) când secvența este una aleatoare
 - (d) răspunsul corect nu apare mai sus
24. În câte moduri putem pune numerele 1,2,3, 4 într-un vector astfel încât vectorul rezultat să poată fi văzut drept un min-heap ?
- (a) 2
 - (b) 3
 - (c) 7
 - (d) altă variantă
25. Ce face următorul cod ?
- ```

void fun(struct node *head){
if (head==NULL)
 return;
fun(head->next);
cout << head->data;
}

```
- (a) printează toate nodurile listei
  - (b) printează toate nodurile liste în ordine inversă
  - (c) printează nodurile cu index par ale listei (Notă: index-ul minim e 0).
  - (d) printează nodurile cu index par ale listei în ordine inversă

# Answer Key for Exam A

Instrucțiuni I. Incercaiți răspunsul/răspunsurile corecte. Toate întrebările contează în mod egal. Puteți folosi marginile hârtiei drept ciornă, **dar niciun alt material**. O întrebare poate avea mai multe răspunsuri corecte. Toate contează în mod egal. **Pe de altă parte, dacă alegeți un răspuns greșit, punctajul vostru la întrebare este zero.**

1. (b)
2. (d)
3. (b)
4. (b)
5. (c)
6. (b), (c)
7. (d)
8. (b)
9. (a), (b) , (c)
10. (b)
11. (c), (e)
12. (b)
- 13.
14. (d)
15. (a)
16. (d)
17. (a)
18. (d)
19. (b)
20. (a)
21. (c)
22. (a), (b) , (c)
23. (b)
24. (b)
25. (b)