

NUME:

PRENUME:

GRUPA:

	Ex1	Ex2	Ex3	Ex4
a				
b				
c				
d				

Punctajele se acorda astfel: a=1p, b=2p, c=3p, d=4p, total $4 \times (1+2+3+4) = 40p$

Punctajul pentru d se acorda doar daca ati rezolvat corect punctul c corespunzator

Acolo unde nu se precizeaza altceva, structurile de date sunt generice

1. Lista si Dictionar

- Care sunt tipurile de liste inlantuite studiate la curs? Dati un exemplu pentru fiecare si explicati ce pointeri ati folosi pentru implementarea fiecareia.
- Se da urmatoarea functie, din care lipseste un rand. Ce ar trebui scris in loc de comentariul "Rand lipsa" pentru ca functia sa inverseze corect o lista simplu inlantuita:

```
struct node
{
    int data;
    struct node* next;
};

/* head_ref este o referinta la pointerul head */
static void reverse(struct node** head_ref)
{
    struct node* prev = NULL;
    struct node* current = *head_ref;
    struct node* next;
    while (current != NULL)
    {
        next = current->next;
        current->next = prev;
        prev = current;
        current = next;
    }
    /***** Rand lipsa *****/
}
```

- Consideram clasa LinkedList pentru liste dublu inlantuite. Scrieti metoda de adaugare de elemente la sfarsitul listei pentru aceasta clasa.
- Consideram clasa Hashtable pentru dictionar ce trateaza coliziunile prin inlantuire directa. Clasa foloseste un vector de LinkedList (clasa ce se considera deja implementata, prin efortul vostru de la punctul precedent). Scrieti: declaratia vectorului, metoda de adaugare a unui element in dictionar si functia de hash. Care sunt criteriile pentru care ati ales aceasta functie de hash?

2. Arbori Binari si Arbori Binari de Cautare

- Ce este un arbore? Dar un arbore binar?
- Un arbore binar complet are n noduri interne. Cate frunze are? De ce?
- Fie clasa `BinarySearchTree` pentru arbore binar de cautare. Implementati metoda de inserare a unui element pentru aceasta clasa. Explicati (prin desene) procesul de adaugare a urmatoarelor elemente: 13, 10, 7, 23, 9, 18, 8, 16, 25, 4, 17.
- Pornind de la clasa mentionata la punctul precedent, explicati doua mecanisme prin care se poate obtine echilibrarea arborelui binar de cautare. Scrieti cod care sa puna in evidenta una dintre ele.

3. Heap si PriorityQueue

- Explicati cum functioneaza un heap si la ce poate fi folosit
- Fie clasa `MaxHeap`, implementata folosind un vector. Considerand ca vectorul contine elementele 25,14,16,8,12,13,10, care va fi continutul vectorului dupa doua operatii de extragere?
- Implementati pentru clasa de la punctul precedent operatiile de inserare si extragere, impreuna cu metodele pe care le folosesc acestea.
- Folositi clasa `MaxHeap` de la punctul anterior pentru a implementa un `PriorityQueue`.

4. Graf

- Ce puteti spune despre graful de mai jos? Este orientat sau neorientat? Cate componente tare-conexe are si care sunt acestea?

	V1	V2	V3	V4	V5
V1	0	1	0	1	0
V2	0	0	1	0	1
V3	0	0	0	0	1
V4	0	0	0	0	1
V5	0	1	0	0	0

```
void functia_misterioasa(int S) {
    Queue<int> Q;
    int x, y;
    Q.enqueue(S);
    visited[S] = 1;

    while (!Q.isEmpty()) {
        x = Q.dequeue();
        for (y = 0; y < g.N; y++)
            if (g.A[x][y] && !visited[y]) {
                visited[y] = 1;
                Q.enqueue(y);
            }
    }
}
```

- Ce face codul de mai sus? Explicati functionarea sa. Ce rezultat produce functia pentru graful de mai sus, considerand ca parametrul S va indica pozitia nodului V1.
- Implementati o functie care sa realizeze parcurgerea DFS. Ce rezultat va produce pentru graful din figura de mai sus?
- Modificati functia de la punctul precedent pentru a realiza o sortare topologica. Ce va obtine functia pentru graful de mai sus? Dati 3 exemple din viata reala in care puteti folosi aceasta functie.