

COLOCVIU LA DISCIPLINA "PROGRAMARE AVANSATĂ PE OBIECTE" – SESIUNEA MAI/IUNIE 2023 –

I. Pentru fiecare dintre cele 5 întrebări de mai jos, indicați varianta de răspuns pe care o considerați corectă:

1. Fie următorul program:

```
public class Test {  
    public static void main(String[] args) {  
        TreeMap m = new TreeMap();  
        m.put("c", "JavaSE");  
        m.put("a", null);  
        m.put("b", "JavaSE");  
        m.put("c", "Python");  
        m.put("a", "C++");  
        System.out.println(m);}  
}
```

După executarea sa, se va afișa:

- a) {a=null, b=JavaSE, c=JavaSE}
- b) {c=Python, a=C++, b=JavaSE}
- c) {a=C++, b=JavaSE, c=Python}
- d) Excepție de tip NullPointerException la executare

2. Fie următorul program Java:

```
class A {  
    int x = 0;  
    public A(int n) { x = n; }  
}  
  
class B extends A {  
    int x = 1;  
    public B(int n) { super(n); }  
}  
  
public class Test {  
    public static void main(String[] args) {  
        A a = new A(5);  
        B b = new B(7);  
        System.out.println(a.x + " " + b.x);  
    }  
}
```

După executarea programului, se va afișa:

- a) 0 5
- b) 5 1
- c) 5 7
- d) 0 1

3. Fie următorul program Java:

```
class Test {  
    public static void main(String[] args) {  
        try {  
            int a[] = {1, 2, 3, 4};  
            for (int i = 1; i <= 4; i++)  
                System.out.print(a[i] / (3 - i) + " ");  
        } catch (ArrayIndexOutOfBoundsException e) {  
            System.out.print("E1 ");  
        } catch (Exception e) {  
            System.out.print("E2 ");  
        }  
    }  
}
```

După executarea programului, se va afișa:

- a) 0 1 3 E2 b) 1 3 E2 c) 1 3 E2 E1 d) 1 3 E2 -4 E1

4. Fie următorul program Java:

```
public class Test {  
    public static void main(String[] args) {  
        List<Integer> numere = new ArrayList<Integer>();  
        for (int i = 0; i < 11; i++)  
            numere.add(i);  
        Iterator<Integer> itr = numere.iterator();  
        while (itr.hasNext()) {  
            Integer nr = itr.next();  
            if (nr % 2 == 0)  
                itr.remove();  
        }  
        System.out.println(numere);  
    }  
}
```

Care dintre următoarele afirmații este adevărată?

- a) Programul va afișa doar numere impare.
b) Unei colecții de tip List nu i se poate atașa un iterator.
c) Apelul metodei remove va genera o excepție de tip ConcurrentModificationException la executare.
d) Apelul metodei remove va genera o excepție de tip ConcurrentModificationException la compilare.
5. Care dintre următoarele variante reprezintă o supraîncărcare corectă (overloading) pentru metoda protected int getNota(String curs)?
- a) protected int getNota (String curs) throws IOException
b) private int getNota (String curs)
c) protected long getNota (String curs)
d) public long getNota (int IDStudent)

2. Se consideră definită o clasă *Carte*, având datele membre *titlu*, *autor*, *editura*, *nrExemplare* și *pret*. Clasa este utilizată pentru a memora informații despre cărțile aflate într-o librărie. Datele membre *titlu*, *autor*, *editura* sunt de tip *String*, data membră *nrExemplare* este de tip *int*, iar data membră *pret* este de tip *double*. Clasa încapsulează constructori, metode de tip *set/get* pentru toate datele membre, precum și metodele *toString()*, *equals()* și *hashCode()*. Creați o listă care să conțină cel puțin 3 obiecte de tip *Carte* și, folosind *stream-uri* bazate pe lista creată și *lambda expresii*, rezolvați următoarele cerințe:
- afișați cărțile din care există cel mult 10 exemplare, în ordinea alfabetică a autorilor;
 - afișați editurile distincte;
 - creați o colecție care să conțină cărțile de la editura ABC cu prețul cuprins între 50 RON și 100 RON;
 - afișați pentru fiecare autor lista cărților publicate.
3. Informațiile despre cărțile aflate în lanțul de librării CLB sunt păstrate în mai multe fișiere text. Fiecare linie dintr-un astfel de fișier conține informații referitoare la o carte, sub forma *titlu,autor,editura,numar_exemplare,pret*. Scrieți o clasă Java care să calculeze, pe baza informațiilor dintr-un fișier de tipul indicat anterior, valoarea stocului unei cărți pentru care se cunosc datele *titlu* și *autor*, folosind un fir de executare dedicat. Scrieți un program care, utilizând clasa definită anterior, citește de la tastatură titlul și autorul unei cărți, după care afișează valoarea totală a stocurilor cărții respective existente în două librării, pe baza informațiilor din fișierele text *librarieCLB_1.txt* și *librarieCLB_2.txt*. Datele de intrare *autor* și *titlu* sunt de tip *String*. Stocul unei cărți este egal cu produsul dintre numărul de exemplare și prețul său.
4. Se consideră definită complet o clasă *Persoana* care permite memorarea următoarelor informații despre o persoană: *nume* (șir de caractere), *vârsta în ani* (număr natural) și *venitul anual* (număr real). Definiți complet o clasă singleton denumită *CitireScrierePersoane* care să permită citirea/scrierea informațiilor despre mai multe persoane dintr-un/într-un fișier text de tip CSV, respectiv informațiile despre o persoană se vor citi/scrie de pe/pe o linie, despărțite între ele prin câte o virgulă. Informațiile despre persoane se vor citi din fișierul text într-un obiect *ArrayList<Persoana>*, respectiv se vor scrie în fișierul text dintr-un obiect de tip *ArrayList<Persoana>*. Numele fișierului text de tip CSV și referința spre obiectul de tip *ArrayList<Persoana>* se vor transmite ca parametrii ai constructorului clasei *CitireScrierePersoane*.

NOTĂ:

- Datele de intrare se consideră corecte.
- Nu se vor trata excepțiile.
- Punctaj: 2.5p. (5 x 0.5p.) + 2.5p. + 2p. + 1p. (din oficiu)