

# Examen PAO

## DESFASURAREA EXAMENULUI

Intre 12:00-12:30 se va face prezenta prin verificarea cartii de identitate/ carnetului de student in fata camerei web.

La ora 12:30 fiecare student va primi pe emailul de tipul @s.unibuc.ro linkul catre formularul de examen. Timpul de lucru este 90 de minute => 12:30 - 14:00.

Formularul de examen se dezactiveaza automat la ora 14:15, cu 15 minute rezervate pentru trimiterea solutiilor.

## DESPRE COMPLETARE

Va rog sa completati cu atentie numele si grupa.

Pentru fiecare întrebare este obligatoriu sa bifați cel puțin un răspuns.

Exista 11 întrebări cu răspuns multiplu si 7 întrebări cu răspuns simplu.

## DESPRE EVALUARE

Atentie, punctajul pe care il primiti nu va contine punctul din oficiu, iar nota finala se va stabili ulterior, dupa ce voi lua in calcul si notele obtinute la laborator!

De asemenea Microsoft Forms nu acorda corect punctaj partial, deci cel mai probabil punctajul va creste in urma verificarii manuale. Mai exact, automat se primesc 0 puncte daca nu selectati exact optiunile corecte, dar daca veti selecta de exemplu 1 din cele 2 optiuni corecte veti primi punctaj partial in urma verificarii manuale.

## IMPORTANT

Pentru a primi pe email rezultatul testului online, trebuie sa bifati optiunea "Send me an email receipt of my responses" aflata deasupra butonului "Submit".

Pe tot parcursul examenului studentii trebuie sa stea conectati in meeting, iar dupa trimiterea raspunsurilor, verificati ca ati primit confirmarea pe mail si apoi deconectati-va din meeting.

Points: **65/90**

### 1. Grupa \*

- ☒ 241
- ☐ 242
- ☐ 243
- ☐ 244

## 2. Nume Prenume \*

Fusneica Florentin-Cristian

## 3. Fie urmatorul cod:

```
public class A {  
    Integer intA;  
    public A(Integer i) {intA = i;}  
    public void set(Integer i) {intA = i;}  
    public A(A other) {intA = other.intA;}  
}  
  
public class B{  
    A obiect;  
    public B(A init) {  
        obiect = new A(init);  
    }  
    public B(B other) {obiect = other.obiect;}  
    public A getA() {  
        return new A(obiect);  
    }  
}  
  
public class C{  
    B obiect;  
    public C(B init) {  
        obiect = new B(init);  
    }  
    public B getB() {  
        return new B(obiect);  
    }  
}
```

Care clase sunt imutabile? \*

(5 Points)

- ☐ B
- ☐ C
- ☐ B si C
- ☒ nici una

## 4. Avem la dispozitie un String text ce contine litere mari si mici din alfabetul englez, de lungime cel putin 5. Care dintre urmatoarele afirmatii sunt adevarate? \*

(5 Points)

- ☒ `text.toLowerCase().equals(text.toLowerCase())`
- ☐ `text.replaceAll("a", "").contains("a")`
- ☒ `text.endsWith(text.substring(5))`
- ☒ `text.toLowerCase().intern()==text.toLowerCase().intern()`

5. Sa presupunem ca un obiect este accesat printr-o referinta la o interfata pe care o implementeaza. Atunci: \*

(5 Points)

- ☒ obiectul poate accesa metodele publice din clasa din care face parte
- ☐ obiectul poate accesa metode ale suprac clasei din care face parte daca acestea au fost rescrise
- ☒ obiectul poate accesa metodele din interfata
- ☒ obiectul poate accesa metodele mostenite din clasa Object

6. Care dintre urmatoarele afirmatii sunt adevarate? \*

(5 Points)

- ☐ Nu exista cazuri cand se poate omite cuvantul cheie return intr-o lambda expresie.
- ☒ Nu este obligatoriu de specificat tipul parametrilor intr-o lambda expresie.
- ☐ Nu exista cazuri cand se pot omite acoladele intr-o lambda expresie.
- ☒ O interfata functionala este o interfata care contine o singura metoda abstracta.

7. Care dintre urmatoarele afirmatii sunt adevarate? \*

(5 Points)

- ☐ o data membra statica previne instantierea clasei
- ☒ o data membra private poate fi accesata doar din clasa din care face parte
- ☒ Garbage Collector nu poate fi fortat de programator sa distruga un obiect
- ☐ o clasa cu modificatorul public poate fi instantiata numai din interiorul pachetului sau

8. Care dintre urmatoarele afirmatii sunt adevarate? \*

(5 Points)

- ☐ Un singur fir poate rula o metoda nestatica synchronized a unui obiect, dar un al doilea fir poate sa ruleze in paralel aceasta metoda pe un alt obiect din aceeasi clasa.

- ☐ Firele de executare detin memorie proprie de tip heap, dar memoria stiva este partajata.
- ☐ Un fir de executare se poate defini numai prin extinderea clasei Thread.
- ☒ Un fir de executare este creat in momentul apelarii metodei start().

9. Care dintre urmatoarele afirmatii sunt adevarate? \*

(5 Points)

- ☐ Daca suprascriem metodele hashCode() si equals() garantam ca nu vom avea coliziuni in HashMap
- ☒ Conflictul intr-un HashSet se rezolva prin inlantuirea valorilor in lista elementelor cu aceeasi cheie.
- ☐ Nu se poate seta capacitatea initiala a unui ArrayList
- ☒ Cheia obiectelor dintr-un HashSet este egala cu rezultatul metodei hashCode()

10. Care dintre urmatoarele afirmatii sunt adevarate? \*

(5 Points)

- ☒ Exceptiile la compilare sunt obligatoriu de tratat.
- ☐ Exceptiile sunt tipuri de date primitive.
- ☒ Daca ExceptieB extinde ExceptieA atunci catch-ul corespunzator lui ExceptieA trebuie sa fie mai jos decat catch-ul corespunzator lui ExceptieB.
- ☐ Erorile Java sunt obligatoriu de tratat.

11. Fie urmatoarea clasa:

```
class A{  
    int a;  
    public A(int i){a=i;}  
    public int hashCode(){ return a;}  
    public boolean equals(Object other){return false;}  
}
```

Sa presupunem ca in main se executa:

```
HashMap<A,Integer> H = new HashMap<>();  
H.put(new A(1), 1);  
H.put(new A(1), 1);  
H.put(new A(2), 2);  
H.put(new A(2), 2);
```

Ce contine tabela de dispersie H? \*

(5 Points)

- ☒ hash(A1) -> 1, hash(A1) -> 1, hash(A2) -> 2, hash(A2) -> 2
- ☐ hash(A1) -> 1, hash(A2) -> 2
- ☐ hash(A1)=hash(A2) -> 1
- ☐ hash(A1)=hash(A2) -> 2

12. Fie urmatorul program Java:

```
class Fir extends Thread{
    int nivel;
    static int numar = 0;
    public Fir(int n){nivel=n;}
    public void run(){
        System.out.print(nivel+" ");
        if (nivel<3){
            Fir fir = new Fir(nivel+1);
            fir.start();
        }
        if (nivel<10000000){
            Fir fir = new Fir(nivel+1);
            fir.start();
        }
    }
}

public class Examen {
    public static void main(String[] args) {
        Fir fir = new Fir(0);
        fir.start();
    }
}
```

Presupunand ca s-ar termina executia programului, de cate ori apare afisata valoarea 9999999  
? \*

(5 Points)

- ☒ 8 ori
- ☐ 9 ori
- ☐ 12 ori
- ☐ 16 ori

13. Fie urmatorul program Java:

```
public class Examen {
    public static void main(String[] args) throws FileNotFoundException {
        for (int i = 1; i <= 5; i++)
            try (Scanner sc = new Scanner(new File("file" + i + ".txt"))) {
                System.out.print(sc.nextInt());
            } catch (FileNotFoundException e) {
                System.out.print("!");
            } catch (Exception e) {
                System.out.print("?");
            } finally {
                System.out.print("F");
            }
        }
    }
}
```

Presupunem ca in radacina directorului unde se executa programul exista doar urmatoarele fisiere text:

file1.txt gol

file2.txt continand numarul 2

file3.txt gol

file4.txt continand numarul 4

Outputul programului este: \*

(5 Points)

- ☐ ?2?4!F, executie terminata cu succes
- ☒ ?F2F?F4F!F, executie terminata cu succes
- ☐ ?F2F?F4F?F, executie terminata cu succes
- ☐ ?F2F?F4FF, executie terminata cu exceptie

14. Sa presupunem ca avem la dispozitie trei fisiere din acelasi pachet Java: A.java, B.java, C.java.

Care din urmatoarele declaratii contin erori: \*

(5 Points)

- ☒ public interface A {...} public interface B extends A {...} public class C extends A,B {...}
- ☐ public class A {...} public class B extends A {...} public final class C extends B {...}
- ☐ public abstract class A {...} public class B extends A {...} public abstract class C extends B {...}
- ☒ public class A {...} public static class B {...} public final class C extends A {...}

15. Care dintre urmatoarele afirmatii sunt adevarate? \*

(5 Points)



- ☐ implicit equals() intoarce egalitatea camp cu camp a datelor membre
- ☒ o data membra a superclasei care este ascunsa de o data membra cu acelasi nume a subclasei devine inaccesibila
- ☒ constructorul superclasei se poate apela doar la inceputul constructorului clasei derivate
- ☒ o metoda suprascrisa poate intoarce un tip mai derivat decat cel specificat in superclasa care contine metoda

16. Presupunem ca exista si sunt implementate corect metodele get si set pentru campurile String nume si double salariu ale clasei Persoana.  
Ce produce urmatorul cod?

```
listaPersoane.stream\(\)  
.filter(p -> p.getNum().startsWith("B"))  
.filter(p -> p.getSalariu() > 2000)  
.map(Persoana::getNum)  
.sorted()  
.map(o -> o.toString())  
.collect(Collectors.joining(", ")) *
```

(5 Points)

- ☐ un sir de caractere produs din concatenarea listei sortate a salariilor mai mari decat 2000 ale persoanelor al caror nume incepe cu B
- ☒ un sir de caractere produs din concatenarea listei sortate a numelor persoanelor care incep cu B si cu salarii mai mari decat 2000
- ☐ lungimea sirului de caractere produs din concatenarea numelor persoanelor care incep cu B si cu salarii mai mari decat 2000
- ☐ arunca exceptie



17. Fie clasa A si urmatoarele interfete functionale:

```
class A{
    int a;
    public A(int i){a=i;}
    public String toString(){return "A" + (a%3);}
    public int get(){return a/3;}
}
interface Interface1{void f(int x);}
interface Interface2{int f();}
interface Interface3{String f(String s);}
interface Interface4{void f(String s);}
```

In main avem:

```
int x=.....;
A obiect = new A(x);
Interface1 ob1 = i -> System.out.println(i * 3);
Interface2 ob2 = () -> obiect.get();
Interface3 ob3 = s -> s.substring(1);
Interface4 ob4 = s -> System.out.println(s);
```

```
ob1.f(ob2.f());
ob4.f(ob3.f(obiect.toString()));
```

Daca programul afiseaza:

42  
2

Care dintre urmatoarele afirmatii sunt adevarate pentru valoarea x? \*  
(5 Points)

- ☐ este multiplu de 3
- ☒ este multiplu de 4
- ☐ este multiplu de 5
- ☐ este multiplu de 7

18. Care dintre urmatoarele afirmatii sunt adevarate? \*  
(5 Points)

- ☐ Dupa deschiderea unui stream se pot efectua o singura operatie intermediara si mai multe operatii de inchidere.
- ☒ Operatiile intermediare asupra unui stream intorc un nou stream.
- ☐ Sortarea este o operatie intermediara de tip stateless.
- ☒ Streamurile nu sunt reutilizabile.

19. Fie urmatorul cod:

```
class A{
    int intA;
    public A met1(){
        return new A();
    }
    final void met2(){
    public void met3(){
    public static void met4(){
    private int met5(int i){return 5;}
}
```

```
class B extends A{
    int intB;
    public B met1(){
        return new B();
    }
    public void met2(){
    private void met3(){
    static void met4(){
    private int met5(){return 5;}
}
```

Cate suprascrieri, supraincarcari, respectiv erori contine codul? \*

(5 Points)

- ☒ 1 suprascriere, 1 supraincarcare, 3 erori
- ☐ 2 suprascrieri, 1 supraincarcare, 2 erori
- ☐ 1 suprascriere, 2 supraincarcari, 2 erori
- ☐ 4 suprascrieri, 1 supraincarcare, 0 erori

20. In care din urmatoarele situatii isi suspenda executia un fir de executare? \*

(5 Points)

- ☐ Cand apeleaza o metoda nestatica, synchronized pe care toate celelalte fire o executa pe cate o instanta diferita a aceleiasi clase.
- ☐ In toate situatiile cand apeleaza o metoda statica synchronized.
- ☒ Cand apeleaza o metoda nestatica, synchronized pe care alt fir inca o executa pe un obiect comun celor doua fire.
- ☒ Cand apeleaza metoda wait() asupra unui obiect oarecare.

This content is created by the owner of the form. The data you submit will be sent to the form owner. Microsoft is not responsible for the privacy or security practices of its customers, including those of this form owner. Never give out your password.

Powered by Microsoft Forms | [Privacy and cookies](#) | [Terms of use](#)