

Examen PAO 2

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, de la ora 12:30 la ora 14:00.

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

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 (daca ati bifat optiunea mentionata mai sus) si apoi deconectati-va din meeting.

DESPRE COMPLETAREA FORMULARULUI

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.

...

Points: 75/90

1. Grupa *

- ☐ 231
- ☐ 232
- ☐ 233
- ☒ 234

2. Nume Prenume *

Buhai Darius

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

(5/5 Points)

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

4. Care dintre urmatoarele afirmatii sunt adevarate? *

(5/5 Points)

- ☒ o metoda suprascrisa poate intoarce un tip mai derivat decat cel specificat in superclasa care contine metoda ✓
- ☐ 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 ✓

5. Sa presupunem ca avem la dispozitie trei fisiere din acelasi pachet Java: A.java, B.java, C.java. Care din urmatoarele declaratii contin erori? *
(5/5 Points)

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

clasele top-level (cele dupa care e numit si fisierul) trebuie sa fie public sau default (deci nu private sau protected), si nici static nu pot sa fie => la B e eroare

6. 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<2){
            Fir fir = new Fir(nivel+1);
            fir.start();
        }
        if (nivel<2){
            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();
    }
}
```

nivel = 0: 1 fir
nivel = 1: 3 fire
nivel = 2: 9 fire
nivel = 3: 9 fire
...
nivel = 9999999: 9 fire

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

(5/5 Points)

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

7. 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("!");
                throw e;
            } catch (NoSuchElementException 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:" *

(5/5 Points)

- ☐ ?F2F?F4F?F, executare terminata cu succes
- ☐ ?F2F?F4F!F, executare terminata cu succes
- ☒ ?F2F?F4F!F, executare terminata cu exceptie ✓
- ☐ ?2?4!F, executare terminata cu succes

8. Care dintre urmatoarele afirmatii sunt adevarate? *

(5/5 Points)

- ☐ un fir de executare se poate defini numai prin implementarea interfetei Runnable
- ☐ un fir de executare este creat in momentul apelarii metodei run()

- ☒ firele de executare detin memorie proprie de tip stiva, dar memoria heap este partajata ✓
- ☒ un fir de executare iese din starea suspendata cu wait() in momentul cand alt fir apeleaza notify() pe acelasi obiect pe care s-a apelat wait() ✓

✗

9. Care dintre urmatoarele afirmatii sunt adevarate? *
(0/5 Points)

- ☒ daca suprascriem metodele hashCode() si equals() garantam ca nu vom avea coliziuni in HashMap
- ☐ exista clase care extind Collection, dar nu implementeaza Iterable
- ☒ folosind metoda add() din LinkedHashSet si ArrayList se retine ordinea elementelor ✓
- ☐ TreeSet nu este o colectie iterabila

✗

10. Fie urmatoarea clasa:

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

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? *
(0/5 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

11. Fie urmatorul cod:

```
class A{
    int intA;
    public A met1(){
        return new A();
    }
    private 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(){    suprascriere
        return new B();
    }
    public void met2(){} suprascriere
    private void met3(){} eroare (trebuia accesul public)
    static void met4(){} eroare (trebuia accesul public)
    private int met5(){return 5;} supraincarcare
}
```

Cate suprascrieri, supraincarcari, respectiv erori contine codul? *
(5/5 Points)

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

12. Care dintre urmatoarele afirmatii sunt adevarate? *
(5/5 Points)

- ☒ lambda expresiile pot avea lista parametrilor vida ✓
- ☐ o interfata functionala este o interfata care contine cel putin o metoda abstracta
- ☐ Comparator este o interfata functionala care este utilizata in scopul determinarii egalitatii dintre doua obiecte
- ☒ exista cazuri cand se pot omite acoladele intr-o lambda expresie ✓

13. Fie clasa A si urmatoarele interfete functionale:

```
class A{
    int a;
    public A(int i){a=i;}
    public String toString(){return "A"+(a%5);}
    public int get(){return a/5;}
}
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 * 5);
Interface2 ob2 = () -> obiect.get();
Interface3 ob3 = s -> s.substring(1);
Interface4 ob4 = s -> System.out.println(s);
```

```
ob1.f(ob2.f()); (x/5) * 5 = 15
ob4.f(ob3.f(obiect.toString())); A(x%5) -> x%5 = 3
```

Daca outputul este:

15
3

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

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

14. Care dintre urmatoarele afirmatii sunt adevarate? *
(5/5 Points)

- ☐ un constructor nu poate fi static sau abstract, dar poate fi final
- ☒ o data membra public poate fi accesata si din afara clasei ✓
- ☐ o clasa cu modificatorul abstract nu poate fi extinsa
- ☒ un constructor poate fi public sau private ✓

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

(5/5 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

16. Care dintre urmatoarele afirmatii sunt adevarate? *
(5/5 Points)

- ☒ dupa deschiderea unui stream se pot efectua mai multe operatii intermediare si o singura operatie de inchidere ✓

- ☐ operatiile intermediare asupra unui stream sunt aplicate in momentul invocarii unei operatii de deschidere
- ☐ streamurile sunt reutilizabile
- ☒ metoda Stream.iterate poate fi utilizata pentru producerea unui sir de numere impare ✓

17. Metodele din clasa corespunzatoare unui obiect Java pot fi accesate folosind: *
(5/5 Points)

- ☒ o referinta la obiect de tipul clasei din care face parte obiectul ✓
- ☐ o referinta la obiect de tipul unei interfate pe care clasa o implementeaza
- ☒ o referinta la obiect de tipul superclasei obiectului, dar in acest caz sunt accesibile doar metodele suprascrise ✓
- ☐ o referinta la obiect de tip Object

×

18. 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? *
(0/5 Points)

- ☐ B
- ☐ C
- ☒ B si C
- ☐ nici una ✓

19. Care dintre urmatoarele afirmatii sunt adevarate? *
(5/5 Points)

- ☐ erorile Java sunt obligatoriu de tratat
- ☒ 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 ✓

20. In care din urmatoarele situatii isi suspenda executarea un fir de executare? *
(5/5 Points)

- ☐ cand alt fir apeleaza metoda notify() asupra unui obiect comun
- ☒ cand apeleaza o metoda nestatica, synchronized pe care alt fir inca o executa pe un obiect comun celor doua fire ✓
- ☐ cand apeleaza o metoda nestatica, synchronized pe care toate celelalte fire o executa pe cate o instanta diferita a aceleiasi clase
- ☒ 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 (<https://go.microsoft.com/fwlink/p/?linkid=857875>) | Terms of use (<https://go.microsoft.com/fwlink/p/?LinkId=2083423>)