

Nume și prenume student: _____

Grupa: _____

EXAMEN SCRIS - GRILĂ cu justificare răspuns, 10% din nota finală

Obs. La examenul scris vor fi două grile – 5p fiecare, vor exista și punctaje parțiale pt explicații corecte

Ce se afișează la rularea codului următor? (Această grila este un exemplu)

```
class AA {
    int x;
    protected AA() { init(1008); }
    protected void init(int x) { this.x = x; }
}

class BB extends AA {
    public BB() { init (super.x * 2); }
    public void init(int x) { super.x = x + 1; }
}

public class Ex2 {
    public static void main(String[] args) {
        BB a = new BB();
        System.out.println(a.x);
    }
}
```

- **Variante de răspuns:**

- a) 2018
- b) 2019
- c) eroare
- d) 0

- **Justificare raspuns:**

- Se apelează constructorul clasei BB, apoi constructorul Clasei AA, apoi init(1008) din BB (polimorfism) x devin 1009, se revine in Constructorul BB, se apeleaza init(2018), se apeleaza init(2018) si valoarea atributului x din clasa AA devine 2019.

Obs. Se acceptă și alte explicații care surprind esența, dar nu povesti care nu au legatura cu grila.

- **Identificati conceptele din curs care au legatura cu grila propusa:**

- moștenire, polimorfism

- **Dificultate:**

(Easy, Medium, Difficult)

Timp de lucru examen scris: 15min.

EXAMEN PRACTIC - MODEL - COMENZI RESTAURANT 40% din nota finală

Se dorește o aplicație cu o interfață grafică intuitivă, care să permită plasarea comenzilor în cadrul unui restaurant.

Cerințe funcționale - 7p:

- 1) **1p** La pornirea aplicației se va deschide **o fereastră pt. angajații restaurantului și câte o fereastră pt. fiecare masă citită din fisierul/tabelul Tables**. Fereastră destinată angajaților va afișa numele “Staff”, și fiecare fereastră asociată unei mese va afișa id-ul acesteia (ex: “Table 3”).

- Se va defini clasa:
Table: { id: int }

- 2) **2p** Fiecare dintre ferestrele asociate meselor va afișa **meniul restaurantului, grupat pe categorii**. Meniul se va citi din fisierul/tabelul **Menu**.

- Se va defini clasa:
MenuItem: { id: int, category: String, item: String, price: float, currency: String }
- Ex. de intrări din fisierul/tabelul Menu:
 - 1, Antreuri, Bruschete cu rosii, 15, RON
 - 2, Antreuri, Salata Caprese, 20, RON
 - 3, Fel Principal, Paste cu sos pesto, 25, RON
 - 4, Fel Principal, Vinete parmigiana, 25, RON
- Meniul va fi afișat sub forma de tabele, câte unul pentru fiecare Categorie. De ex:

Antreuri

Bruschete cu rosii	15 RON
Salata Caprese	20 RON

Fel Principal

Paste cu sos pesto	25 RON
Vinete parmigiana	25 RON

- 3) **2p** În ferestrele asociate meselor, clienții pot **selecta mai multe intrări din meniu** (prin multiple selection sau checkbox-uri etc.), iar apoi pot **plasa comanda** aferentă produselor selectate prin acționarea unui buton “Place order”.

Comenzile plasate se vor salva în fișierele/tabelele Orders și OrderItems, având data calculată automat și statusul PLACED.

Observație: Pt. simplitate vom considera că o comandă poate conține o singură bucată dintr-un anumit produs.

- Se vor defini clasele:
Order: { id: int, table: int/Table, menuitems: List<Integer>/List<MenuItem>, date: LocalDateTime, status: OrderStatus }, unde OrderStatus este un enum {PLACED, PREPARING, SERVED}
- Ex. de conținut fișiere/tabele Orders și OrderItems:

Order:

id=1, tableId=3, date=2022-01-20T12:00, status=PLACED

OrderItems:

orderId=1, menuItemId=1

orderId=1, menuItemId=2

- 4) **2p** În momentul în care o comandă este plasată aceasta va apărea **instantaneu** (fără a fi nevoie de vreo acțiune din partea utilizatorului) **în fereastră “Staff”**, în cadrul unui tabel Placed Orders. Intrările din acest tabel vor fi

ordonate crescator dupa data la care s-a plasat comanda. Se vor afisa: id-ul mesei, data, numele produselor comandate (nu id-urile!).

Cerinte non-functionale - 3p: 0.6p*5

- Validarea datelor de intrare
Exemplu: Nu se poate plasa o comanda fara a selecta niciun produs
- Procesarea va avea loc numai la nivel de service sau de controller; interactiunea cu sursa de date se va face numai prin intermediul repository-ului (bază de date)
- Interactiunea cu utilizatorul va avea loc numai in UI (GUI)
- Se incurajează construirea unei aplicații de la 0 sau eliminarea codului care nu este folosit, precum si a functionalitatilor care nu s-au cerut (daca ati lucrat cu ceva template de la lab)
- Clasele, atributele și metodele lor vor avea exact numele cerute in problema sau nume sugestiv daca nu s-a specificat explicit numele lor

IMPORTANT

- Se puncteaza doar cerintele functionale care ruleaza
- Orice cod care nu poate fi explicat, atrage dupa sine nepunctarea cerintei/cerintelor din care face parte
- Nu aveti voie sa comunicati in timpul examenului, in nicio modalitate posibila de a face acest lucru (chat, mail, ...etc.)

Cerinte extra – pt. bonus: 1p in plus la nota finala:

Pt. 1p in plus la nota finala se poate rezolva oricare dintre urmatoarele cerinte (nu ambele):

- 1) Adaugarea posibilitatii de a selecta cantitatea dorita pt. fiecare produs atunci cand se plaseaza o comanda.
 - Se va adauga clasa **OrderItem** { **order**: int/Order, **menuitem**: int/Menuitem, **quantity**: int }
 - Field-ul **menuitems** din clasa **Order** va continue o lista de id-uri de OrderItem sau o lista de OrderItem
 - Pentru fiecare intrare din fisierul/tabelul OrderItems se va adauga si cantitatea
 - Pe fiecare rand din tabelele in care este afisat meniul se va adauga o noua coloana ce va continue un input field "Quantity". Acesta va fi disabled pt. produsele care nu au fost selectate.
- 2) Fereastra "Staff" va permite schimbarea statusului comenzilor in felul urmator:
 - O comanda PLACED poate fi marcata ca PREPARING DELIVERED (prin actionarea unui buton). In acest caz comanda se va muta din tabelul Placed Orders intr-un nou tabel Preparing Orders (ordonat tot dupa data ascending). Fereastra corespunzatoare mesei asociata comenzii va afisa instantaneu o notificare "Your order is being prepared"
 - O comanda PREPARING (din tabelul Preparing Orders) poate fi marcata ca DELIVERED (prin actionarea unui buton). In acest caz comanda va disparea din tabelul Preparing Orders. Fereastra corespunzatoare mesei asociata comenzii va afisa instantaneu o notificare "Your order has been delivered to your table"
 - Toate aceste modificari de status vor fi salvate corespunzator in fisierul/tabelul Orders

Timp de lucru examen practic: 2h 30min.