

## **English**

### **A. Hospital**

#### **A.1.**

When a patient is admitted in the application, they are saved. Each patient has the possibility to pay extra for certain facilities such as: adjustable bed, breakfast included, room slippers, indoor robe. If the patient does not choose these extra facilities, they are set to false in the application. Develop the module that ensures the creation of patient-type objects with extra options.

#### **A.2.**

In the application, hospital staff is of several types. These are saved in an enum { Stretcher Bearer, Assistant, Doctor }. Implement the module that enables the creation of objects from the HospitalStaff family depending on the type received as a parameter.

#### **A.3.**

In the application, hospital staff is of several types: Stretcher Bearer, Assistant, Doctor, Secretary, Registrar. Implement the module that enables the creation of objects from the HospitalStaff family, knowing that the staff is divided into two categories: medical staff and non-medical staff.

#### **A.4.**

The hospital also has a laboratory where several chemists produce different recipes for medications. When a recipe is produced, the quantities of the solutions used must be taken into account. If a recipe is created, it is recommended to be used for creating future medications without going through the process of creating the recipe. Implement the module that facilitates the creation of new recipe objects without needing to call the constructor.

#### **A.5.**

The hospital owns a small shop for medications and has an application for purchasing medications based on prescriptions. The hospital enters into a contract with a specialized pharmacy and wishes to integrate the pharmacy's information system with the existing software system in the small medication shop. The pharmacy's developers need to integrate these two applications so that the pharmacy application can use the Medication type objects from the hospital application. The Medication class from the hospital application has the methods purchaseMedication() and presentPrescription(). The presentPrescription() method is called from purchaseMedication() to verify the prescription. The Medication class from the pharmacy application has a single method buyMedication() in which prescription verification is not done, because the pharmacy will also sell medications without a prescription.

#### **A.6.**

For the admission of a patient to the hospital, the severity of the patient's condition must be verified through the Patient class, verification of the Doctor's confirmation that admission is necessary, verification of the availability of a bed in the room through the Ward class which has a

list of free and occupied beds. The hospital wants the hospital staff not to have to make these verifications separately but to have a module developed that allows for the easy verification of these things.

#### **A.7.**

The hospital wishes to test the provision of results via the online platform rather than just printed. However, there is a risk of returning to the initial form of providing results to patients. The addition of this new functionality for the software system is desired, which would allow returning to the initial situation.

#### **A.8.**

It is desired to represent the hospital departments in the application. Each department contains subdepartments or sections. Sections do not contain subsections. Implement the module that allows the hierarchical representation of hospital departments and sections.

#### **A.9.**

Because the Hospital is overcrowded, it is required that when patients want to be admitted, only people who have health insurance should be admitted. Create an intermediate level that allows the admission of only these people.

#### **A.10.**

For each admission, information about the admitted patient must be retained, such as: name, phone number, address, etc., as well as information about the ward where they are admitted: ward number, bed number, number of hospitalization days, etc. Thus, if a patient is admitted multiple times over time, the information about them is the same and is repeated, occupying a lot of memory. Implement the module for storing all admissions so that it does not occupy too much memory.

#### **A.11.**

It is desired to implement the payment module for patients who have been admitted to the hospital. The payment method is decided by the person who pays at the time they need to make the payment. Payment can be made by card or cash. Implement the payment module.

#### **A.12.**

The hospital wants to notify all patients who have been in the hospital whenever an emergency regarding existing viruses in the city appears. Thus, it is desired to implement a module that, when an epidemic or a new virus appears, sends notifications to all people subscribed to the hospital's notifications.

#### **A.13.**

The hospital wants to implement a module for patient management. A patient can have one of the following states: Admitted, UnderObservation, Discharged. When a patient is brought to the hospital, they enter the Admitted state. If their health condition is serious, then they are moved to

the UnderObservation state. When they heal and are sent home, the patient enters the Discharged state.

#### **A.14.**

The admission of a patient is done following these steps: The difficulty of the patient's condition is analyzed, The availability in the hospital wards is checked, An admission form is issued. Implement the module that performs patient admission in the application according to this pattern.

#### **A.15.**

The hospital manager wants to speed up the process at the emergency reception, and thus the operator who receives patients at triage will give commands for admission or immediate treatment for the arriving patients and will place them in separate queues. The commands are sent to doctors, but the operator can receive other patients much faster. Implement the module that allows sending commands to doctors from the reception operator.

## Română

### **A. Spital**

#### **A.1.**

Atunci când un pacient este internat în cadrul aplicației se face salvarea acestuia. Fiecare pacient are posibilitatea să plătească extra pentru anumite facilități precum: pat rabatabil, mic dejun inclus, papuci de camera, halat pentru interior. În cazul în care pacientul nu alege aceste facilități extra, în cadrul aplicației sunt setate cu false. Să se dezvolte modulul care asigură crearea de obiecte de tipul pacient cu opțiuni extra.

#### **A.2.**

În cadrul aplicației personalul spitalului este de mai multe tipuri. Acestea sunt salvate într-un enum { Brancardier, Asistent, Medic }. Să se implementeze modulul care pune la dispoziție crearea de obiecte din familia obiectelor PersonalSpital în funcție de tipul primit ca parametru.

#### **A.3.**

În cadrul aplicației personalul spitalului este de mai multe tipuri: Brancardier, Asistent, Medic, Secretar, Registrator. Să se implementeze modulul care pune la dispoziție crearea de obiecte din familia obiectelor PersonalSpital știind faptul că personalul este împărțit în două categorii: personal medical și personal non-Medical.

#### **A.4.**

Spitalul are în dotare și un laborator în care mai mulți chimiști produc diferite rețete pentru medicamente. În momentul în care o rețetă este produsă trebuie ținut cont de cantitățile din soluțiile care sunt folosite. Dacă o rețetă este creată este recomandat să fie folosită pentru crearea viitoarelor medicamente fără a se trece prin procesul de creare al rețetei. Să se implementeze modulul care facilitează crearea de noi obiecte de rețete fără a fi nevoie de apelul constructorului.

#### **A.5.**

Spitalul deține un mic magazin pentru medicamente și are o aplicație pentru cumpărarea de medicamente pe baza de rețetă. Spitalul încheie un contract cu o farmacie specializată și dorește să integreze sistemul informatic al farmaciei cu sistemul software existent în micul magazin de medicamente. Dezvoltatorii farmaciei trebuie să integreze aceste două aplicații, astfel încât aplicația farmaciei să poată folosi obiectele de tip Medicament din aplicația spitalului. Clasa Medicament din aplicația spitalului are metodele achizitioneazaMedicament() și prezintaReteta(). Metoda prezintaReteta() este apelată din achizitioneazaMedicament() pentru verificarea rețetei. Clasa Medicament din aplicația farmaciei are o singură metodă cumparaMedicament() în care nu se face verificarea rețetei, deoarece farmacia o să vândă medicamente și fără rețetă.

#### **A.6.**

Pentru internarea unui pacient în spital trebuie verificată gravitatea stării pacientului prin intermediul clasei Pacient, verificarea confirmării Medicului că trebuie internat, verificarea disponibilității unui pat în cameră prin intermediul clasei Salon care are lista cu paturile libere și ocupate. Spitalul dorește ca personalul spitalului să nu fi nevoit să facă aceste verificări separat ci să fie dezvoltat un modul care să permită această verificare facilă a acestor lucruri.

#### **A.7.**

Spitalul dorește să testeze punerea la dispoziție a rezultatelor prin intermediul platformei online ci nu doar printat. Există însă riscul să se revină la forma inițială de punere la dispoziția pacienților a rezultatelor. Se dorește adăugarea acestei noi funcționalități pentru sistemul software, care să permită revenirea la situația inițială.

#### **A.8.**

Este dorită reprezentarea departamentelor spitalului în cadrul aplicației. Fiecare departament conține subdepartamente sau secții. Secțiile nu conțin subsecții. Să se implementeze modulul care permite reprezentarea arborescentă a departamentelor și secțiilor spitalului.

#### **A.9.**

Deoarece Spitalul este supraaglomerat se impune ca atunci când pacienții doresc internarea să fie internate doar persoane care au asigurare de sănătate. Să se realizeze un nivel intermediar care să permită internarea doar acestor persoane.

#### **A.10.**

Pentru fiecare internare trebuie să se rețină informații cu privire la pacientul internat precum: nume, număr de telefon, adresă, etc, precum și informațiile despre salonul unde este internat: număr salon, număr pat, număr zile spitalizare, etc. Astfel, dacă un pacient este internat de mai multe ori de-a lungul timpului, informațiile despre acesta sunt aceleași și se repetă, ocupând foarte multă memorie. Să se implementeze modulul de memorare al tuturor internărilor astfel încât să nu ocupe memorie foarte multă.

#### **A.11.**

Este dorită implementarea modului de plată pentru pacienții care au fost internați în spital. Modul de plată îl decide persoana care plătește în momentul în care trebuie să facă plata. Plata se poate realiza cu cardul sau cash. Să se implementeze modulul de plata.

#### **A.12.**

Spitalul dorește să anunțe toți pacienții care au fost în spital ori de câte ori apare vreo urgență cu privire la viruși existenți în oraș. Astfel se dorește implementarea unui modul care atunci când apare o epidemie sau un virus nou să se trimită notificări tuturor persoanelor abonate la notificările spitalului.

#### **A.13.**

Spitalul dorește implementarea unui modul pentru gestiunea pacienților. Un pacient poate să aibă una din următoarele stări: Internat, SubObservatie, Externat. Atunci când un pacient este adus în spital intră în starea Internat. Dacă starea sa de sănătate este gravă, atunci este trecut în starea SubObservatie. Când se vindeca și este trimis acasă, pacientul intră în starea Externat.

#### **A.14.**

Internarea unui pacient se face după următorii pași: Se analizează dificultatea stării pacientului, Se verifică disponibilitatea în saloanele spitalului, Se emite fișa de internare. Să se implementeze modulul care realizează în aplicație internarea pacienților după acest pattern.

#### **A.15.**

Managerul spitalului dorește să grăbească procesul de la primiri urgente și astfel operatorul care primește pacienții la triaj va da comenzi de internare sau de tratare imediată pentru pacienții veniți și îi va așeza astfel la alte cozi separate. Comenzile sunt trimise către medici, însă operatorul poate primi alți pacienți mult mai rapid. Să se implementeze modulul care permite trimiterea de comenzi către medici de la operatorul de primire.