ourdiu.com | Shared by Md. Mehedi Hasan

Daffodil International University
Faculty of Science & Information Technology
Department of Software Engineering
Course Code: SE216; Course Title: Object Oriented Programming

Time: 40 Mins | Marks: 15

1.	a)	A **Smart Healthcare Monitoring System** manages different types of patients and provides various healthcare services. The system should be designed using **interfaces, interface inheritance, abstraction, method overriding, and runtime polymorphism.** a) Create an interface named **`Monitorable`** containing one abstract method named `monitor()`. b) Create another interface named **`Reportable`** containing one abstract method named `generateReport()`. c) Create another interface named **`Alertable`** containing one abstract method named `sendAlert()`. d) Create an interface named **`HealthService`** that extends both `Monitorable` and `Reportable`. Add another abstract method named `calculateHealthScore()`. e) Create an **abstract class** **`Patient`** that implements both `HealthService` and `Alertable`. The class must contain two abstract methods, `admit()` and `discharge()`, and one concrete method, `displayBasicInfo()`. f) Create two subclasses named **`InPatient`** and **`EmergencyPatient`** that extend `Patient`. Implement all required abstract methods in both classes. g) Override the `displayBasicInfo()` method in both subclasses and provide different information for each patient type. h) Implement the methods of `InPatient` and `EmergencyPatient` differently to demonstrate **method overriding**. i) Create a class named **`TesterClass`**. Create objects of both subclasses using **`Patient`** class **references** and demonstrate **upcasting and runtime polymorphism.** **Output:** Patient Type: Inpatient Displaying inpatient information... Admitting inpatient... Monitoring inpatient vital signs... Generating inpatient health report... Health Score: 85 Inpatient alert sent to doctor. Inpatient discharged successfully. Patient Type: Emergency Patient Displaying emergency patient information... Admitting emergency patient... Continuously monitoring emergency patient... Generating emergency health report... Health Score: 65 Emergency alert sent to medical team. Emergency patient discharged successfully.	[8]
	b)	A **Cinema Ticket System** stores movie names and the number of available seats for each movie in two parallel arrays. Write a Java program to process a seat booking request for a given movie position. Your program must: 1. Ask the user to enter an **array index** and the **number of seats to book**. 2. Use a **nested `try-catch` structure** to handle:	[7]

ourdiu.com

ourdiu.com | Shared by Md. Mehedi Hasan

- Invalid index → catch `ArrayIndexOutOfBoundsException` and display `"Invalid movie selection."`
- Requested seats greater than available seats → throw a custom exception named `InsufficientSeatException` and display `"Booking failed: Not enough seats available."`

3. If the booking is successful, display:
   - Movie name
   - Number of seats booked
   - Remaining seats
4. Use a `finally` block to always display:
   `"=== Booking process completed ==="`
5. Use the following hard-coded data:
   - Movies: `{"Avatar", "Inception", "Titanic", "Interstellar"}`
   - Seats: `{10, 5, 0, 8}`

ourdiu.com