

Beispielklausur 2 zu Softwaretechnik

Prof.-Dr. R. Hennicker

Aufgabe 1

(5 Punkte)

- (a) Was versteht man unter einer Template Operation? (2 Punkte)
- (b) Geben Sie ein Beispiel für eine Template Operation an! (3 Punkte)

Aufgabe 2

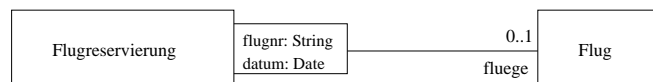
(5 Punkte)

In dem auf Arbeitsblatt 1 angegebenen Sequenzdiagramm befinden sich vier (syntaktische oder semantische) Fehler. Markieren Sie die fehlerhaften Stellen und beschreiben Sie (auf Arbeitsblatt 1), was falsch gemacht wurde!

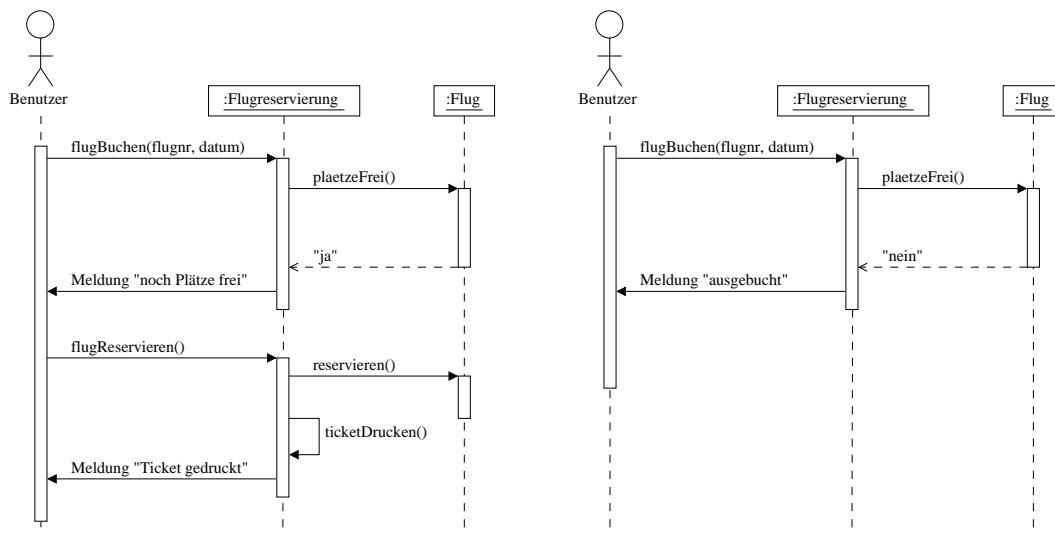
Aufgabe 3

(12 Punkte)

Gegeben sei das folgende statische Modell für ein einfaches Flugreservierungs-System.



Die beiden folgenden Sequenzdiagramme beschreiben mögliche Szenarien bei der Buchung eines Flugs. (Die Szenarien können beliebig wiederholt werden.)



- (a) Entwickeln Sie, gemäß der in der Vorlesung angegebenen Methode ein Zustandsdiagramm (mit zwei stabilen und zwei Aktivitätszuständen), das das Verhalten eines Objekts der Klasse *Flugreservierung* zeigt. (4 Punkte)

- (b) Für jeden der beiden Aktivitätszustände ist ein Aktivitätsdiagramm anzugeben. (5 Punkte)
- (c) Welche Operationen müssen in einem Objektentwurf zu den Klassen *Flugreservierung* und *Flug* hinzugenommen werden? (3 Punkte)

Aufgabe 4

(13 Punkte)

Eine Aktiensimulation soll folgende Funktionalität erfüllen: Die Benutzerschnittstelle der Simulation hat zwei Eingabefelder *nameField* und *kursField*, in die ein Aktienname und der aktuelle Kurs der Aktie eingetragen werden können. Bei Betätigen des Knopfes *bestaetigungsButton* werden die beiden Felder eingelesen. Falls der aktuelle Kurs der Aktie um mehr als die Hälfte unter dem bisherigen Kurs liegt, erscheint in einem Ausgabefeld namens *disasterField* die Meldung "Disaster". In jedem Fall wird der Kurswert der Aktie auf den aktuellen Wert geändert. Die Simulation wird durch Betätigen des Knopfes *stopButton* beendet. Auf den Arbeitsblättern 2 und 3 sind zwei (noch zu vervollständigende) Subsysteme für die Benutzerschnittstelle und den Anwendungskern des Systems angegeben. Folgende Aufgaben sind zu lösen:

- (a) Geben Sie eine 2-Schichten Architektur für das Aktiensimulations-System an. Es sind alle notwendigen Import-Abhängigkeiten einzuzeichnen sowie eine Klasse *AktienSimulation* mit der Methode *main(args: String[])* und deren Code anzugeben. Pakete der Java API brauchen nicht berücksichtigt zu werden. (2 Punkte)
- (b) Ergänzen Sie im Subsystem *aktiengui* (auf Arbeitsblatt 2) den Code der Operation *actionPerformed* an der mit (A) gekennzeichneten Stelle. (Zur Umwandlung von Strings in Double-Werte kann die Java-Methode *Double.parseDouble* verwendet werden.) (3 Punkte)

Die Meldung eines Disasters vom Anwendungskern an die Benutzerschnittstelle soll mittels indirekter Kommunikation durch Beobachter erfolgen.

- (c) Fügen Sie in beiden Subsystemen die dazu benötigten Klassen bzw. Interfaces einschließlich Vererbungs- und Realisierungsbeziehungen ein. Folgende Operationen sind dabei an den richtigen Stellen einzutragen und gegebenenfalls durch Angabe des Codes zu implementieren: *setChanged()*, *update(o: Observable, arg: Object)*, *notifyObservers()*, *addObserver(o: Observer)*. (4 Punkte)
- (d) Ergänzen Sie den Code des Konstruktors der Klasse *AktienFrame* an der mit (B) gekennzeichneten Stelle, so dass die indirekte Kommunikation wirksam wird. (1 Punkt)
- (e) Fügen Sie den Code der Operation *meldeDisaster* der Klasse *Aktienmanager* an der mit (C) gekennzeichneten Stelle ein. (1 Punkte)
- (f) Fügen Sie den Code der Operation *changeKurs* der Klasse *Aktie* an der mit (D) gekennzeichneten Stelle ein. (2 Punkte)

Arbeitsblatt 1

