

Name:	Matrikel-Nr.:
-------	---------------

Aufgabe 1: Verschiedenes (11 Punkte)

- a) Erläutern Sie die Funktion der Übergangs-Klassen (Boundary) in der Drei-Schichten-Architektur.
- b) Erläutern Sie die Begriffe Aggregation und Komposition und beschreiben Sie den Unterschied zwischen ihnen.

Name:	Matrikel-Nr.:
-------	---------------

- c) Beurteilen Sie, ob die nachfolgenden Aussagen richtig oder falsch sind. Beachten Sie, dass falsche Kreuze Punktabzug geben! Sie können in diesem Aufgabenteil nicht weniger als 0 Punkte erreichen.

Diese Aussage ist ...	richtig	falsch
Ein Softwaresystem wird als <i>zuverlässig</i> bezeichnet, wenn es sich auch durch fehlerhafte Benutzereingaben nicht aus dem Tritt bringen lässt.		
Wenn an einem Use Case zwei Aktoren beteiligt sind, braucht man in der Realität mindestens zwei Personen, um den Use Case durchzuführen.		
Wenn ein Modell des Problembereichs <i>präzise</i> ist, dann lässt es nur die Modellierung von Situationen zu, die in der Realität relevant sind.		
Abstrakte Klassen können Basis für Vererbung sein, sie können aber nicht selbst von anderen Klassen erben.		
Aktionen einer Aktivität können selbst wieder Aktivitäten sein.		
Produktcharakteristiken von Typ LEGAL sind erfüllt, wenn das fertige System nur Programmcode enthält, der entsprechend der für die Programmiersprache gültigen Konventionen erstellt wurde.		
Ein globaler Datenraum wirkt sich negativ auf die Wartbarkeit eines Softwaresystems aus.		
Wenn ein Statechart und ein Sequenzdiagramm konsistent sind, dann muss die im Sequenzdiagramm beschriebene Aufrufreihenfolge vom Statechart erlaubt sein.		
Beim Testen ist die starke Trennung zwischen Konstruktion und Test des Softwaresystems im V-Modell ein Vorteil		
Äquivalenzklassen dienen dazu, die Ein- und Ausgabemenge eines Testobjekts so zu unterteilen, dass alle Werte aus einer Klasse zu einem äquivalenten Verhalten des Testobjekts führen.		

Name:	Matrikel-Nr.:
-------	---------------

Aufgabe 2: Modell des Problembereichs (20 Punkte)

Geben Sie ein Modell des Problembereichs an, das den in der Referenz enthaltenen Text so präzise und vollständig wie möglich modelliert.

Name:	Matrikel-Nr.:
-------	---------------

Achtung: Platz für einen zweiten Versuch. Nicht doppelt bearbeiten! Lösung eindeutig markieren!

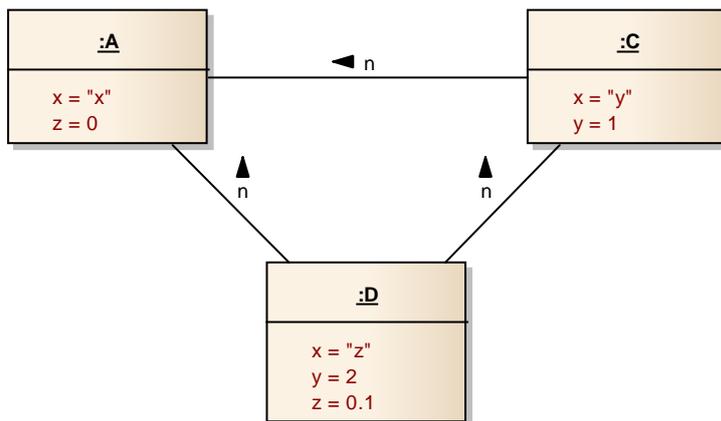
Name:

Matrikel-Nr.:

Aufgabe 3: Klassen- und Objektdiagramme (9 Punkte)

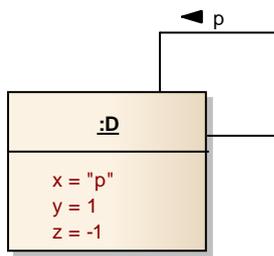
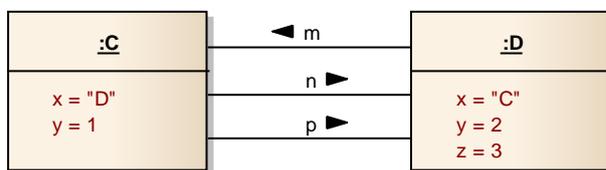
Prüfen Sie die folgenden Objektdiagramme auf Konsistenz mit dem Klassendiagramm in der Referenz. Wenn ein Objektdiagramm nicht konsistent ist, geben Sie alle Stellen an, an denen die Konsistenz verletzt ist, und erläutern Sie die jeweilige Verletzung kurz.

Objektdiagramm1:



Name:

Matrikel-Nr.:

Objektdiagramm 2:**Objektdiagramm 3:**

Name:	Matrikel-Nr.:
-------	---------------

Aufgabe 4: Aktivitätsdiagramme (16 Punkte)

Die Tabellen in der Referenz stellen den Use Case „Zigaretten kaufen“ inkl. drei Alternativszenarien dar. Erstellen Sie auf dieser Basis ein Aktivitätsdiagramm. Verwenden Sie Nebenläufigkeit, wo dies möglich ist.

Hinweis: Benutzen Sie zur Bezeichnung der Aktionen die Schrittnummierungen. Sie brauchen keine Swimlanes anzugeben.

Name:	Matrikel-Nr.:
-------	---------------

Achtung: Platz für einen zweiten Versuch. Nicht doppelt bearbeiten! Lösung eindeutig markieren!

Name:	Matrikel-Nr.:
-------	---------------

Aufgabe 5: Architektur (10 Punkte)

- a) Stellen Sie den Architekturstil „Model View Controller“ visuell dar und beschriften Sie die Elemente und Beziehungen (maximal 1-2 Wörter pro Element/Beziehung).

- b) Beschreiben Sie textuell das Zusammenspiel von Model und View im Architekturstil „Model View Controller“. Benennen Sie dabei das verwendete Design Pattern.

Name:	Matrikel-Nr.:
-------	---------------

Aufgabe 6: Analyse-Tabelle (20 Punkte)

Erstellen Sie zu den zwei Analysesequenzdiagrammen aus der Referenz eine Analysetabelle, die die Eigenschaften der genutzten Klassen zusammenführt. Geben Sie zusätzlich zum Klassennamen immer auch das Stereotyp-Symbol an.

Klassen	Aufgaben	Attribute	kennt (dauerhaft)

Name:	Matrikel-Nr.:
-------	---------------

Achtung: Platz für 2. Versuch. Nicht doppelt bearbeiten! Lösung eindeutig markieren!

Klassen	Aufgaben	Attribute	kennt (dauerhaft)

Name:	Matrikel-Nr.:
-------	---------------

Aufgabe 7: Statecharts (16 Punkte)

In der Referenz sind Informationen zu der Funktion einer Lichtanlage gegeben. Konstruieren Sie daraus ein Statechart, das alle beschriebenen Anforderungen umsetzt.

Name:	Matrikel-Nr.:
-------	---------------

Achtung: Platz für einen zweiten Versuch. Nicht doppelt bearbeiten! Lösung eindeutig markieren!

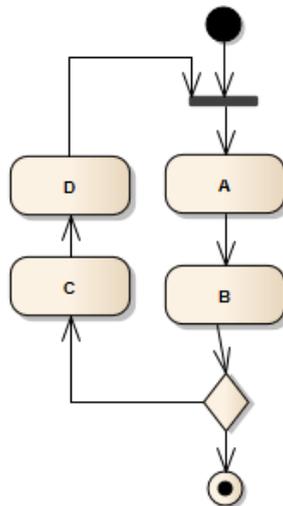
Name:

Matrikel-Nr.:

Aufgabe 8: Aktivitätsdiagramme (6 Punkte)

- a) Modellieren Sie ein Aktivitätsdiagramm, das die Actions **A**, **B**, **C**, und **D** enthält. Die Ausführung des Aktivitätsdiagramm soll genau die folgenden Ausführungsreihenfolgen der Actions unterstützen: **ABCD**, **ACBD**, **ACDB**, **CDAB**, **CADB** und **CABD**. Benutzen Sie dabei das in der Vorlesung vorgestellte Pattern.

- b) Beschreiben Sie kurz die Ausführung des folgenden Aktivitätendiagramms.



Name:

Matrikel-Nr.:

Aufgabe 9: Implementierung (12 Punkte)

Das unten angegebene Klassendiagramm soll in Java-Code übersetzt werden. Erläutern Sie stichwortartig die drei in der Vorlesung vorgestellten Möglichkeiten zur Übersetzung und diskutieren Sie sie in Bezug auf Zeit- und Platzeffizienz sowie die Erhaltung der referentiellen Integrität.

