

## Objektorientierte Modellierung Übungsblatt 3

### Aufgabe 1.

#### Lösung.

- **Sequenzdiagramm** Beschreibt den zeitlichen und logischen Nachrichtenfluss.  
**Kommunikationsdiagramm** Stellt Beziehungen zwischen Interaktionspartnern dar.  
**Zeitdiagramm** Drückt Zustandsänderungen der Interaktionspartner aus.  
**Interaktionsübersichtsdiagramm** Übersicht über verschiedene Interaktionsdiagramme.  
Primär zum Modellieren von Nachrichten zwischen verschiedenen Interaktionspartnern in bestimmten Kontexten.
- Das Sequenzdiagramm besteht aus der horizontalen Achse, an der die Interaktionspartner angeführt werden und der vertikalen Achse, die die Zeit darstellt. Es kann Lebenslinien sowie Nachrichten enthalten.
- Synchrone Nachrichten sind Nachrichten, bei denen der Sender so lange wartet und nichts tut, bis eine Antwort bei ihm angekommen ist.  
Asynchrone Nachrichten sind im Prinzip Signale, das heißt der Sender sendet seine Nachricht, wartet aber nicht explizit auf eine Antwort, sondern erledigt dazwischen zum Beispiel andere Dinge.
- Aktive Objekte haben einen eigenen Kontrollfluss, das heißt sie agieren unabhängig von anderen Objekten.  
Passive Objekte wiederum können beispielsweise von aktiven Objekten aufgerufen werden, besitzen aber keinen eigenen Kontrollfluss.

### Aufgabe 2.

#### Lösung.

- Zustandsinvarianten stellen sicher, dass ein bestimmter Zustand zu einem bestimmten Zeitpunkt erfüllt ist.  
Zeiteinschränkungen können einerseits absolut, andererseits relativ angegeben werden. Außerdem gibt es die Möglichkeit nicht nur Zeitpunkte sondern auch Zeitdauern zu modellieren.

- Für Verzweigungen und Schleifen stehen vier Operatoren zur Verfügung: alt, opt, break und loop.

Der **alt-Operator** entspricht einem if-then-else-Konstrukt. Das heißt es wird eine Bedingung beim Operator angegeben und wenn diese wahr ist, dann wird der Inhalt des Operanden ausgeführt. Wenn die Bedingung falsch ist, dann wird der Inhalt des else-Operators ausgeführt. Wichtig ist, dass hier mehrere Bedingungen angeführt werden können. Nur wenn keine Bedingung wahr ist wird der Inhalt des else-Operators ausgeführt.

Der **opt-Operator** ist im Prinzip dasselbe wie der alt-Operator, es kann aber nur eine Bedingung festgelegt werden. Wenn diese wahr ist, dann wird der Inhalt der Bedingung ausgeführt, wenn nicht, dann läuft die Interaktion einfach weiter. Die Bedingung wird dann einfach übersprungen.

Der **break-Operator** wird ähnlich wie der opt-Operator dargestellt. Der Unterschied ist der, dass der break Operator, wenn die Bedingung wahr ist, den Inhalt der Bedingung ausführt und dann aber nichts mehr ausführt und jegliche Interaktionen die danach kommen überspringt.

Der **loop-Operator** führt bestimmte Interaktionen so lange aus, bis die anfängliche Bedingung nicht mehr wahr ist.

- Für parallele Abläufe und Ordnungen stehen vier Operatoren zur Verfügung: seq, strict, par, critical.

Der **seq-Operator** stellt die Default-Ordnung dar. Hier können aber die einzelnen Nachrichten vertauscht werden, wenn es die Interaktionspartner zulassen.

Der **strict-Operator** legt fest, in welcher Reihenfolge die Nachrichten ausgetauscht werden müssen. Hier kann man die Reihenfolge auf keinen Fall verändern.

Der **par-Operator** ermöglicht es Nachrichten in irgendeiner Reihenfolge zu verschicken, solange die Nachrichten auf der lokalen Zeitachse der Interaktionspartner der durch den seq-Operator festgelegten Reihenfolge folgen.

Der **critical-Operator** legt eine Menge an Nachrichten fest, die in der festgelegten Reihenfolge ablaufen müssen und nicht unterbrochen werden dürfen.

- Für Filterungen und Zusicherungen gibt es folgende Operatoren: ignore, consider, assert, neg.

Der **ignore-Operator** gibt Nachrichten an, die auftreten aber getrost ignoriert werden können.

Der **consider-Operator** ist das Gegenstück zum ignore-Operator und wird verwendet um Nachrichten hervorzuheben.

Der **assert-Operator** besagt, dass Nachrichten genau so ausgeführt werden sollen und es keine Abweichungen geben darf.

Der **neg-Operator** drückt einen Sachverhalt aus, der nicht eintreten darf.

### Aufgabe 3.

#### Lösung.

Eine Mitarbeiterin informiert einen Gast über die Zimmerpreise.	<input checked="" type="checkbox"/> synchron	<input type="checkbox"/> asynchron
Auf einem Infoscreen im Wellnessbereich werden die neuesten Massagen und Behandlungen eingeblendet.	<input type="checkbox"/> synchron	<input checked="" type="checkbox"/> asynchron
Eine Mitarbeiterin begleitet die Hotelgäste auf ihr Zimmer.	<input checked="" type="checkbox"/> synchron	<input type="checkbox"/> asynchron
Über das Zimmerradio ertönt beruhigende Entspannungsmusik.	<input type="checkbox"/> synchron	<input checked="" type="checkbox"/> asynchron
Der Gast schaltet sämtliche Lichtquellen des Hotelzimmers per Touch-Screen ein.	<input checked="" type="checkbox"/> synchron	<input type="checkbox"/> asynchron
Ein Gast nutzt den Computer in der Hotellobby, um einem Freund ein E-Mail zu senden.	<input type="checkbox"/> synchron	<input checked="" type="checkbox"/> asynchron
Ein Gast beschwert sich per Telefon bei der Rezeption, dass ein Bademantel fehlt.	<input checked="" type="checkbox"/> synchron	<input type="checkbox"/> asynchron
Ein Gast füllt einen Fragebogen zur Qualität des Hotels aus.	<input type="checkbox"/> synchron	<input checked="" type="checkbox"/> asynchron
Ein Ehepaar diskutiert, ob es zuerst die Nachmittagsjause oder den Fitnessraum in Anspruch nehmen soll.	<input checked="" type="checkbox"/> synchron	<input type="checkbox"/> asynchron
Vor der Abreise bezahlt ein Gast die Hotelrechnung mit Kreditkarte.	<input checked="" type="checkbox"/> synchron	<input type="checkbox"/> asynchron

### Aufgabe 4.

#### Lösung.

- a)
- Die beiden Diagramme sind **nicht äquivalent**, weil bei a) die Nachricht b auf jeden Fall verschickt wird und bei b) nur dann, wenn  $x \neq 5$  ist.
  - Die beiden Diagramme sind **äquivalent**.
  - Die beiden Diagramme sind **äquivalent**.
- b)
- In strict muss genau diese Reihenfolge eingehalten werden:  $b \rightarrow c$ .
  - Im par-Teil ist die Reihenfolge irrelevant, außer dass g nach d kommen muss und e und f direkt aufeinanderfolgend sein müssen und e nach d und vor g stehen muss.

$$d \rightarrow e \rightarrow f \rightarrow g \rightarrow h$$

$$h \rightarrow d \rightarrow e \rightarrow f \rightarrow g$$

**Aufgabe 5.****Lösung.**