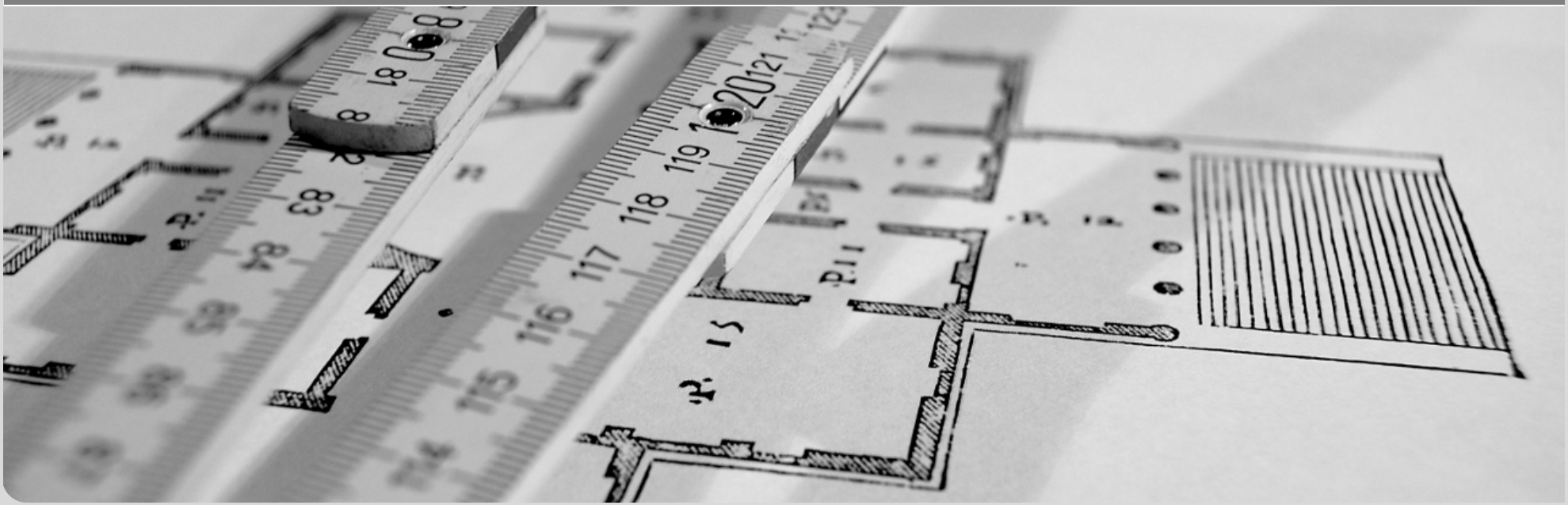
Karlsruher Institut für Technologie

Softwaretechnik 1 - 2. Tutorium

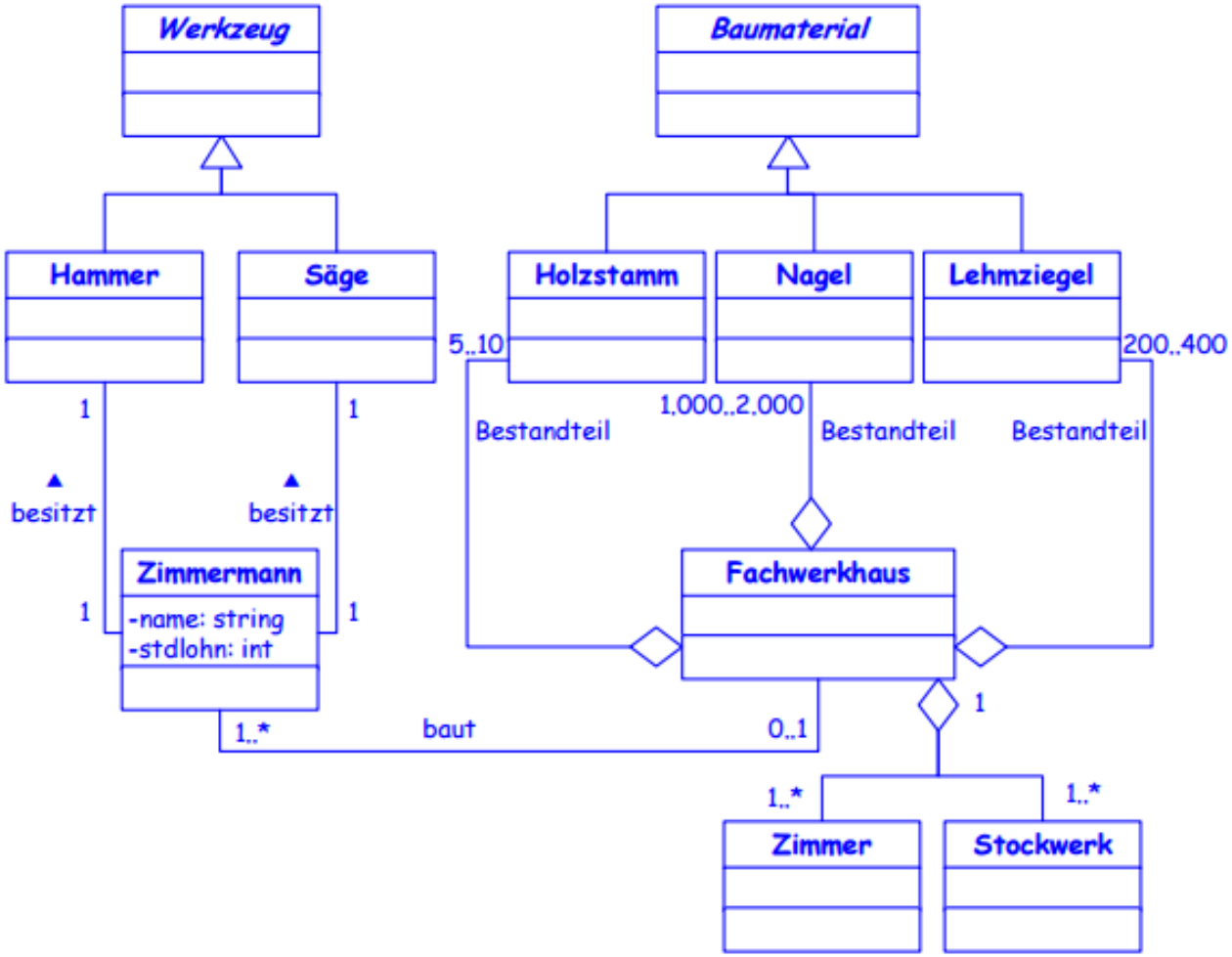
Tutorium 18

Felix Bachmann | 22.05.2018

KIT - INSTITUT FÜR PROGRAMMSTRUKTUREN UND DATENORGANISATION (IPD)

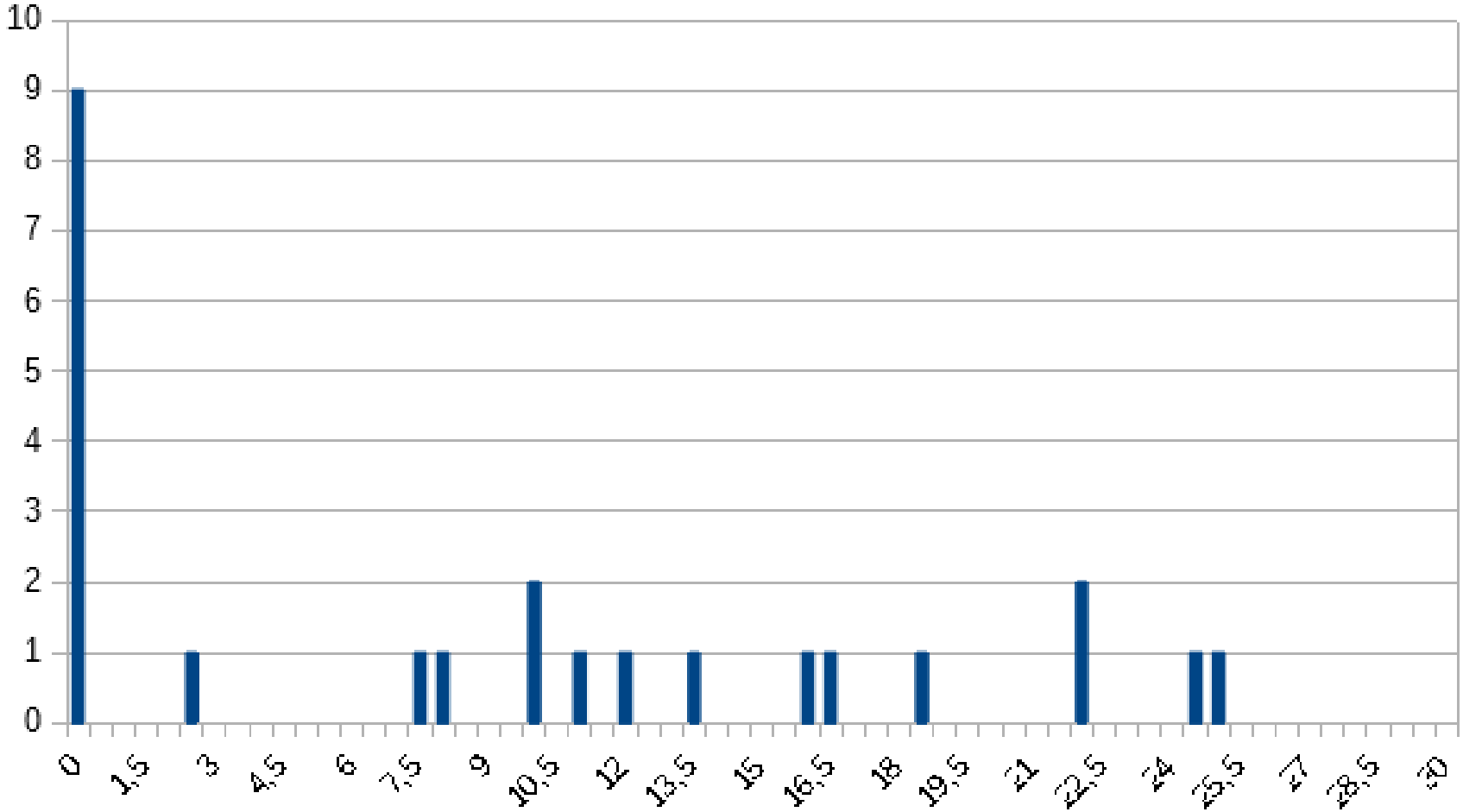


Musterlösung Klassendiagramm



- Zimmer und Stockwerk müsste wirklich eine Komposition sein

2. Übungsblatt Statistik



Häufige Fehler

Allgemein

Geben Sie Ihre Lösung mit Deckblatt (mit Name, Matrikelnummer und die Nummer Ihres Tutoriums deutlich lesbar) ab, damit Ihr Tutor Korrekturhinweise und Ihre Punkte notieren kann. Werfen Sie es in die Holzkiste vor Raum 369 im Informatik-Hauptgebäude 50.34. Verwenden Sie ausschließlich das Deckblatt zur SWT1 aus ILIAS.

- $\Longrightarrow$ nur das offizielle Deckblatt verwenden!

Häufige Fehler

Allgemein

Geben Sie Ihre Lösung mit Deckblatt (mit Name, Matrikelnummer und die Nummer Ihres Tutoriums deutlich lesbar) ab, damit Ihr Tutor Korrekturhinweise und Ihre Punkte notieren kann. Werfen Sie es in die Holzkiste vor Raum 369 im Informatik-Hauptgebäude 50.34. Verwenden Sie ausschließlich das Deckblatt zur SWT1 aus ILIAS.

- $\Longrightarrow$ nur das offizielle Deckblatt verwenden!
- häufigster Fehler: Aufgaben nicht abgegeben

Häufige Fehler

Aufgabe 1 (Lastenheft)

- Unnötiges aus Vorlage durfte man löschen (z.B. “Siehe https://en.wikibooks.org/wiki/LaTeX/Glossary“)

Häufige Fehler

Aufgabe 1 (Lastenheft)

- Unnötiges aus Vorlage durfte man löschen (z.B. “Siehe https://en.wikibooks.org/wiki/LaTeX/Glossary“)
- “relevante Daten” unpräzise

Häufige Fehler

Aufgabe 1 (Lastenheft)

- Unnötiges aus Vorlage durfte man löschen (z.B. “Siehe https://en.wikibooks.org/wiki/LaTeX/Glossary“)
- “relevante Daten” unpräzise
- Szenarien zu allgemein

Häufige Fehler

Aufgabe 1 (Lastenheft)

- Unnötiges aus Vorlage durfte man löschen (z.B. “Siehe https://en.wikibooks.org/wiki/LaTeX/Glossary“)
- “relevante Daten” unpräzise
- Szenarien zu allgemein
- Diagramm
  - Server ist Teil des Systems
  - direkte Verbindung zwischen Anwendungsfällen nicht definiert

Häufige Fehler

Aufgabe 2 (Klassendiagramm)

- äußere Form, UML-Syntax

Häufige Fehler

Aufgabe 2 (Klassendiagramm)

- äußere Form, UML-Syntax
- Farbtiefe sollte Enum sein

Häufige Fehler

Aufgabe 2 (Klassendiagramm)

- äußere Form, UML-Syntax
- Farbtiefe sollte Enum sein
- Assoziation vs. Methode

Häufige Fehler

Aufgabe 2 (Klassendiagramm)

- äußere Form, UML-Syntax
- Farbtiefe sollte Enum sein
- Assoziation vs. Methode
- bei Collections immer []-Schreibweise nutzen, nicht ArrayList$< XY >$

Häufige Fehler

Aufgabe 3 (Durchführbarkeitsanalyse)

- nur Stichpunkte

Häufige Fehler

Aufgabe 3 (Durchführbarkeitsanalyse)

- nur Stichpunkte
- Fragen beantworten, nicht stellen!
  z.B. “Es werden 3 Java-Entwickler benötigt.“ $\Longrightarrow$ “Da wir 5 zur Zeit untätige Java-Entwickler in der Firma haben, ist das Projekt aus personeller Sicht für die Pear Corp. durchführbar.“

Häufige Fehler

Aufgabe 4 (Shutterpile programmieren)

- Alpha-Kanal nicht sichergestellt (im Supplier)

Häufige Fehler

Aufgabe 4 (Shutterpile programmieren)

- Alpha-Kanal nicht sichergestellt (im Supplier)
- die aufwendigeren Sachen nicht abgegeben

Häufige Fehler

Aufgabe 4 (Shutterpile programmieren)

- Alpha-Kanal nicht sichergestellt (im Supplier)
- die aufwendigeren Sachen nicht abgegeben
- Testfälle unzureichend

Aufgabe 5 (Kommandozeilen-Tool)

- 4 Abgaben . . .
- kleine Fehler in pom (dependencies kopieren!)

Aufgabe 6 (Farb-Wasserzeichen)

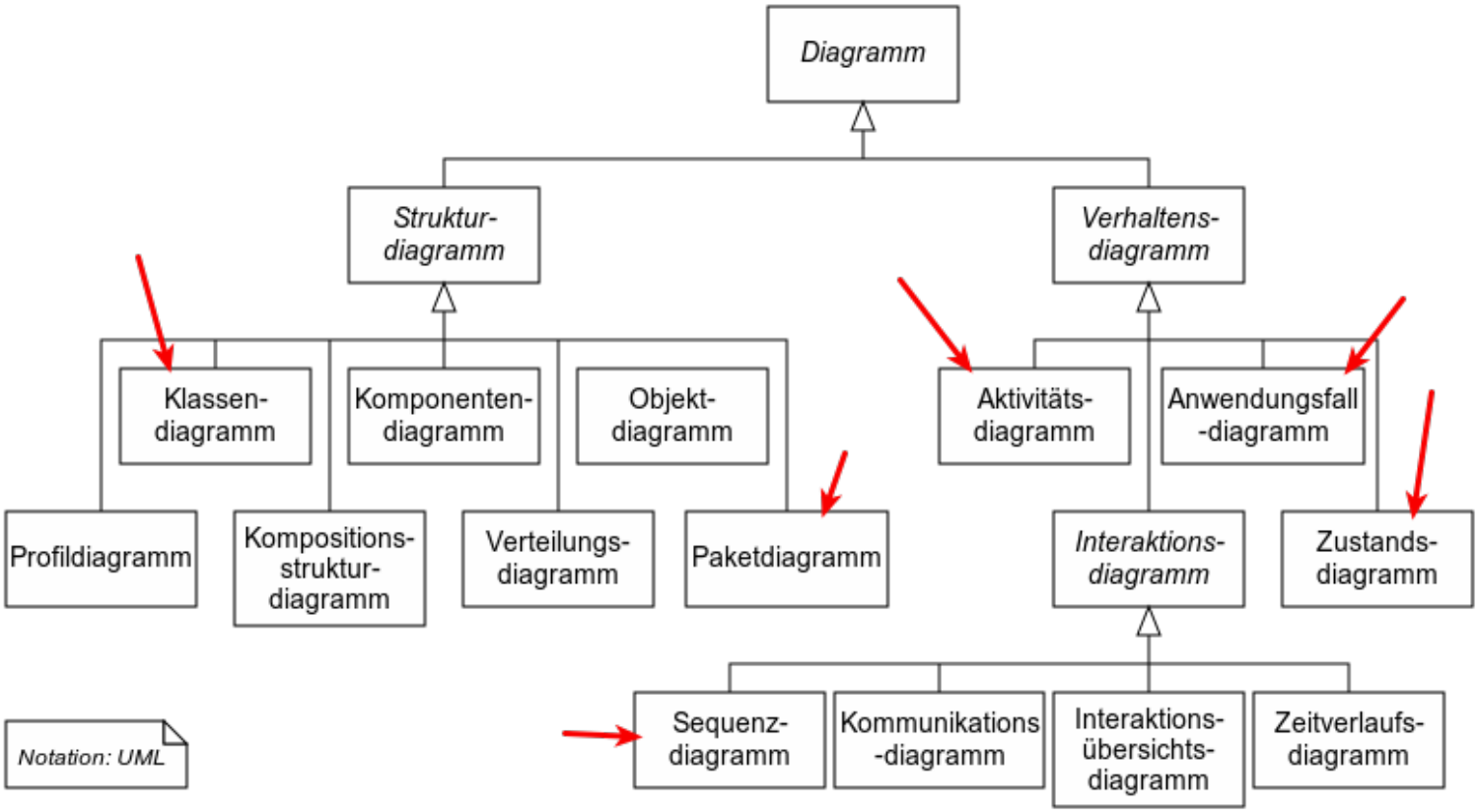
- 3 Abgaben . . .

Wo sind wir? Pflichtenheft!

1. Zielbestimmung
2. Produkteinsatz
3. Produktumgebung
4. Funktionale Anforderungen
5. Produktdaten
6. Nichtfunktionale Anforderungen
7. Globale Testfälle
8. Systemmodelle
   - Szenarien
   - Anwendungsfälle
   - Objektmodelle $\Longrightarrow$ UML-Klassendiagramme (letztes mal)
   - **Dynamische Modelle**
     - UML-Zustandsdiagramm
     - UML-Aktivitätsdiagramm
     - UML-Sequenzdiagramm

     } Heute!
   - Benutzerschnittstelle $\Longrightarrow$ Zeichnungen/Screenshots
9. Glossar

Begriffsklärung



Zustandsdiagramm - Allgemein

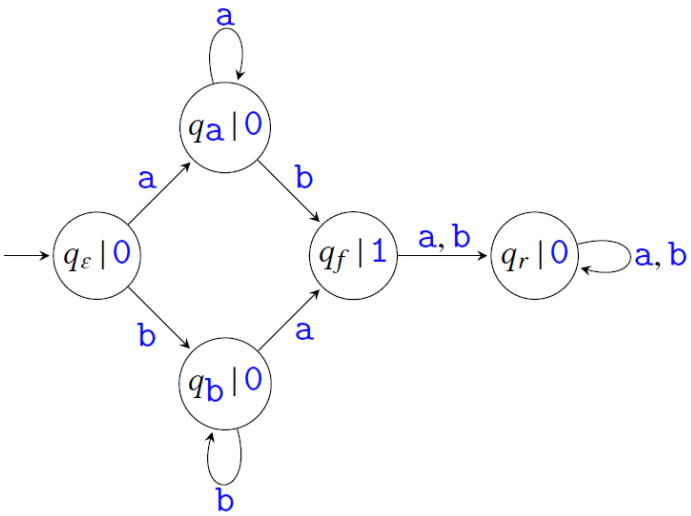
Wozu braucht man das?

Zustandsdiagramm - Allgemein

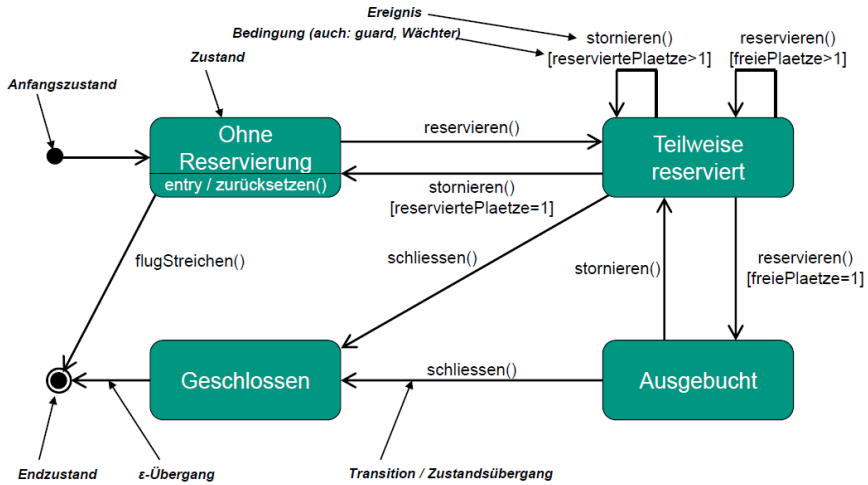
Wozu braucht man das?

- Zustand **eines Objektes** beschreiben
- Zustandsüberführungsfunktion?

Zustandsdiagramm ≈ endlicher Automat

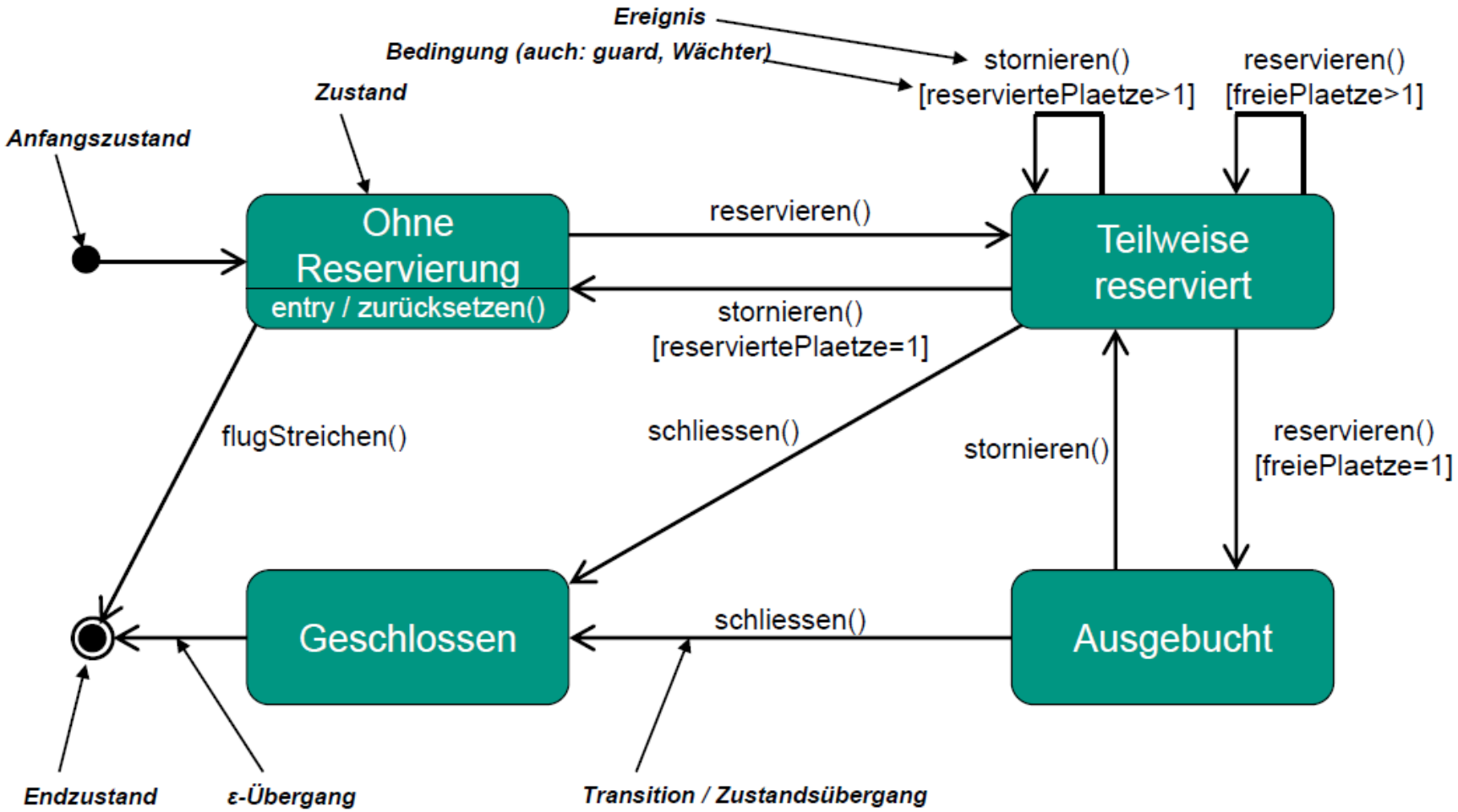


(a) GBI: DEA

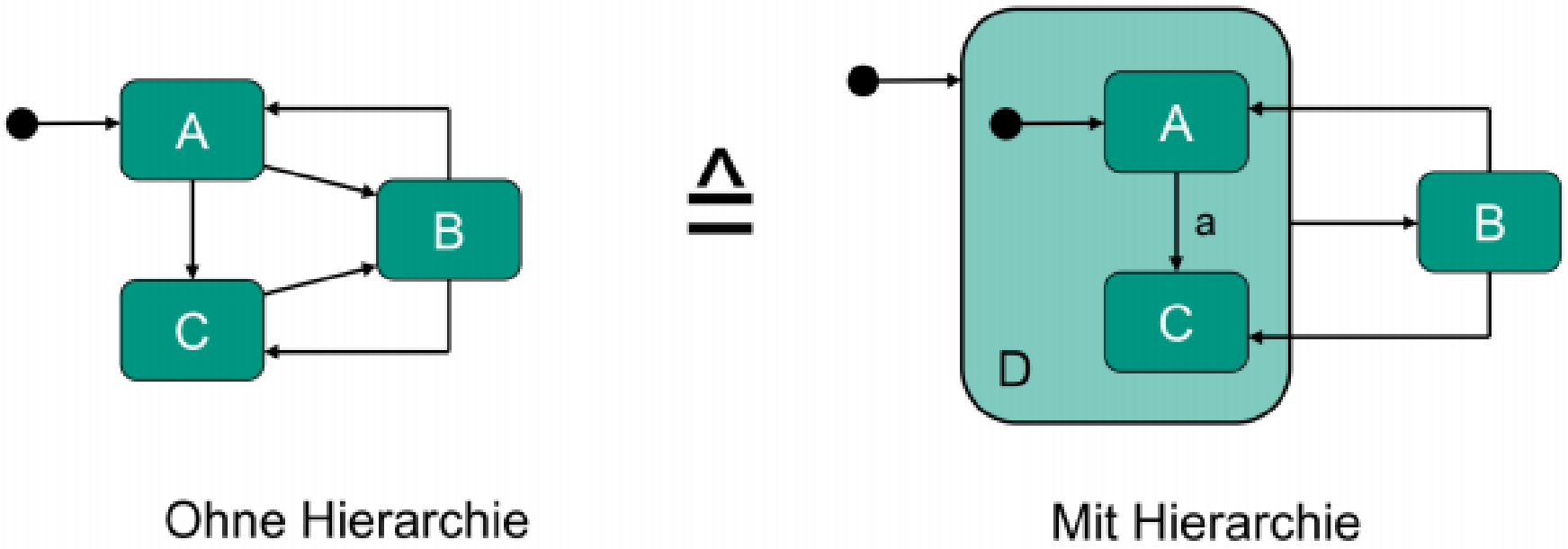


(b) SWT: Zustandsdiagramm

Zustandsdiagramm: Syntax

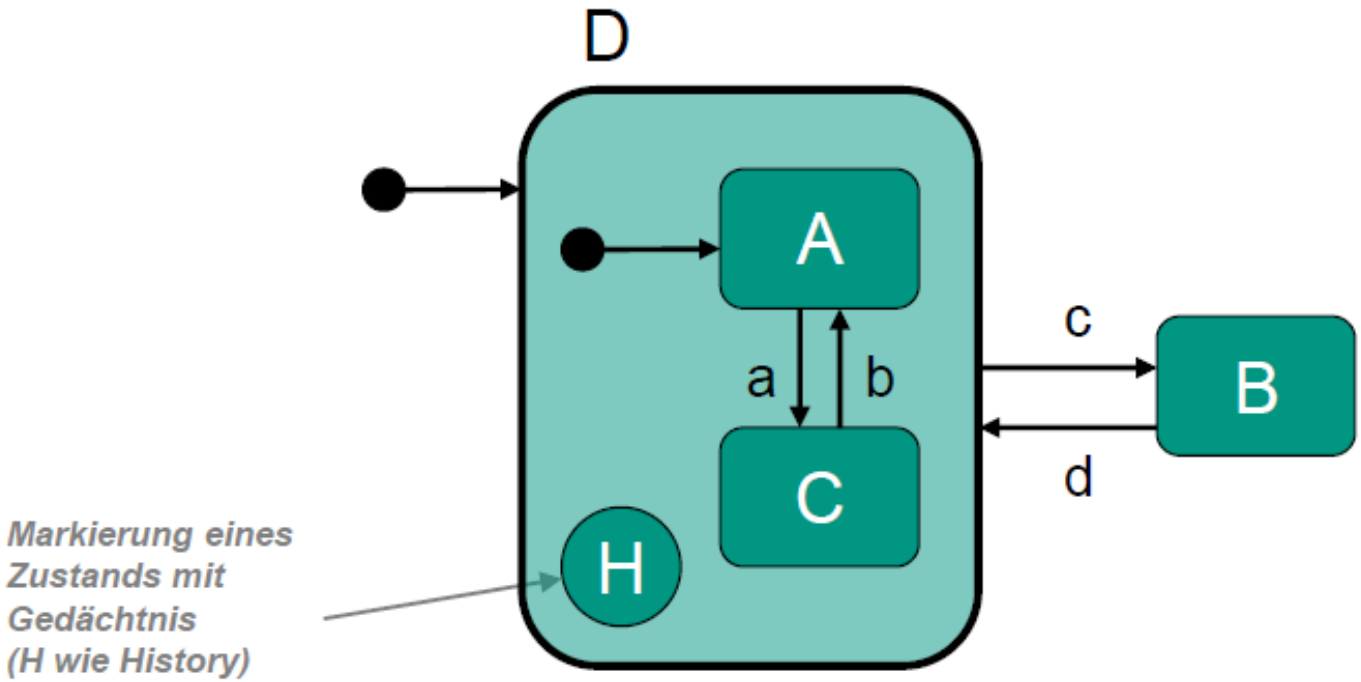


Zustandsdiagramm: Hierarchie



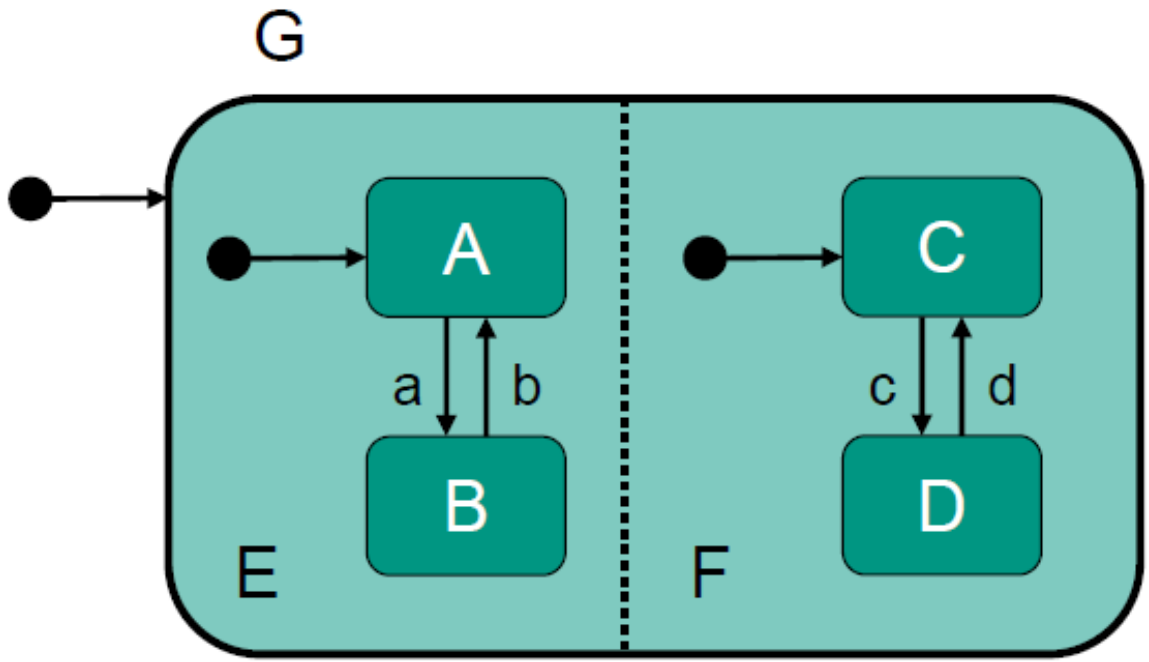
Zustandsdiagramm: Hierarchie - History

- History-Element, damit sich Hierarchie den letzten Zustand merkt



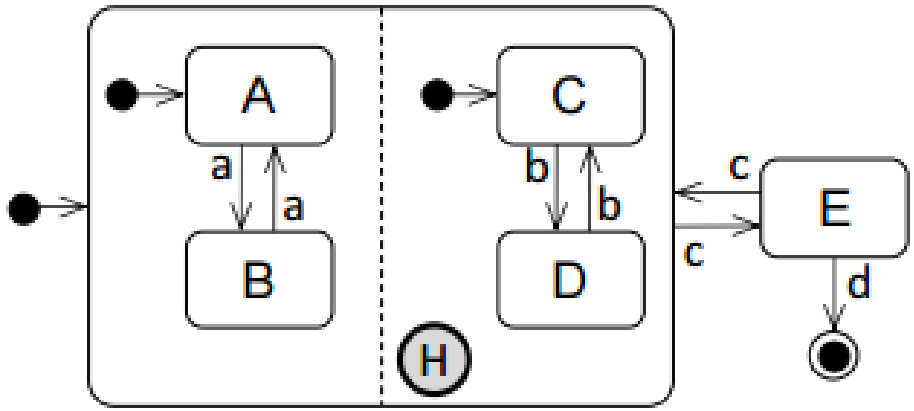
Zustandsdiagramm: Nebenläufigkeit

- mehrere Zustandsdiagramme in einem



Klausuraufgabe SS09

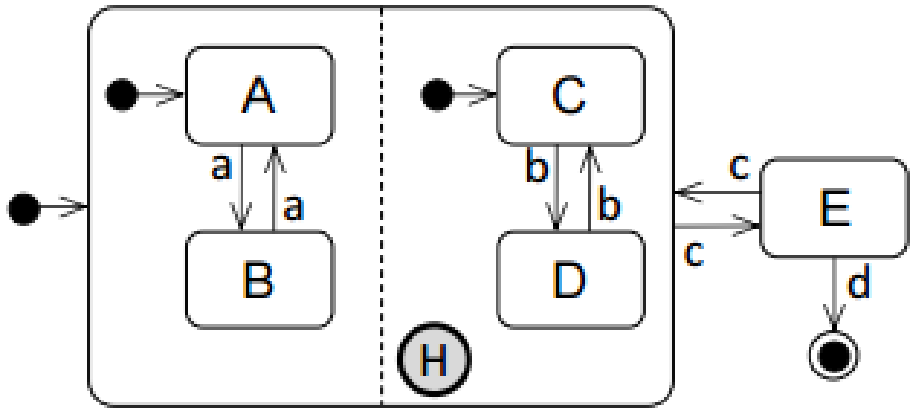
Gegeben ist der folgende UML-Zustandsautomat. Geben Sie an, in welcher Zustandskombination sich der Zustandsautomat, jeweils ausgehend vom Startzustand, nach den beiden Eingabefolgen befindet.



- a, b, c, c

Klausuraufgabe SS09

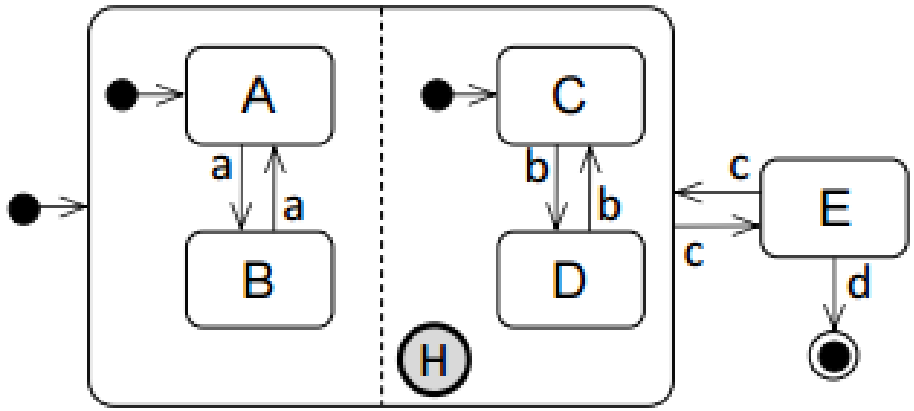
Gegeben ist der folgende UML-Zustandsautomat. Geben Sie an, in welcher Zustandskombination sich der Zustandsautomat, jeweils ausgehend vom Startzustand, nach den beiden Eingabefolgen befindet.



- a, b, c, c $\Longrightarrow$ AxD
- c, c, a, b, b, a, c, c, a

Klausuraufgabe SS09

Gegeben ist der folgende UML-Zustandsautomat. Geben Sie an, in welcher Zustandskombination sich der Zustandsautomat, jeweils ausgehend vom Startzustand, nach den beiden Eingabefolgen befindet.



- a, b, c, c $\Longrightarrow$ AxD
- c, c, a, b, b, a, c, c, a $\Longrightarrow$ BxC

Aktivitätsdiagramm - Allgemein

Wozu braucht man das?

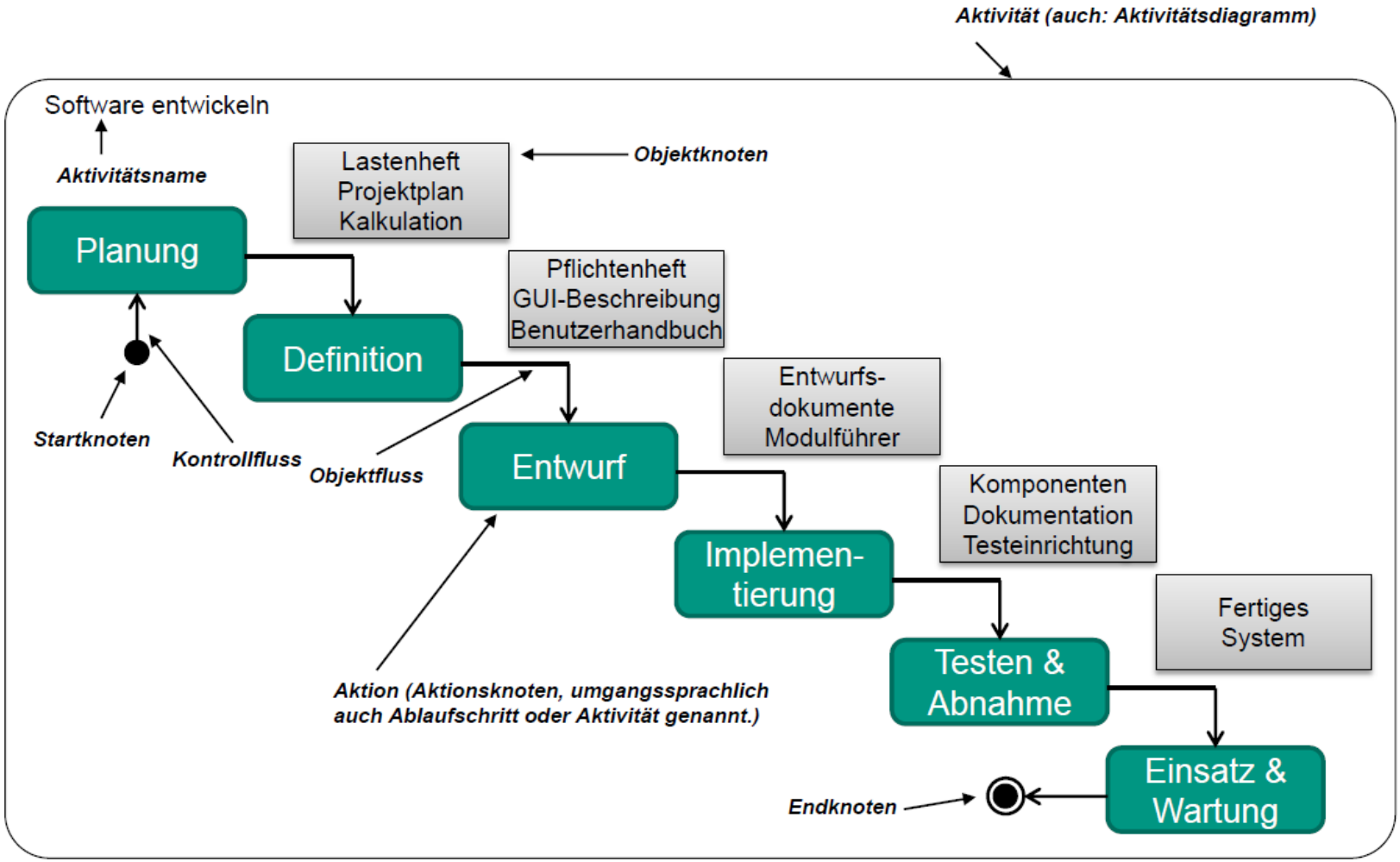
Aktivitätsdiagramm - Allgemein

Wozu braucht man das?

- Ablaufbeschreibungen (Kontrollfluss, Objektfluss)
- i.A. **mehrere verschiedene** Objekte

Aktivitätsdiagramm - Beispiel

- ist ebenfalls nicht neues!

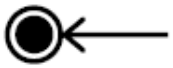


Aktivitätsdiagramm - Syntax

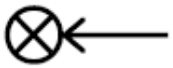
- Aktionen
  - Elementare Aktion
  - Verschachtelte Aktion

- Knoten
  - Startknoten
    - Startpunkt eines Ablaufs

  - Endknoten
    - Beendet alle Aktionen und Kontrollflüsse

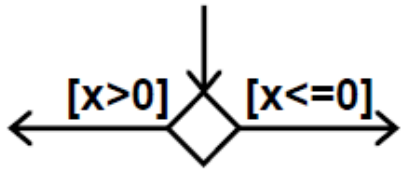


  - Ablaufende
    - Beendet einen einzelnen Objekt- und Kontrollfluss

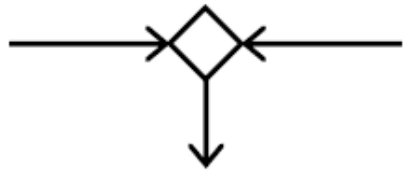


Aktivitätsdiagramm - Syntax

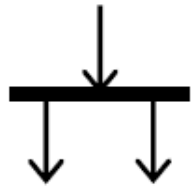
- Entscheidung
  - bedingte Verzweigung



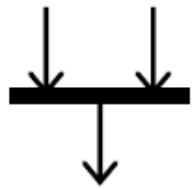
- Zusammenführung
  - „oder“-Verknüpfung



- Teilung
  - Aufteilung eines Kontrollflusses

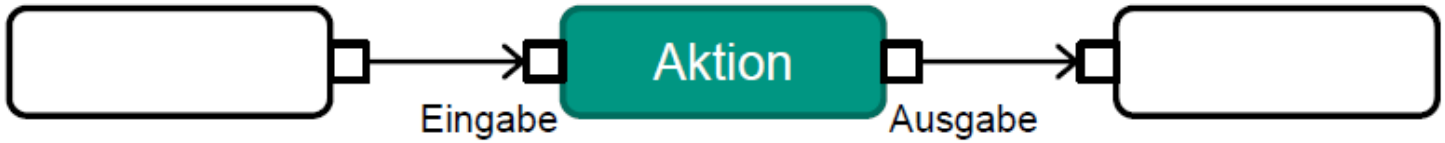


- Synchronisation
  - „und“-Verknüpfung

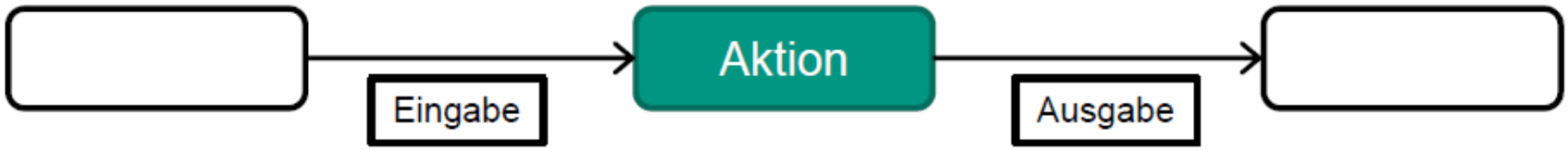


Aktivitätsdiagramm - Syntax

- Objektknoten
  - Eingabe- und Ausgabedaten einer Aktion
  - Darstellung durch Stecker (engl. *pin*)



  - Alternative Darstellungen

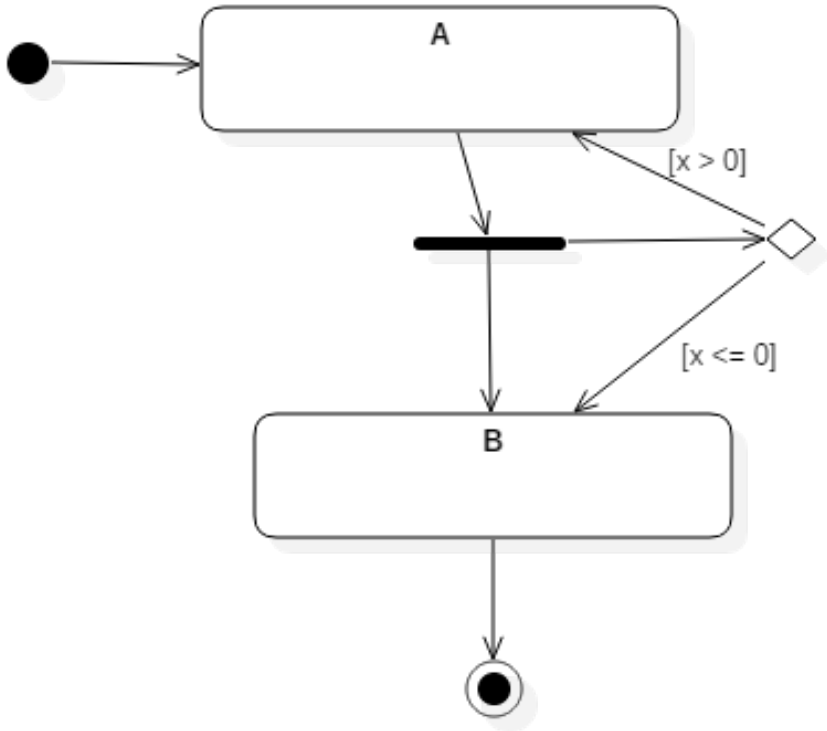


Aktivitätsdiagramm - Ablauf

- Start am Startknoten mit einer Marke
- Aktionen werden erst ausgeführt, wenn an jedem Eingang eine Marke anliegt
- wurde eine Aktion ausgeführt, erscheinen an all ihren Ausgängen Marken

Aktivitätsdiagramm - Beispiel

Wie kommt man hier zum Endknoten?



Sequenzdiagramm - Allgemein

Wozu braucht man das?

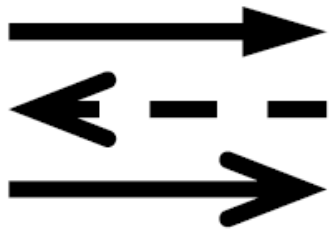
Sequenzdiagramm - Allgemein

Wozu braucht man das?

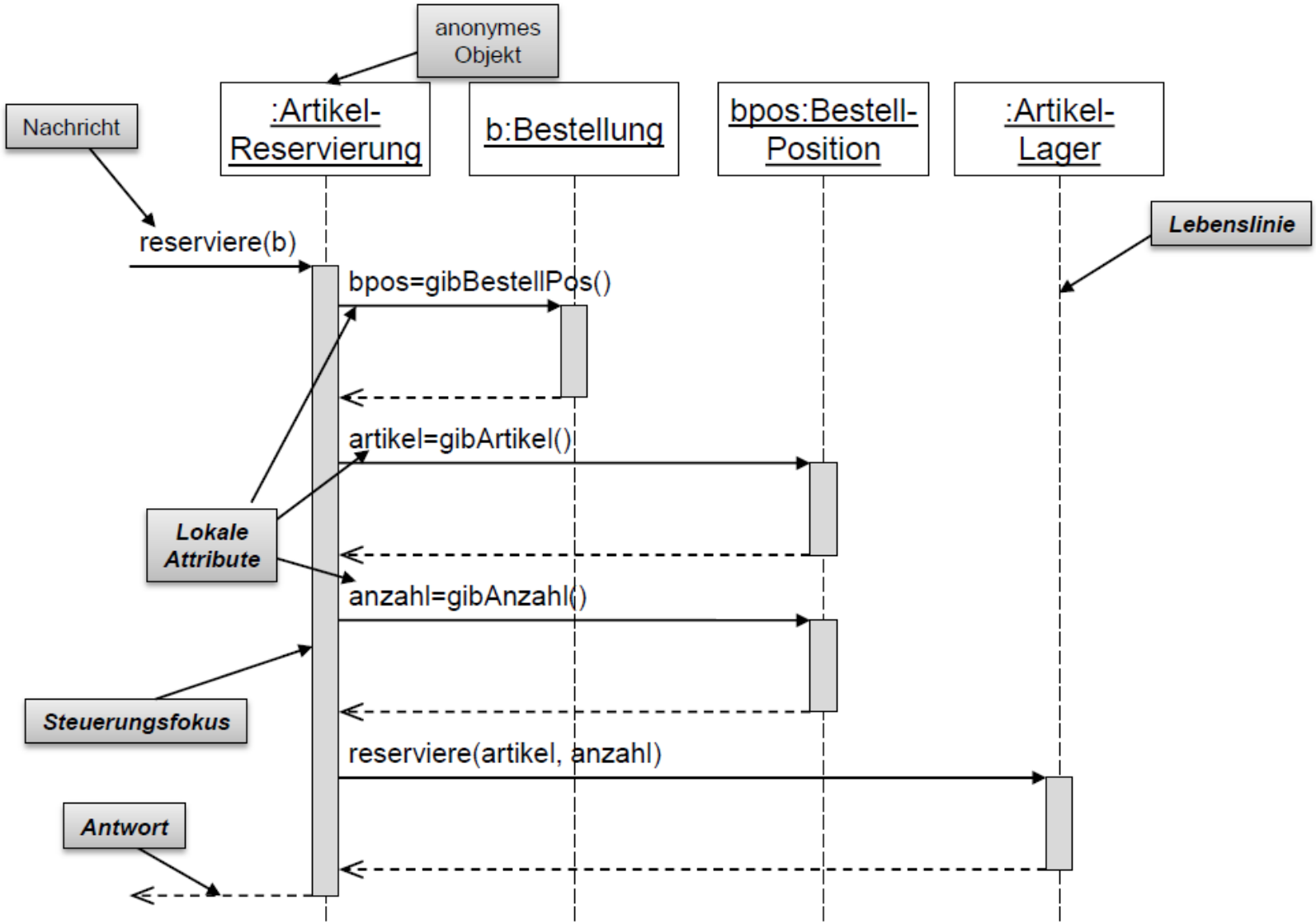
- stellt den möglichen Ablauf eines Anwendungsfalls dar
- **zeitlicher Verlauf** von Methodenaufrufen, Objekterstellung, Objektzerstörung

Sequenzdiagramm - Syntax

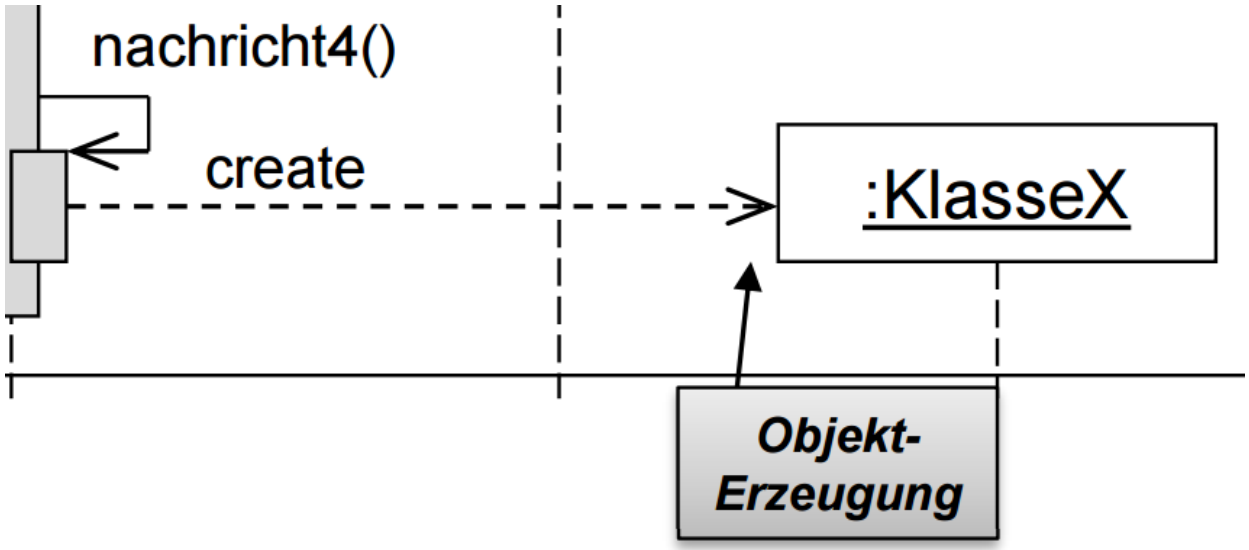
- Zeit verläuft von oben nach unten
- Lebenslinie
  - gestrichelte senkrechte Linie
  - eine pro Objekt
- Steuerungsfokus
  - dicker Balken über Lebenslinie
  - zeigt, dass Objekt gerade aktiv ist
- Nachrichtentypen
  - Synchrone Nachricht (blockierend)
  - Antwort (optional)
  - Asynchrone Nachricht



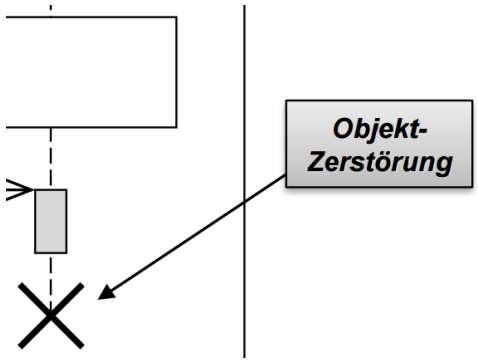
Sequenzdiagramm - Syntax



Sequenzdiagramm - Syntax

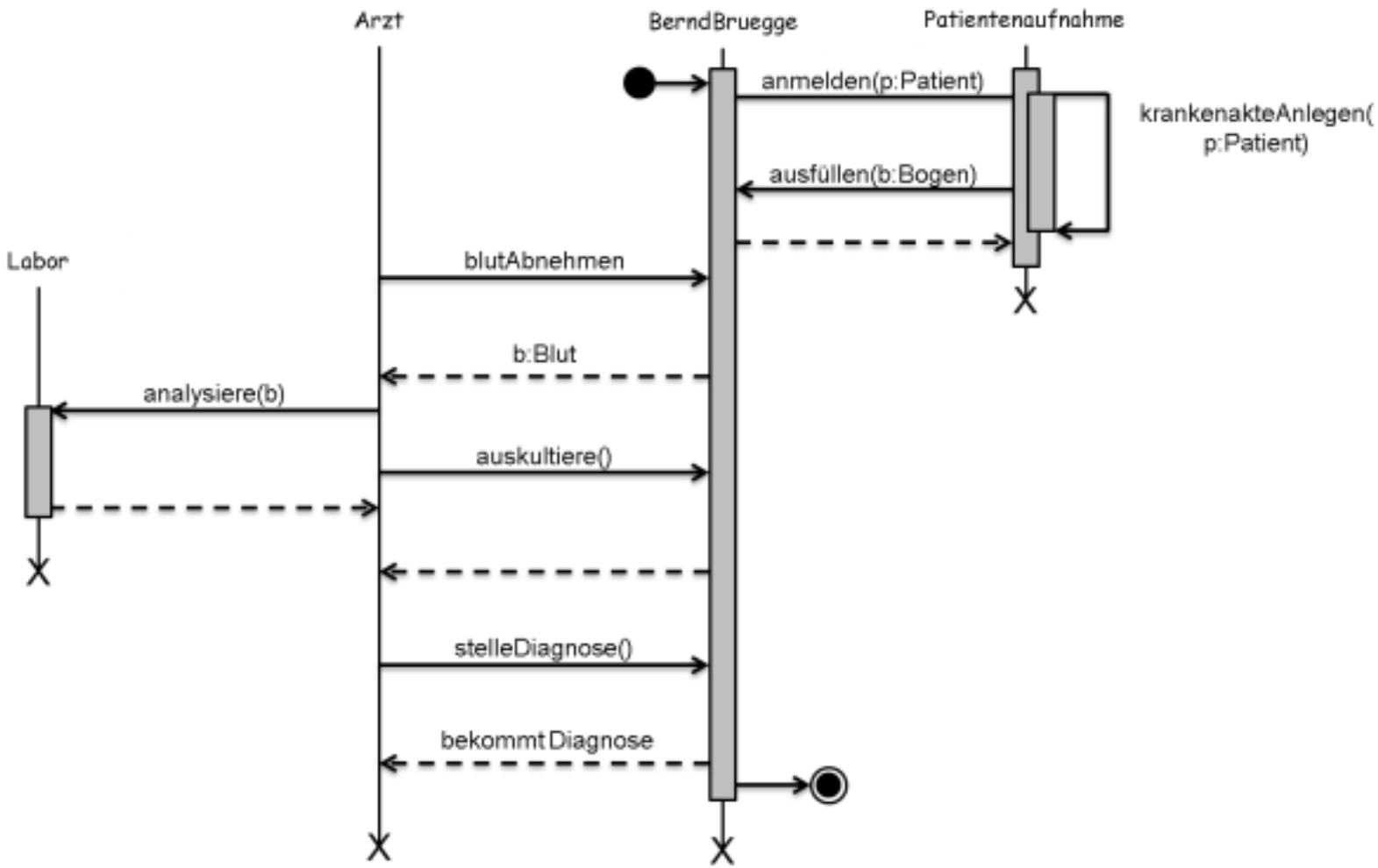


(a) Objekt-Erzeugung



(b) Objekt-Zerstörung

Klausuraufgabe SS14



Hier stimmt was nicht…

Wozu nochmal was?

Aktivitätsdiagramm

- Kontrollfluss, Objektfluss

Wozu nochmal was?

Aktivitätsdiagramm

- Kontrollfluss, Objektfluss

Zustandsdiagramm

- Zustände eines Objektes

Wozu nochmal was?

Aktivitätsdiagramm

- Kontrollfluss, Objektfluss

Zustandsdiagramm

- Zustände eines Objektes

Sequenzdiagramm

- zeitlicher Verlauf
- Aufrufe zwischen verschiedenen Objekten/Klassen

Gruppenarbeit: Jetzt seid ihr dran!

Ablauf:

1. Einteilung in Expertengruppen
2. Bearbeitet eure Aufgabe in der Gruppe
3. Vergleicht eure Lösung mit der Musterlösung
4. Durchmischen der Gruppen
5. Stellt den anderen eure Aufgabe & Lösung vor
   - Erklärt insbesondere, was ihr evtl. falsch gemacht habt

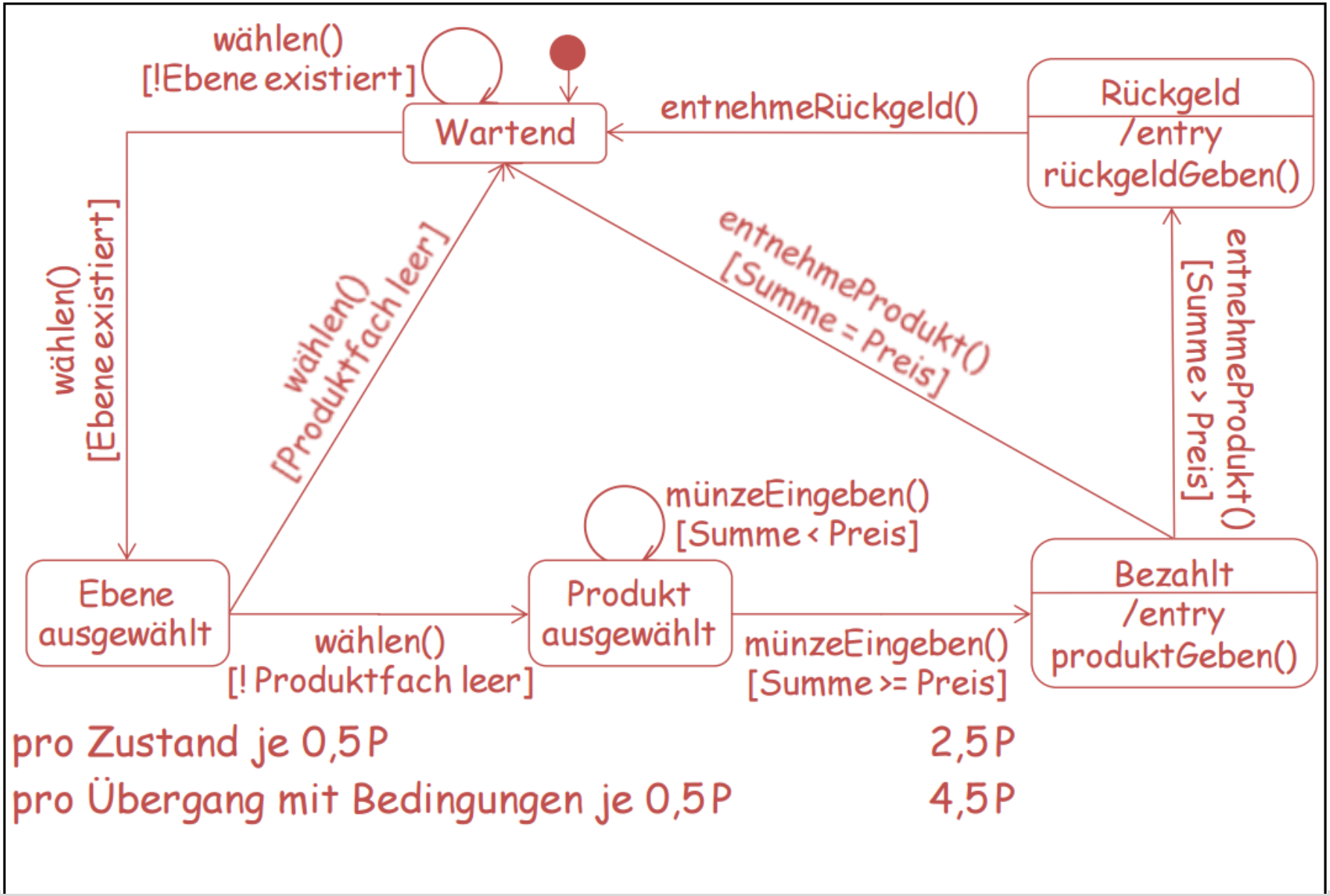
Gruppe 1: Zustandsdiagramm (Nachklausur 09)

Gegeben ist die folgende Beschreibung eines Automaten zum Verkauf von Getränken und Süßwaren:

Zu Beginn wartet der Automat auf die Auswahl des Produktes durch den Kunden. Die Produktauswahl findet in zwei Schritten statt. Zunächst wählt der Kunde die Ebene, in welcher sich das gewünschte Produkt befindet. Wählt der Kunde eine Ebene aus, die nicht existiert, wartet der Automat weiter auf die Produktauswahl. Ist die Ebene gewählt, gibt der Kunde das Fach des gewünschten Produktes an. Ist das gewählte Produktfach ausverkauft, bricht der Automat den Kaufvorgang ab und wartet erneut auf die Produktauswahl. Nach erfolgreicher Produktauswahl wirft der Kunde so lange Münzen ein, bis der eingeworfene Betrag gleich oder größer dem Preis des ausgewählten Produktes ist. Solange der Kunde nicht ausreichend Geld in den Automaten eingeworfen hat, wartet der Automat auf den Einwurf des fehlenden Geldbetrages. Hat der Kunde ausreichend Geld eingeworfen, befördert der Automat das gewählte Produkt in den Ausgabeschacht. Danach entnimmt der Kunde das Produkt. Hat der Kunde genau so viel Geld eingeworfen, wie das Produkt kostet, wartet der Automat auf die nächste Produktauswahl. Hat der Kunde das Produkt entnommen und mehr Geld eingeworfen, als das ausgewählte Produkt kostet, so gibt der Automat das Rückgeld in den Ausgabeschacht aus. Nachdem der Kunde das Rückgeld entnommen hat, wartet der Automat wieder auf die nächste Produktauswahl.

Modellieren Sie das Verhalten des Automaten wie im obigen Szenario beschrieben als UMLZustandsdiagramm. Geben Sie zu jedem Übergang das auslösende Ereignis sowie ggf. die notwendige Bedingungen an.

Gruppe 1: Musterlösung



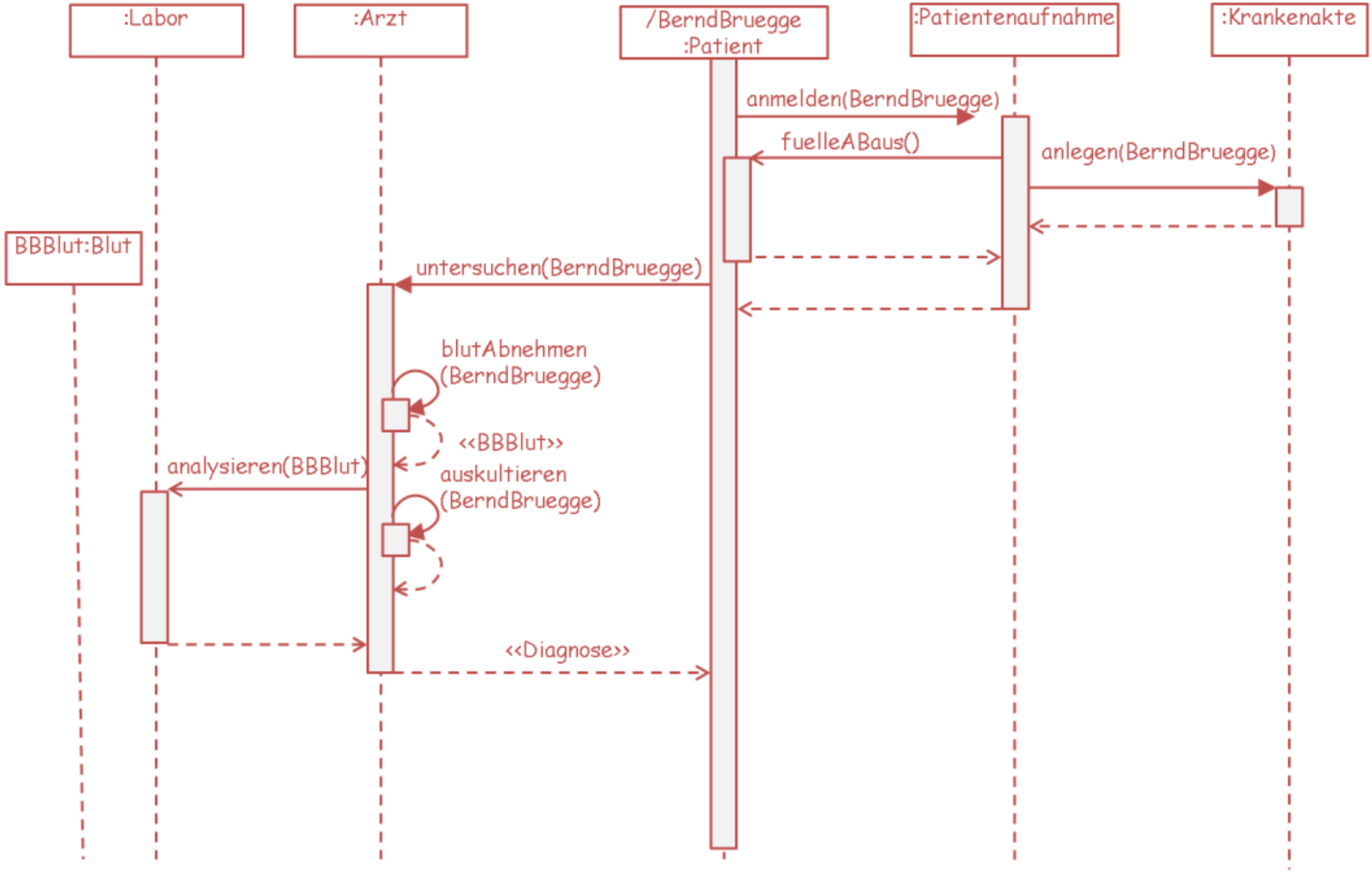
Gruppe 2: Sequenzdiagramm (Nachklausur 11)

Gegeben sei folgendes Szenario, welches eine Untersuchung in einem Krankenhaus beschreibt:

Bernd Bruegge fühlt sich nicht wohl und möchte sich untersuchen lassen, um eine Diagnose zu erhalten. Er geht dazu ins Krankenhaus und meldet sich an der Patientenaufnahme an. Während der Sachbearbeiter an der Patientenaufnahme die Krankenakte anlegt, füllt Bernd B. den Anamnesebogen aus, den ihm der Sachbearbeiter gegeben hat. Nachdem Bernd B. den ausgefüllten Bogen dem Sachbearbeiter zurückgegeben hat, wird Bernd B. vom Arzt untersucht. Dazu nimmt er Bernd B. zunächst Blut ab. Während das Labor das Blut analysiert, führt der Arzt bei Bernd B. eine Auskultation durch. Schließlich bekommt Bernd B. vom Arzt seine Diagnose.

Modellieren Sie das gegebene Szenario als UML-Sequenzdiagramm im Kasten auf der nächsten Seite (Querformat!). Verwenden Sie bei Ihrer Modellierung korrekte UML-Notation. Achten Sie bei Ihrer Modellierung darauf, auf welchen Objekten die Methoden sinnvollerweise aufgerufen werden müssen. Geben Sie ggf. Argumente der Methoden an.

Gruppe 2: Musterlösung



Gruppe 2: Musterlösung

Methodenaufruf mit Antwort:	7 ×0,5 P
Benennung der Lebenslinien:	5 ×0,5 P
Aktivitätsbalken:	5 ×0,5 P
Instanz „BerndBruegge“	1 ×0,5 P
Rest (UML-Notation, etc.)	1 ×1,0 P
Aktivitätsbalken fehlen selten:	-0,5 P
Aktivitätsbalken oft/immer:	-1,0 P
Antworten fehlen selten:	-0,5 P
Antworten oft/immer:	-1,0 P

(Alternative im Diagramm: Patientenaufnahme ruft Arzt.untersuchen() auf. Insbesondere aber ist die Methode untersuchen() keine Methode auf Patient.)

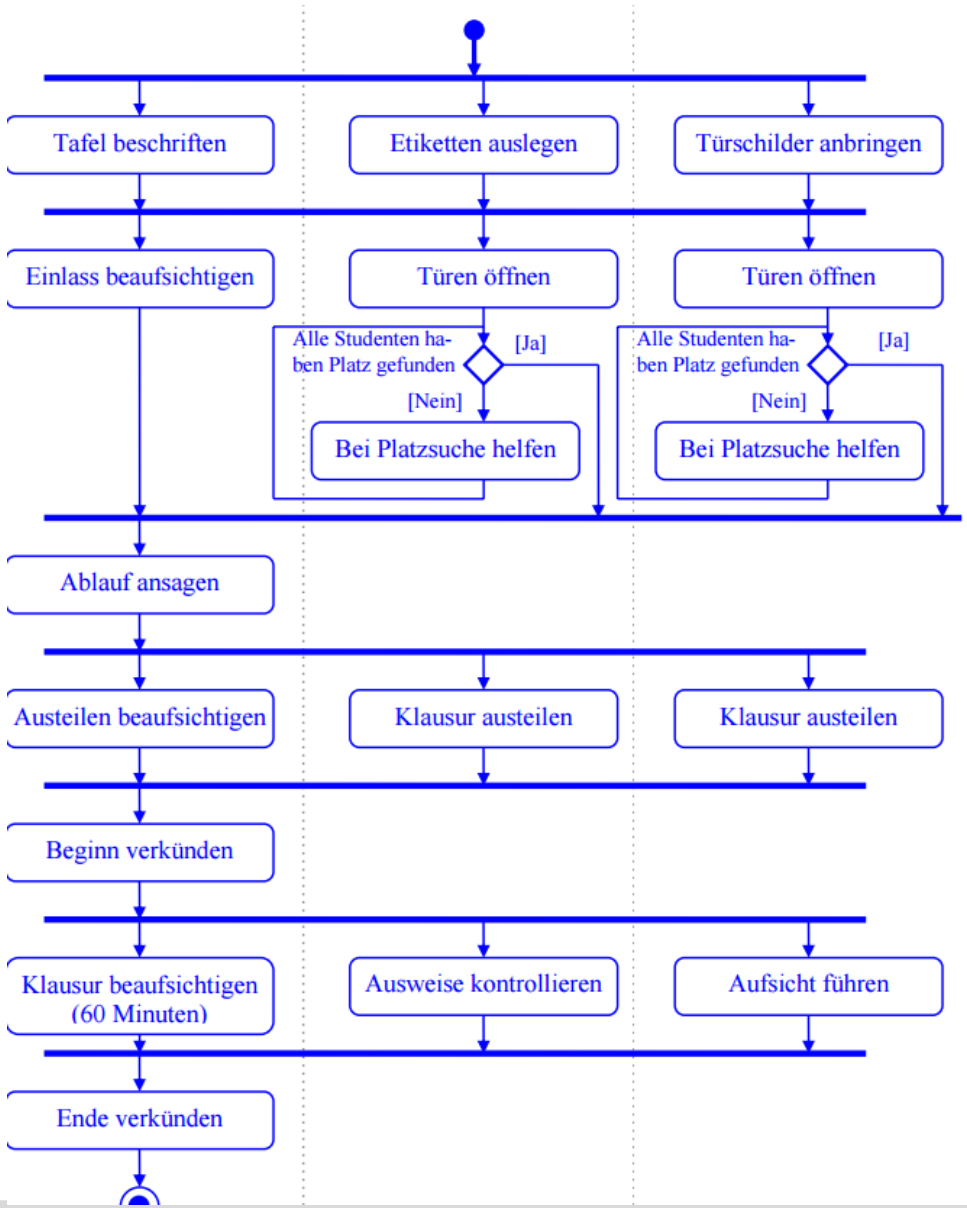
Gruppe 3: Aktivitätsdiagramm (Hauptklausur 06)

Entwerfen Sie ein Aktivitätsdiagramm, das die Durchführung einer Klausur beschreibt. Halten Sie sich dabei so eng wie möglich an die nachfolgende Beschreibung und achten Sie darauf, parallele Aktivitäten korrekt zu synchronisieren. Beginnen Sie mit der Modellierung der Aktivitäten nach Betreten des Hörsaals. Hinweis: Aktivitäten der Studenten sind nicht zu modellieren.

Nach Betreten des Hörsaals beginnt A1 die Tafel zu beschriften, während einer seiner Helfer die Etiketten auslegt und der andere die mitgebrachten Türschilder („Bitte nicht stören! Klausur!“) an den Türen anbringt. Auf den Etiketten stehen der Vor- und Nachname jedes Studenten, seine Matrikelnummer und eine fortlaufende Nummer, die später die Verwaltung der Klausur vereinfacht. Sind diese Aufgaben erledigt, kann der Einlass beginnen. Dazu öffnen die Helfer die Türen des Hörsaals und A1 beaufsichtigt den Einlass.

Solange noch nicht alle Studenten ihren Platz gefunden haben, helfen A2 und A3 bei der Platzsuche. Der Hauptverantwortliche A1 wartet ab, bis alle ihre Plätze gefunden haben. Dann erklärt er den Ablauf, beaufsichtigt das Austeilen der Klausurblätter und verkündet den Beginn der Klausur. Anschließend beaufsichtigt er 60 Minuten lang die Klausur und sagt das Ende der Klausur an. Nachdem A1 den Ablauf der Klausur angesagt hat, sind A2 und A3 für das Austeilen der Klausur, die anschließende Aufsicht zuständig. Während der Bearbeitungszeit geht zusätzlich einer von beiden herum und kontrolliert die Studentenausweise, während der andere Aufsicht führt. Während der Klausur verlässt keine Aufsicht den Hörsaal.

Gruppe 3: Musterlösung



Tipps - 3. Übungsblatt

Aufgabe 1-3: Plug-In programmieren

Tipps - 3. Übungsblatt

Aufgabe 1-3: Plug-In programmieren

- JavaDoc + CheckStyle . . .

Tipps - 3. Übungsblatt

Aufgabe 1-3: Plug-In programmieren

- JavaDoc + CheckStyle . . .
- Fügt Abhängigkeiten in die jeweilige Untermodul-pom ein

Tipps - 3. Übungsblatt

Aufgabe 1-3: Plug-In programmieren

- JavaDoc + CheckStyle . . .
- Fügt Abhängigkeiten in die jeweilige Untermodul-pom ein
- Java Swing benutzen (schaut euch die Java-Klassen JMenu und JMenuItem an)

Tipps - 3. Übungsblatt

Aufgabe 1-3: Plug-In programmieren

- JavaDoc + CheckStyle . . .
- Fügt Abhängigkeiten in die jeweilige Untermodul-pom ein
- Java Swing benutzen (schaut euch die Java-Klassen JMenu und JMenuItem an)

Aufgabe 4: Aktivitätsdiagramm

- Partitionen siehe VL

Tipps - 3. Übungsblatt

Aufgabe 5: Zustandsdiagramm

Tipps - 3. Übungsblatt

Aufgabe 5: Zustandsdiagramm

- History-Vererbung? siehe flache vs. tiefe History!

Tipps - 3. Übungsblatt

Aufgabe 5: Zustandsdiagramm

- History-Vererbung? siehe flache vs. tiefe History!

Aufgabe 6: Sequenzdiagramm

Tipps - 3. Übungsblatt

Aufgabe 5: Zustandsdiagramm

- History-Vererbung? siehe flache vs. tiefe History!

Aufgabe 6: Sequenzdiagramm

- statische Methode: Unterstreichung + Name des Objekts + Doppelpunkt beim Kasten weglassen

Tipps - 3. Übungsblatt

Aufgabe 5: Zustandsdiagramm

- History-Vererbung? siehe flache vs. tiefe History!

Aufgabe 6: Sequenzdiagramm

- statische Methode: Unterstreichung + Name des Objekts + Doppelpunkt beim Kasten weglassen
- Lebenslinie + Kasten erst hinmalen wenn Objekt schon existiert

Tipps - 3. Übungsblatt

Aufgabe 5: Zustandsdiagramm

- History-Vererbung? siehe flache vs. tiefe History!

Aufgabe 6: Sequenzdiagramm

- statische Methode: Unterstreichung + Name des Objekts + Doppelpunkt beim Kasten weglassen
- Lebenslinie + Kasten erst hinmalen wenn Objekt schon existiert

Aufgabe 7: Substitutionsprinzip

- Definition anschauen und scharf überlegen, was schief gehen kann

Denkt dran!

Abgabe

- Deadline am 30.5. um 12:00
- Aufgabe 4+5+6+7 handschriftlich (Diagramme: auf saubere Syntax achten!)
- an das Deckblatt denken

Bis dann! (dann := 05.06.18)