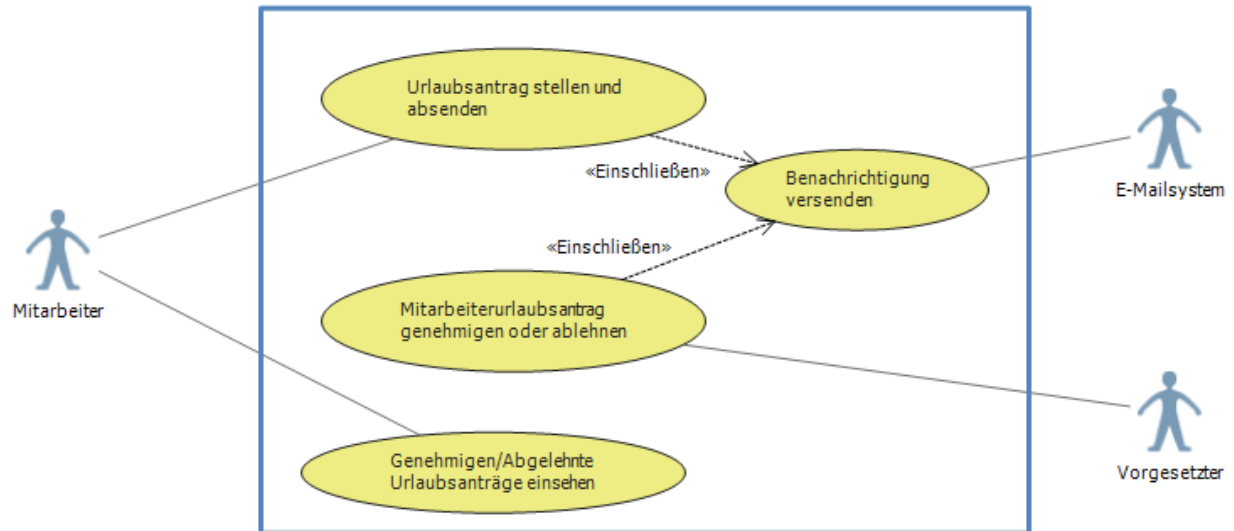


1. Aufgabe

Informieren Sie sich über den Zweck und den Aufbau von Anwendungsfalldiagrammen.
mögliche Quelle: <https://de.wikipedia.org/wiki/Anwendungsfalldiagramm>

2. Aufgabe

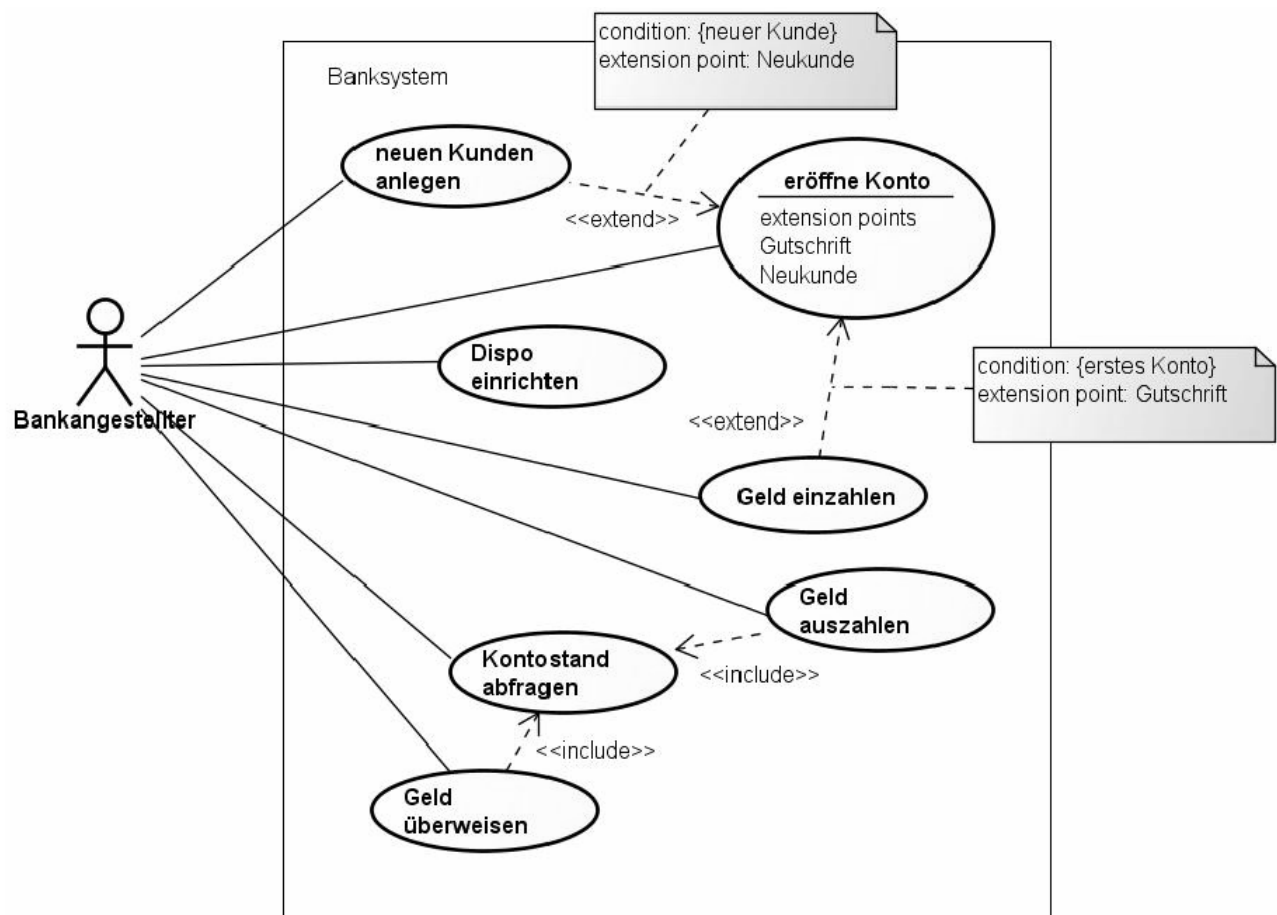
Erläutern Sie das abgebildete Anwendungsfalldiagramm vollständig.



3. Aufgabe

Es soll ein Anwendungssystem für Banken (Banksystem) geschrieben werden, das Bankangestellte bei der Verwaltung von Kunden und deren Konten unterstützt.

Es existiert bereits ein Anwendungsfalldiagramm. Erläutern Sie die dargestellten Anwendungsfälle. Gehen Sie dabei auch auf den Unterschied zwischen include- und extend-Beziehungen ein.



1. Aufgabe

Informieren Sie sich über den Zweck und den Aufbau von Sequenzdiagrammen.

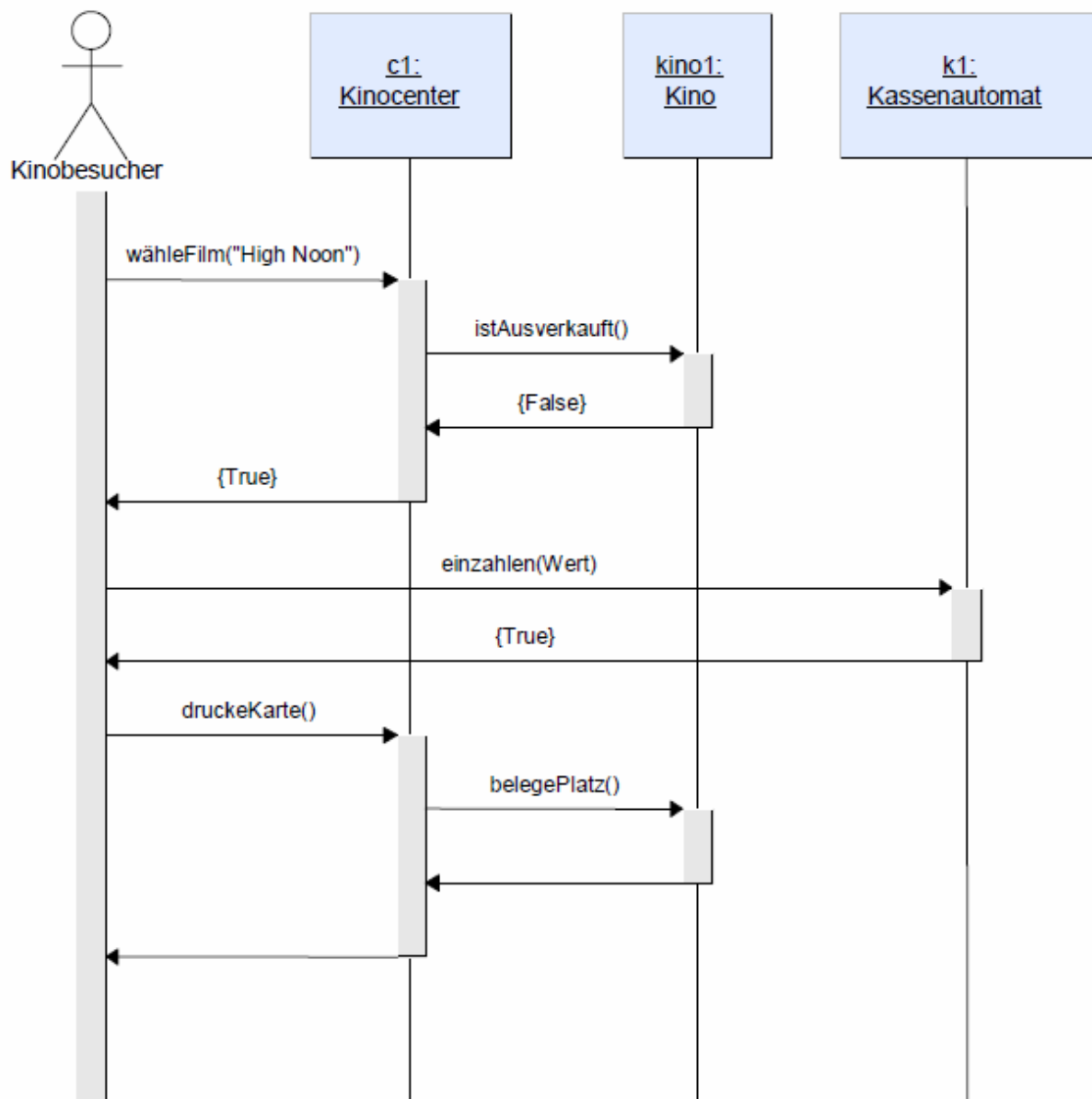
mögliche Quelle: <https://de.wikipedia.org/wiki/Sequenzdiagramm>

2. Aufgabe

Zu einem Kinocenter gehören mehrere Kinos und genau ein Kassenautomat. Das folgende Sequenzdiagramm zeigt, wie ein Kinobesucher (Kinobesucher = main-Methode einer Startklasse) einen Film auswählt, am Kassenautomat den Preis (z. B. 8,25 Euro) bezahlt und sich dann eine Karte ausdrucken lässt.

Hinweis: Die beiden nachfolgenden Teilaufgaben sollten parallel bearbeitet werden.

- Erläutern Sie ausführlich den im Sequenzdiagramm dargestellten Programmablauf.
- Erstellen Sie für das Sequenzdiagramm ein passendes Klassendiagramm. Das Klassendiagramm muss alle Informationen enthalten, welche dem Sequenzdiagramm entnommen werden können. Zusätzliche Informationen (z. B. Konstruktoren) dürfen entfallen.



1. Aufgabe

Bearbeiten Sie die komplexe Aufgabe „Bank“.

2. Aufgabe

Bearbeiten Sie die komplexe Aufgabe „Telefonbuch“.

3. Aufgabe

Bearbeiten Sie folgende Abiturprüfungen:

- 2017 / Vorschlag B (Ersttermin) / Aufgaben 1.1 bis 1.4
- 2019 / Vorschlag B (Ersttermin) / Aufgaben 1.1 und 1.2 (1.2.3 und 1.2.4 sind Zusatzaufgaben)

1. Aufgabe

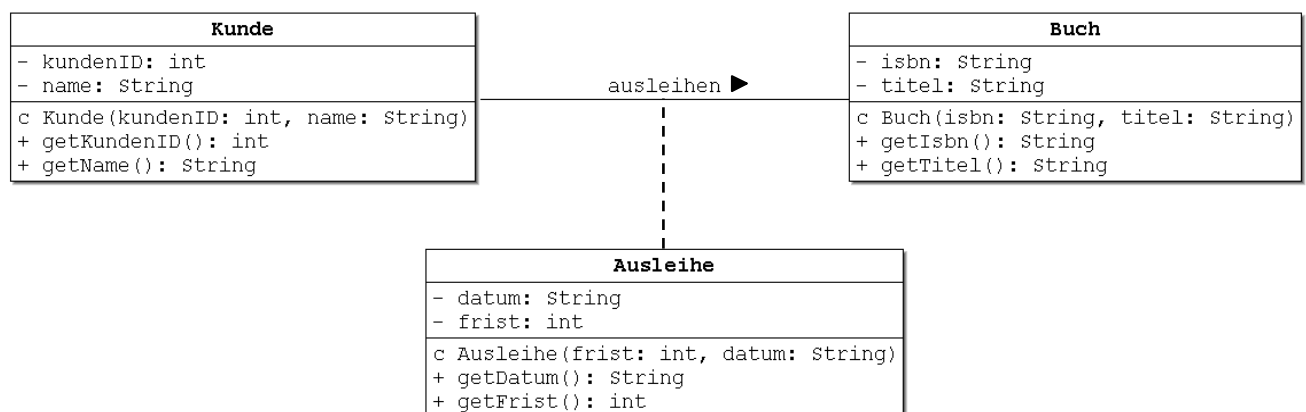
Unter den Assoziationen gibt es den Spezialfall der "qualifizierten Assoziation". Erstellen Sie für das folgende Beispiel ein Klassendiagramm und erläutern Sie daran, was man unter der "qualifizierten Assoziation" zu verstehen hat:

Die KFZ-Zulassungsstelle des Landkreises Limburg-Weilburg verwaltet viele Kraftfahrzeuge. Für jedes aktuell zugelassene KFZ existieren ein eindeutiges Kennzeichen (z. B. "LMAA-235") und das Datum der Zulassung.

2. Aufgabe

In einer Bibliothek werden viele Kunden und viele Bücher verwaltet. Von jedem Kunden werden eine Kunden-ID und ein Name gespeichert. Von jedem Buch werden die ISBN-Nummer und der Titel gespeichert. Die Kunden können sich bis zu 5 Bücher gleichzeitig (pro Tag) ausleihen. Beim Ausleihen wird das Ausleihdatum und die Frist für die Rückgabe (z. B. 30 Tage) gespeichert. Für die Bibliothek soll eine Software entwickelt werden, die nachfolgende Abbildung zeigt den Ansatz eines Klassendiagramms. Zur Beziehung ausleihen zwischen den Klassen Kunde und Buch gehört die Assoziationsklasse (Beziehungs-klass)e Ausleihe.

Hinweis: Es darf davon ausgegangen werden, dass ein Kunde nur einmal pro Tag Bücher ausleiht.



- Ignorieren Sie zunächst die Assoziationsklasse Ausleihe und tragen Sie für die Beziehung ausleihen sinnvolle Multiplizitäten in das Klassendiagramm ein.
- Erklären Sie anhand des oben abgebildeten Klassendiagramms, was man unter einer Assoziationsklasse zu verstehen hat.
- Java und viele andere Programmiersprachen kennen keine Assoziationsklassen. Diese Klassen müssen aufgelöst werden. Dies geschieht z. B. durch das Verteilen der zugehörigen Attribute/Methoden auf diejenigen Klassen, zwischen denen die Assoziation besteht. Häufig ist dies aber nicht möglich. In diesem Fall muss die Assoziationsklasse in eine eigenständige ("normale") Klasse umgewandelt werden.

Ändern Sie das gegebene Klassendiagramm so, dass es direkt in den Javacode umgesetzt werden kann.

1. Aufgabe

Mitte der 60er Jahre kam es zur so genannten Softwarekrise. Beantworten Sie dazu folgende Fragen:

- Was ist unter dem Begriff Softwarekrise zu verstehen und welche Merkmale besitzt diese?
- Welche Ursache besitzt diese Krise?
- Welche Gegenmaßnahmen wurden zur Bekämpfung dieser Krise ergriffen?

2. Aufgabe

Es gibt funktionale und nichtfunktionale Anforderungen an eine Software. Die funktionalen Anforderungen beziehen sich auf eine konkrete Software (z. B. ein Reservierungssystem für Hotels) und werden in einer Anforderungsbeschreibung formuliert.

- a) Erläutern Sie, was man unter "nichtfunktionalen Anforderungen" zu verstehen hat.
- b) Nennen Sie vier nichtfunktionale Anforderungen, welche an eine Software gestellt werden.
Jede Anforderung ist kurz zu erläutern.

3. Aufgabe

Beim Entwickeln einer Software unterscheidet man grundsätzlich zwischen 2 Strategien:

- Top-Down- Design
- Bottom-Up-Design

Erklären Sie den Unterschied zwischen beiden Strategien.

4. Aufgabe

Es existieren mehrere "klassische" Vorgehensmodelle, welche in der Softwareentwicklung eingesetzt werden. Eines dieser Modelle wird Wasserfallmodell genannt.

- a) Erläutern Sie das Wasserfallmodell.
- b) Nennen Sie 2 weitere "klassische" Vorgehensmodelle.

5. Aufgabe

In der Softwareentwicklung wird u. a. auch das Prinzip "Prototyping" genutzt.

- a) Erläutern Sie kurz dieses Prinzip. Beschreiben Sie einen entsprechenden Vorteil gegenüber dem Wasserfallmodell.
- b) Für das Entwicklungsmodell "Prototyping" existieren viele Varianten.
Erläutern Sie kurz die Variante "Evolutionäres Prototyping".

6. Aufgabe

Viele moderne Vorgehensmodelle zur Softwareentwicklung unterscheiden fünf grundlegende Phasen:

1. Analyse (Definition)
2. Entwurf (Design)
3. Implementierung
4. Test
5. Einsatz / Wartung

Erläutern Sie die o. g. Phasen. Welche Tätigkeiten sind in den jeweiligen Phasen durchzuführen?
Hinweis: Eine ausführliche Behandlung von Punkt 4 (Test) erfolgt zu einem späteren Zeitpunkt.

1. Aufgabe

Wenn eine Software erstellt wurde, so muss diese getestet werden.

- a) Erläutern Sie, was man unter einem Testfall zu verstehen hat. Welche Bestandteile sollte die Beschreibung eines Testfalls enthalten?
- b) Erläutern Sie die beiden Testmethoden White-Box-Test und Black-Box-Test. Welcher wesentliche Unterschied besteht zwischen beiden Testmethoden?

2. Aufgabe

Es wurde ein Programm erstellt, welches u. a. die Eingabe eines Jahres (Integer-Wert, z. B. 1945 oder 2022) erfordert. Nach Eingabe des Jahres muss dieses Programm prüfen, ob es sich um ein Schaltjahr handelt oder nicht. Die Prüfung soll ausschließlich auf der Grundlage des Gregorianischen Kalenders erfolgen.

Erstellen Sie eine Tabelle mit genau 7 Testwerten und dem zu erwartenden Ergebnis. Mit diesen Testwerten soll das Programm **bestmöglich** getestet werden können

Testwerte	Ergebnis (erwartet)
1945	kein Schaltjahr
...	...

Hinweis: Es soll nur die Prüfung bzgl. des Schaltjahres erfolgen. Das Verhalten bei Fehleingaben (z. B. "Meier" statt 1945) ist nicht zu testen.

3. Aufgabe**Anforderungsbeschreibung für ein Programm:**

Ein Online-Händler verlangt für Aufträge unter 50 € einen Versandkostenbetrag von 5 €. Ab 200 € wird versandkostenfrei geliefert. In allen übrigen Fällen betragen die Versandkosten 3 €. Ab einer Auftragssumme von 400 € gibt es 2% Rabatt.

Ein Programm soll nach Eingabe der Auftragssumme den Gesamtbetrag der Rechnung ausgeben.

Beispiel: Eingabe = 100 => Ausgabe = 103 (Versandkosten = 3, Rabatt = 0)

Aufgaben:

- a) Erstellen Sie ein Testprotokoll in tabellarischer Form: Geben Sie 6 einzugebende Werte an, mit denen das Programm bestmöglich getestet werden kann. Das Protokoll muss für jede Eingabe die jeweils zu erwartende Ausgabe enthalten.
- b) Sie haben das Programm als kompilierten Java-Byte-Code erhalten (Datei: `Programm1.class`). Gehen Sie auf der Kommandozeile in das Verzeichnis, in welchem Sie das Programm gespeichert haben und starten Sie es mit dieser Anweisung: `java Programm1`
Testen Sie das Programm unter Verwendung des in a) erstellten Protokolls.
Beurteilen Sie die Testergebnisse.
- c) **Zusatzaufgabe:** Versuchen Sie, die Ursachen der in b) ermittelten Fehler zu bestimmen. Dafür ist eventuell das Eingeben zusätzlicher Testwerte erforderlich.
- d) Erstellen Sie ein Struktogramm, welches zur Anforderungsbeschreibung passt.