

UML - SequenzDiagramme

(1.) Kopieren Sie das erste Beispiel in Dateien und lassen Sie es laufen! Zeichnen Sie das zugehörige Sequenzdiagramm aus dem Quellcode(evtl. rechte Seite der Folie abdecken!)

(2.) Gegeben ist nachstehender Quellcode. Zeichnen Sie jeweils das zugehörige Sequenzdiagramm **mit Stift und Papier**(alternativ UMLetino)!

(a.) Aufruf von f:

```
class B {
    void g() {
        h();
        h();
    }
    void h() {}
    void k() {}
}

class A {
    B b=new B();
    void f() {
        b.g();
        b.k();
    }
    void l() {
        b.g();
        b.k();
    }
}

public class TestAB {
    public static void main(String[] args) {
        A a = new A();
        a.f();
        a.l();
    }
}
```

(b.) Aufruf von f

```
class A {
    B b=new B();
    void f() {
        b.g();
        b.h();
    }
}

class B {
    C c=new C();
    void g() {
        c.n();
        h();
    }
    void h() {
        c.l();
    }
}

class C {
    void l() {
        m();
    }
    void m() {
    }
    void n() {}
}

public class TestAB {
    public static void main(String[] args) {
        A a = new A();
        a.f();
    }
}
```

(c.) Aufruf von f:

```
class B {  
    void g() {  
        for (int i =1;i<2 ;i++ )  
            h();  
    }  
    void h() {}  
}  
  
class A {  
    B b=new B();  
    void f() {  
        b.g();  
        b.h();  
    }  
}  
  
public class TestAB {  
    public static void main(String[] args) {  
        A a = new A();  
        a.f();  
    }  
}
```

(d.)

```
class B {  
    void g() {  
        h();  
    }  
    void h() {}  
}
```

```
class A {  
    B b=new B();  
    void f() {  
        int i=2;  
        if (i<3)  
            b.g();  
            b.h();  
    }  
}
```

```
public class TestAB {  
    public static void main(String[] args) {  
        A a = new A();  
        a.f();  
    }  
}
```

(e.) Aufruf von f:

```
class A {  
    B b=new B();  
    void f() {  
        b.g();  
        b.k();  
    }  
}  
class B {  
    void g() {  
        h();  
    }  
    void h() {}  
    void k() {}  
}
```

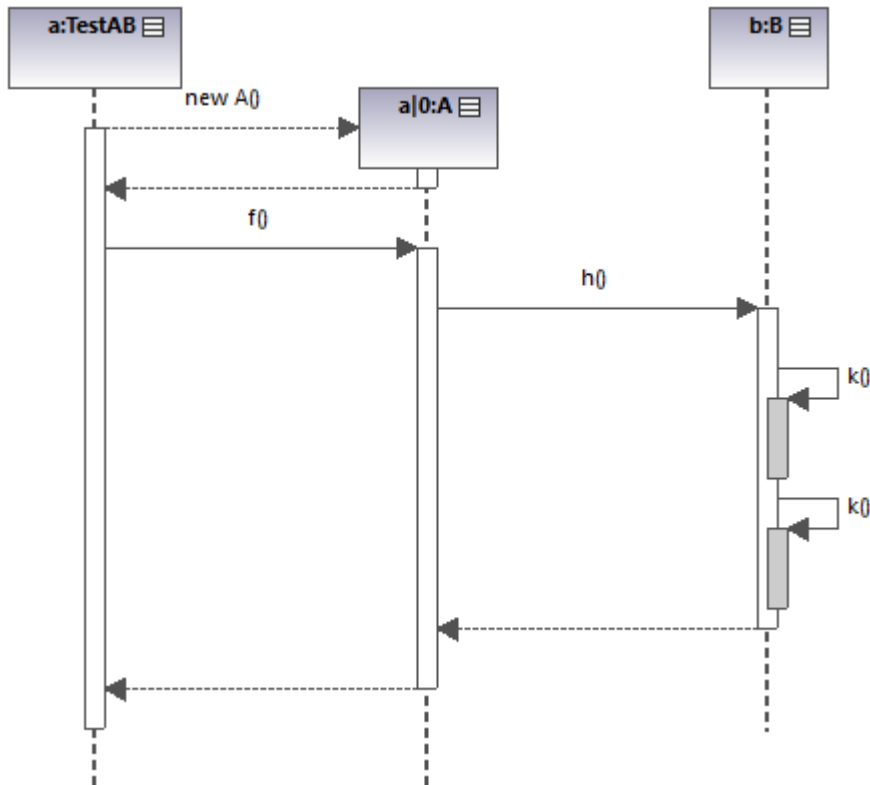
(f.) Aufruf von C6Prisma, Dateien im Vorlageordner!

(g.) Aufruf von f, Dateien im Vorlageordner!

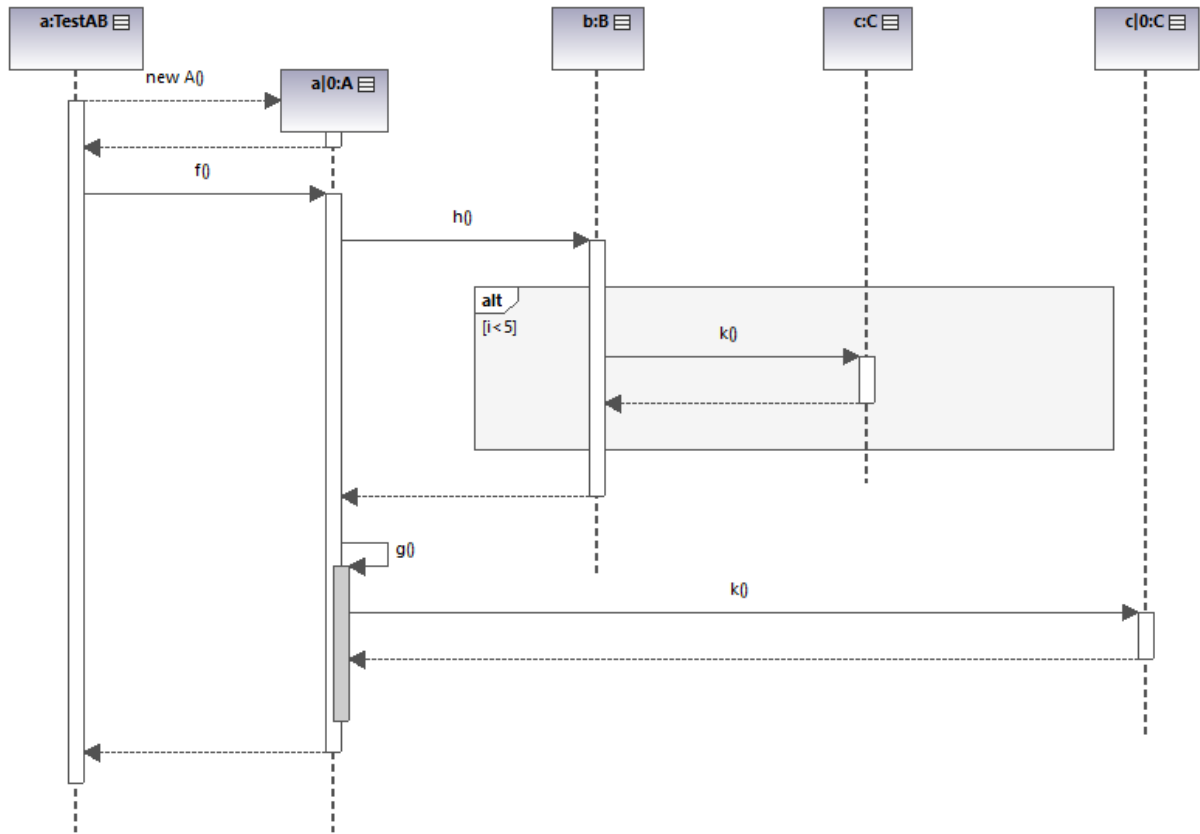
(3.) Generieren Sie den Quellcode, der zu nachfolgenden Sequenzdiagrammen gehört!
Das Hauptprogramm soll folgende Form haben:

```
public class TestAB {  
    public static void main(String[] args) {  
        A a = new A();  
        a.f();  
    }  
}
```

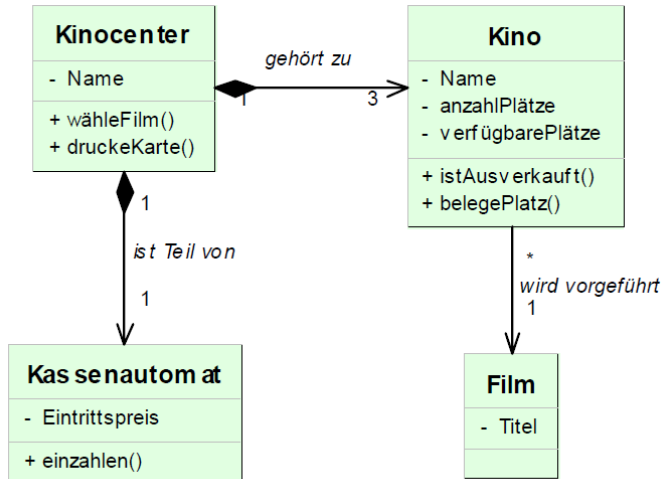
(a.)



(b.)



(4.) Problembeschreibung Eine Kinokette baut standardisierte Kinocenter. Das Auswählen eines Filmes, Zahlen des Eintrittspreises und Drucken der Eintrittskarte sollen mit einer entsprechenden Software vollständig automatisiert werden. Aus folgender Situationsbeschreibung ist in einem ersten Analyseschritt untenstehendes Klassendiagramm erarbeitet worden: Ein Kinobesucher wählt einen Filmtitel aus. Das Kinocenter überprüft das zugehörige Kino auf freie Plätze: Wenn das Kino noch nicht ausverkauft ist, erhält der Kinobesucher die Antwort, dass der gewählte Film möglich ist. Nun erfolgt das Einzahlen des Eintrittspreises am Kassenautomat. Nachdem der Kassenautomat die korrekte Zahlung des Eintrittspreises festgestellt hat, meldet er dies. Daraufhin kann der Besucher vom Kinocenter eine Karte anfordern. Das Kinocenter belegt im zugehörigen Kino einen Platz und druckt die Eintrittskarte.

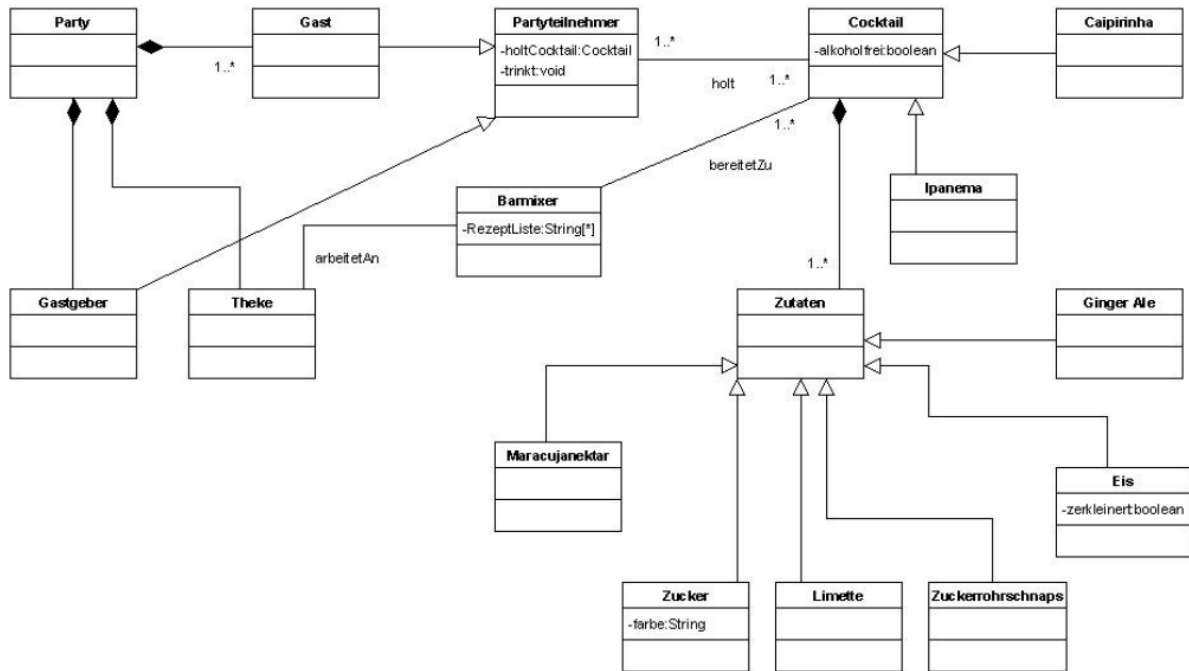


- (a.) Welche Aussagen lassen sich zu dem gegebenen Klassendiagramm machen (Stichworte: Klasse, Attribut, Methode, Beziehung, Kardinalität)?
- (b.) Ergänzen Sie das Klassendiagramm um geeignete Datentypen, notwendige Methoden-Parameter und erforderliche Rückgabewerte. Zeichnen Sie das Klassendiagramm und begründen Sie ihre Entscheidungen.
- (c.) Objektdiagramm Stellen Sie die folgende Situation in der Form eines Objektdiagramms dar: Das Kinocenter „Kinopolis“ umfasst die Kinos „Gloria“, „Heli“ und „Capitol“. Alle Kinos verfügen über 280 Plätze. Der Eintrittspreis für die Kinos beträgt einheitlich 7,50 E. Im Gloria wird der Film „Casablanca“, im Heli „High Noon“ und im Capitol „Capote“ gezeigt.
- (d.) Stellen Sie folgenden Anwendungsfall in einem Sequenzdiagramm dar: Ein Kinobesucher wählt den Film „Casablanca“. Da das Kino noch nicht ausverkauft ist, erhält der Besucher nach korrekter Zahlung des Eintrittspreises eine Eintrittskarte. Arbeitshinweis: Beachten Sie die Situationsbeschreibung und das Klassendiagramm!

(5.) Um ein Auto (vorzugsweise einen Daimler E 220 CDI T – Modell) anzulassen bedarf es der folgenden Vorgänge:

Der Schlüssel wird in das Schloss eingesteckt. Hierauf teilt die Elektronik der Anzeige mit, dass ein Fahrer im Auto sitzt, der im Display alle wichtigen Informationen angezeigt bekommt. Der Fahrer dreht den Schlüssel bis zur Einraststellung 2. Die Elektronik bemerkt dies und teilt der Motorsteuerung mit, dass die Glühkerzen im Motor vorgeglüht werden können. Besitzen die Glühkerzen ausreichend Temperatur, so teilt der Motor dies der Motorsteuerung mit. Diese gibt die Information an die Elektronik weiter, welche daraufhin im Display den Fahrer informiert. Der Fahrer dreht den Schlüssel in Einraststellung 3, was die Elektronik bemerkt und dem Motor über die Motorsteuerung das Signal zum Starten schickt. Der Motor startet, teilt dies der Elektronik mit, welche im Display den Fahrer über alle wichtigen Informationen während der Fahrt informiert. Der Fahrer bewegt den Schlüssel in Einraststellung 2. Anschließend fährt er los. Erstellen Sie ein Sequenzdiagramm für die obige Situationsbeschreibung!

(6.) Gegeben ist die folgende Situation eines Restaurants durch ein Klassendiagramm: Gegeben ist das folgende Klassendiagramm:



Erstellen Sie ein Sequenzdiagramm für den Vorgang ab dem Moment, da der Gast sein Essen bestellt, bis zu dem Moment bis er sein Essen serviert bekommt! Ein Gast möchte sich etwas zu trinken holen. Hierzu geht er an die Theke und bittet den Barkeeper einen Cocktail zu mixen. Der Barkeeper fragt zurück, ob der Cocktail Alkohol enthalten soll oder nicht. Der Gast antwortet. Der Barkeeper bereitet den Cocktail zu. Der Gast nimmt diesen Cocktail und trinkt ihn aus.

Erstellen Sie zu diesem Ablauf das entsprechende Sequenzdiagramm!

(7.) Gegeben ist der Quellcode für eine Autoverwaltung im Ordner 05 – Vorlagen
Erstellen Sie ein Sequenzdiagramm für die Methode buildCar!

(8.) Sie haben den Auftrag, eine Online-Videothek zu realisieren. Sie haben dazu folgende Angaben erhalten:

- Die Videothek unterstützt das „Ausleihen“ von Filmen für registrierte Kunden. Dazu müssen Kunden sich zunächst mit ihrer Kundennummer und ihrem Passwort anmelden.
- Kunden werden zusammen mit ihrem Guthaben verwaltet.
- Filme besitzen einen individuellen Namen und Preis.
- Ein Film wird über einen Streaming-Server bereitgestellt. Der Server kann hierzu einen film- und kundenspezifischen Link generieren.

(a.) Modellieren Sie diesen Sachverhalt anhand eines Klassendiagramms. Wählen Sie sinnvolle Operationen (mit möglichst vollständigen Signaturen) und Attribute für Ihre Klassen. Ergänzen Sie die Klassen um sinnvolle Beziehungen und deren Kardinalitäten.

(b.) Modellieren Sie für die Online-Videothek (siehe Aufgabe 1) die Film Ausleihen Funktion. Erstellen Sie dazu ein Sequenzdiagramm für folgenden Ablauf der Ausleihe:

- Die Videothek berechnet zuerst, ob das Guthaben des Kunden reicht um den Film zu bezahlen.
- Reicht das Guthaben nicht aus, wird stattdessen eine Anforderung zum Auffüllen des Guthabens angezeigt.
- Falls das aktuelle Guthaben des Mitglieds ausreicht, veranlasst die Videothek einen StreamingServer einen Link für den Film zu generieren.
- Die Videothek zeigt dem Benutzer den Link an, unter dem der Film zugreifbar ist.

Gehen Sie davon aus, dass sich das Mitglied bereits auf der Seite des gewünschten Films befindet.

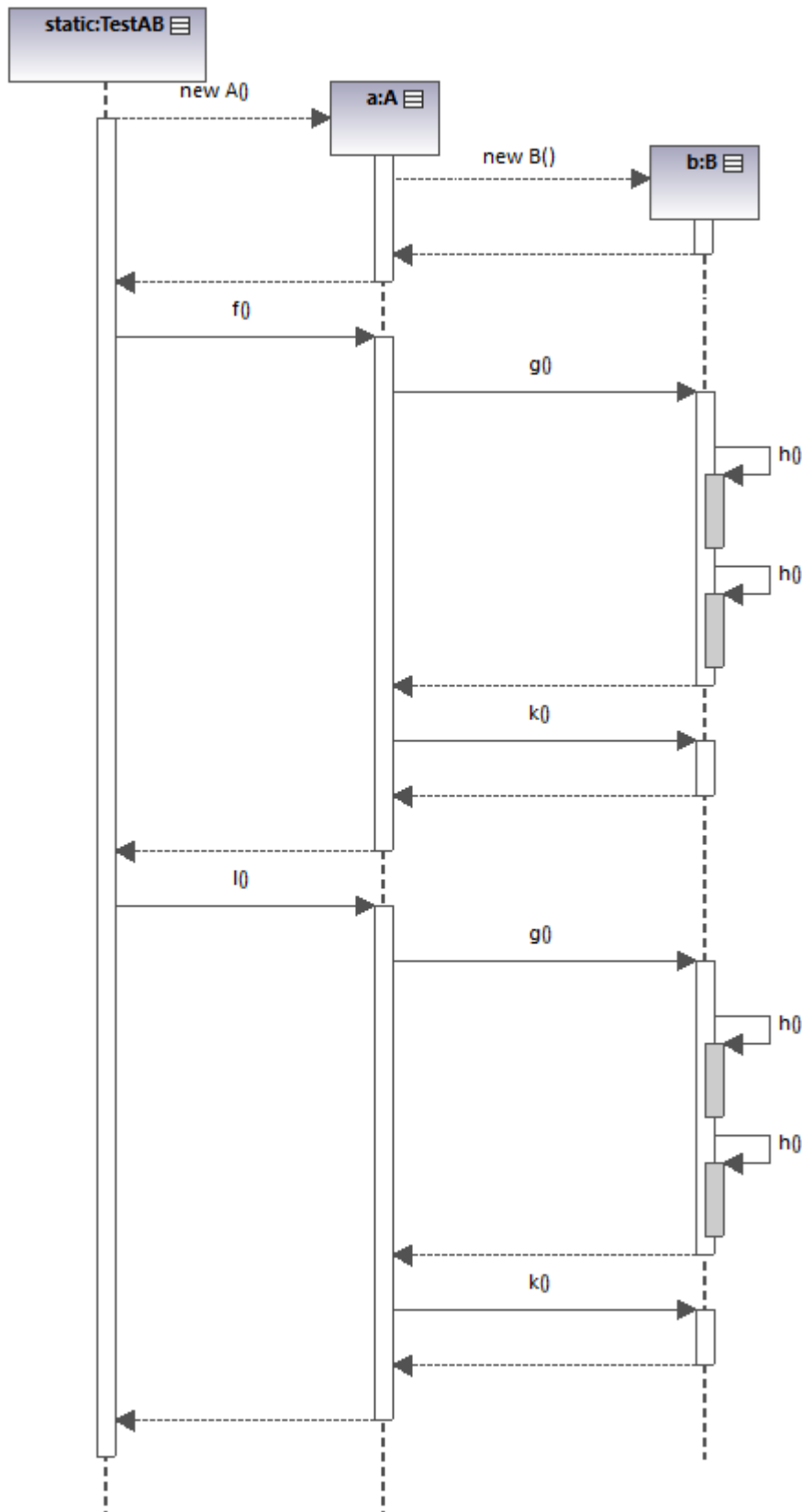
Wählen Sie geeignete Namen für die Elemente Ihres Diagramms.

Wichtig: Achten Sie darauf, dass Ihre Diagramme aus Aufgabe 1 und 2 konsistent sind. Nutzen Sie im Sequenzdiagramm nur Klassen, Operationen, etc. die im Klassendiagramm aus Aufgabe 1 enthalten sind. Falls Sie im Sequenzdiagramm zusätzliche Operationen, Parameter, etc. brauchen, ergänzen Sie das Klassendiagramm entsprechend!

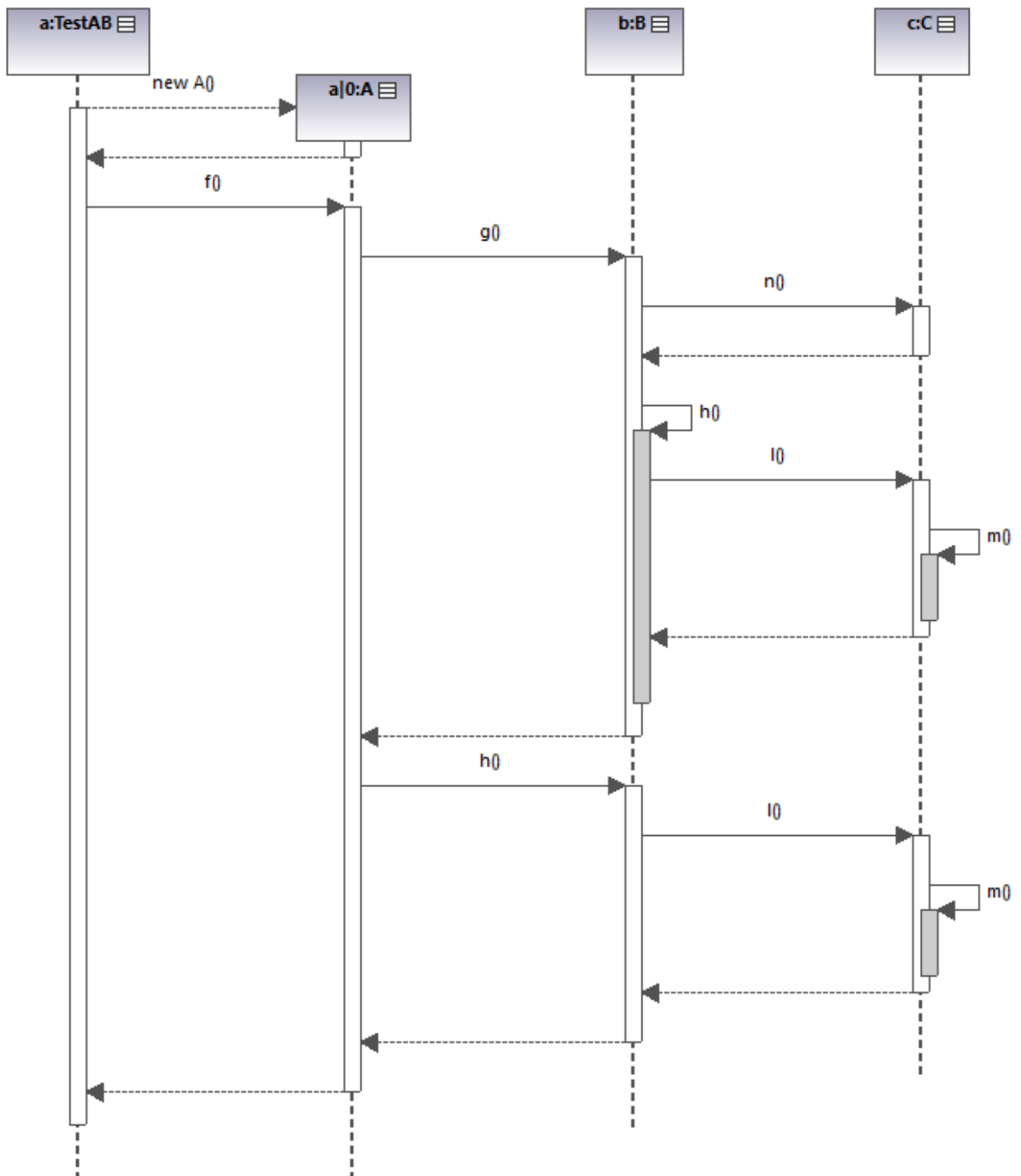
UML - Sequenzdiagramme - Lösungen

(1.) entfällt

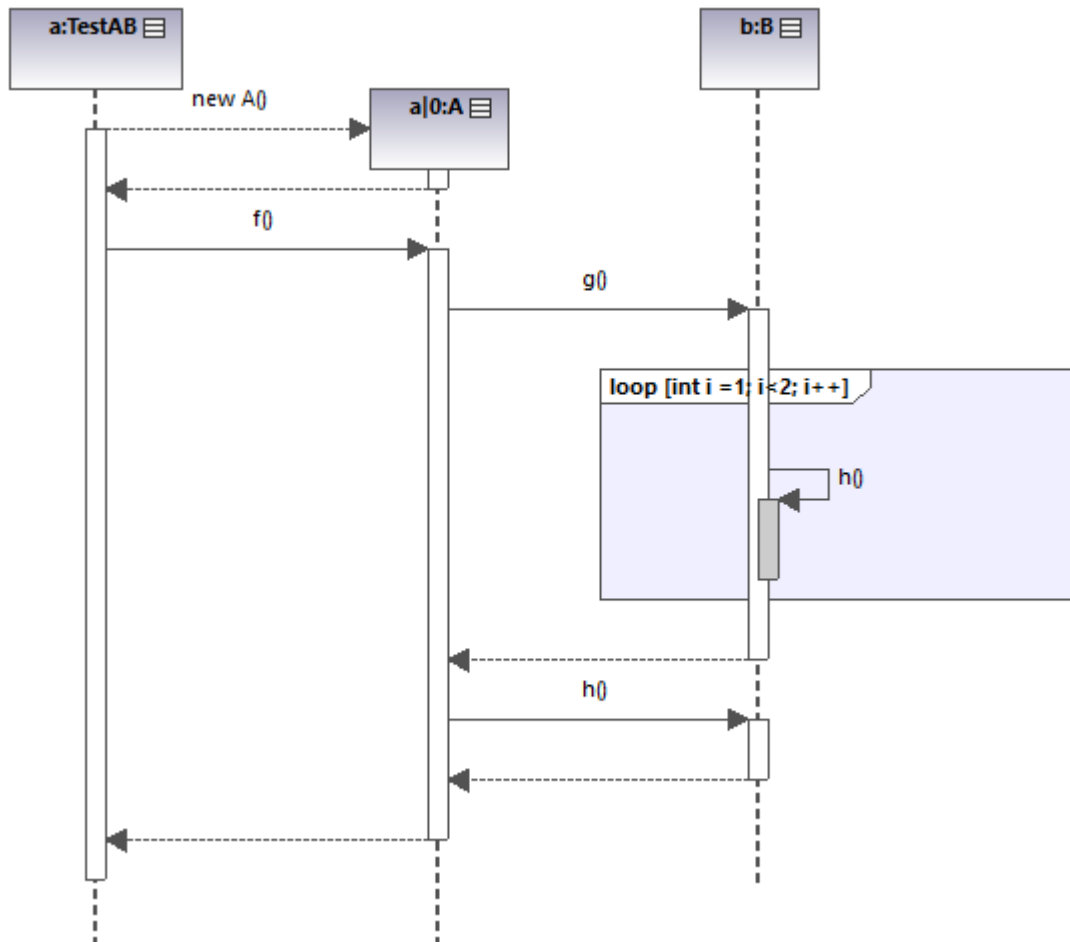
(2.) (a.)



(b.)

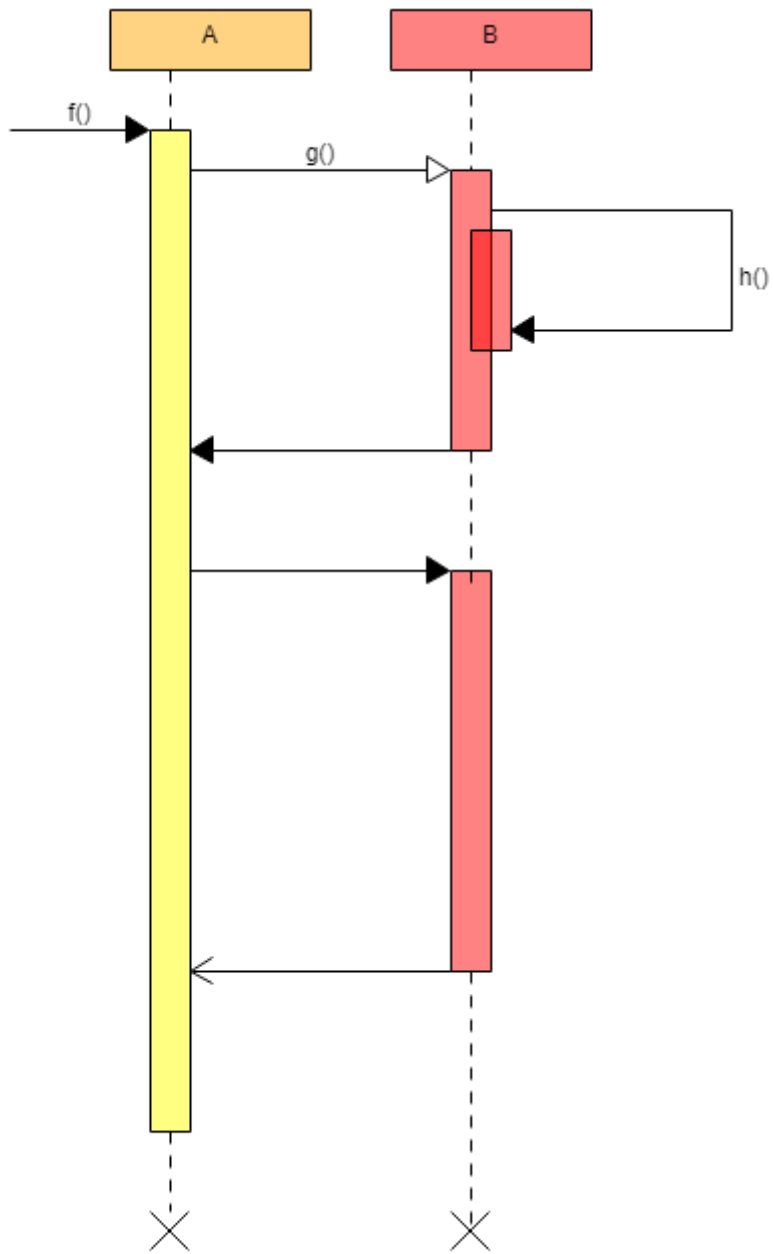


(c.)

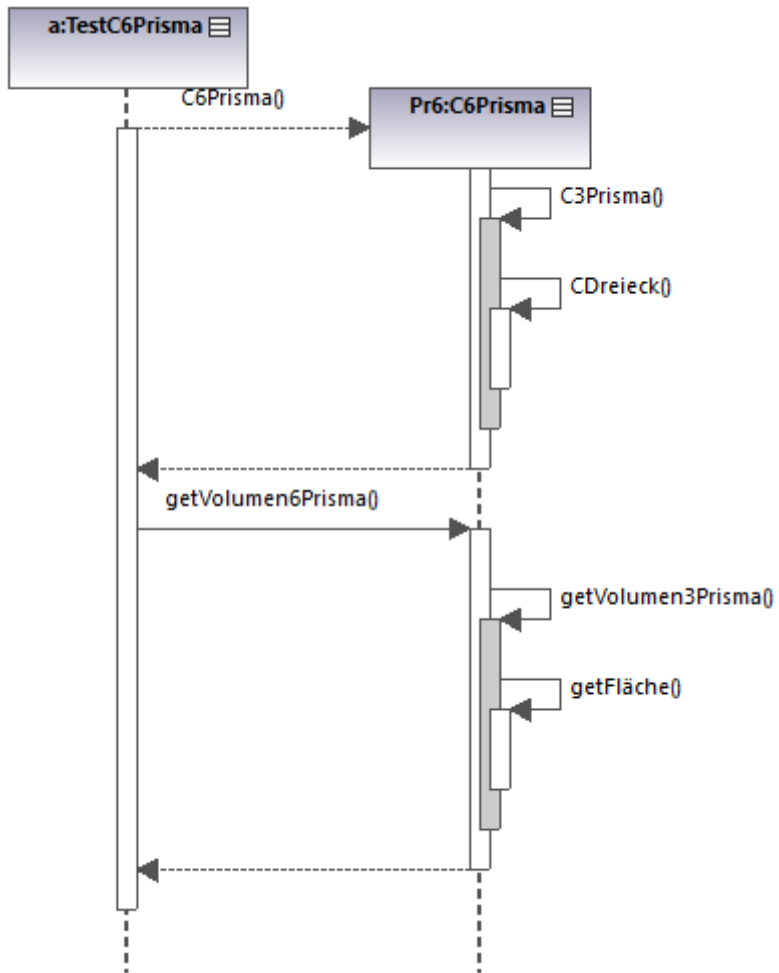


(d.)

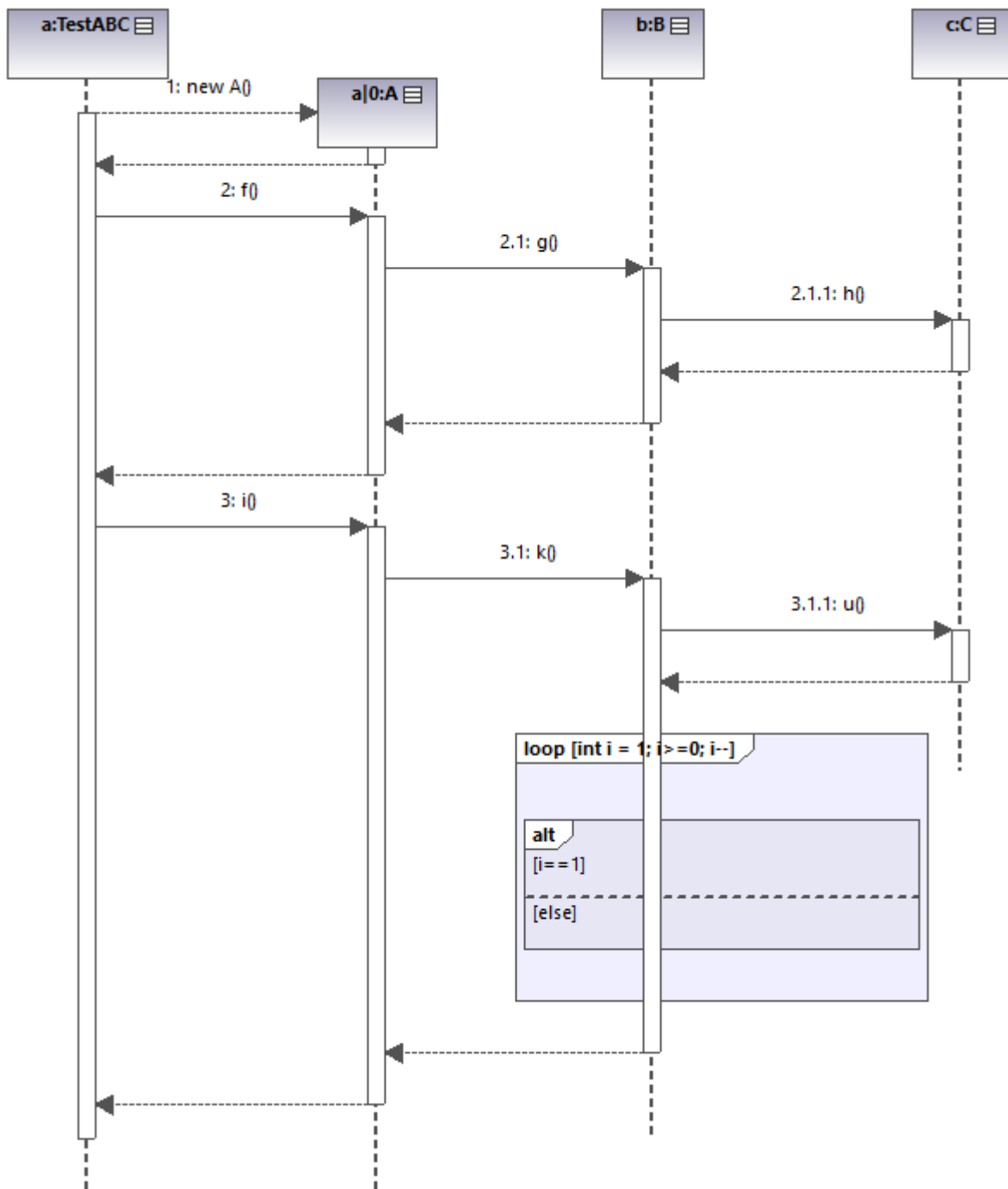
(e.)



(f.)



(g.)



(3.)

(a.)

```
class A {
    B b=new B();
    void f() {
        b.h();
    }
    void g() {
        b.k();
    }
}
```

```
class B {
    void h() {
        k();
        k();
    }
    void k() {}
}
```

(b.)

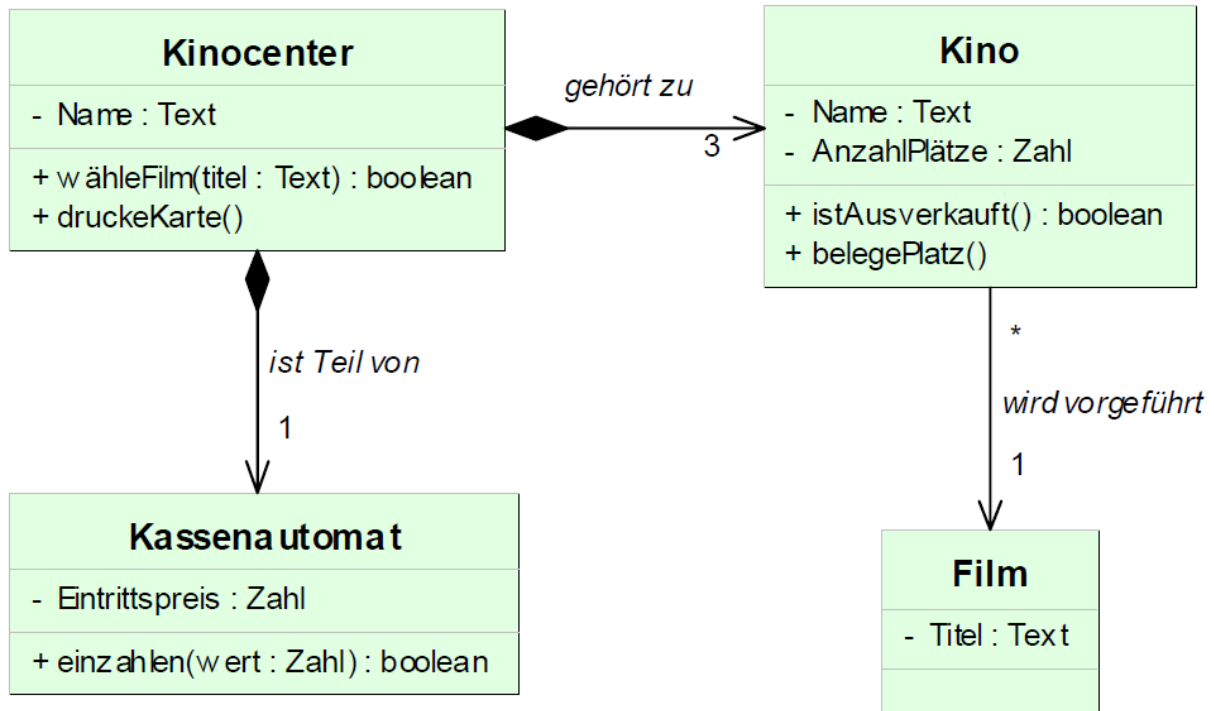
```
class A {
    B b=new B();
    C c=new C();
    void f() {
        b.h();
        g();
    }
    void g() {
        c.k();
    }
}
```

```
class B {
    C c =new C();
    void h() {
        int i=3;
        if (i<5)
            c.k();
    }
}
```

```
class C {
    void k() {
    }
}
```

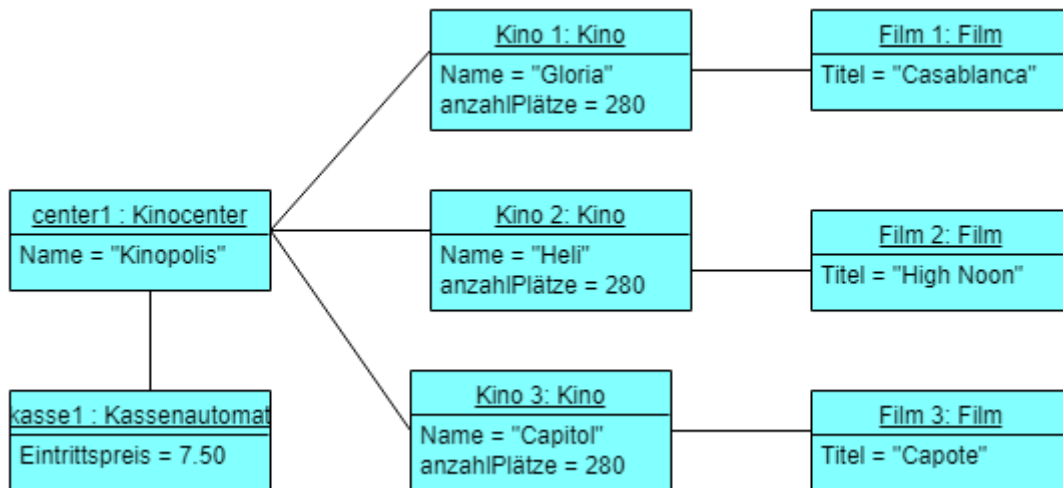
(4.) (a.) Von den Klassen *Kassenautomat*, *Film*, *Kino* und *Kinocenter* können Objekte gebildet werden. Auf die Attribute (*Name*, *AnzahlPlätze*, *Titel*, *Eintrittspreis*) kann nicht von außerhalb der Klasse zugegriffen werden, da sie mit dem Zugriffsmodifizierer "Private" gekennzeichnet sind. Alle Methoden (*wähleFilm()*, *chuckeKarte()*, *istAusverkauft()*, *belegePlatz()*, *einzahlen()*) sind öffentlich (Zugriffsmodifizierer „Public“), d.h. auf sie kann von außerhalb der Klasse zugegriffen werden. Zwischen der Klasse *Kinocenter* und den Klassen *Kassenautomat* bzw. *Kino* besteht eine Komposition. Diese Klassen sind Teile des *Kinocenters* und von diesem existenzabhängig. Die Beziehungen sind gerichtet, d.h. der Zugriff ist nur in Pfeilrichtung. Jedes *Kinocenter* hat genau einen *Kassenautomaten* und 3 *Kinos*. In jedem *Kino* wird genau ein *Film* vorgeführt, ein *Film* kann in beliebig vielen *Kinos* vorgeführt werden.

(b.)

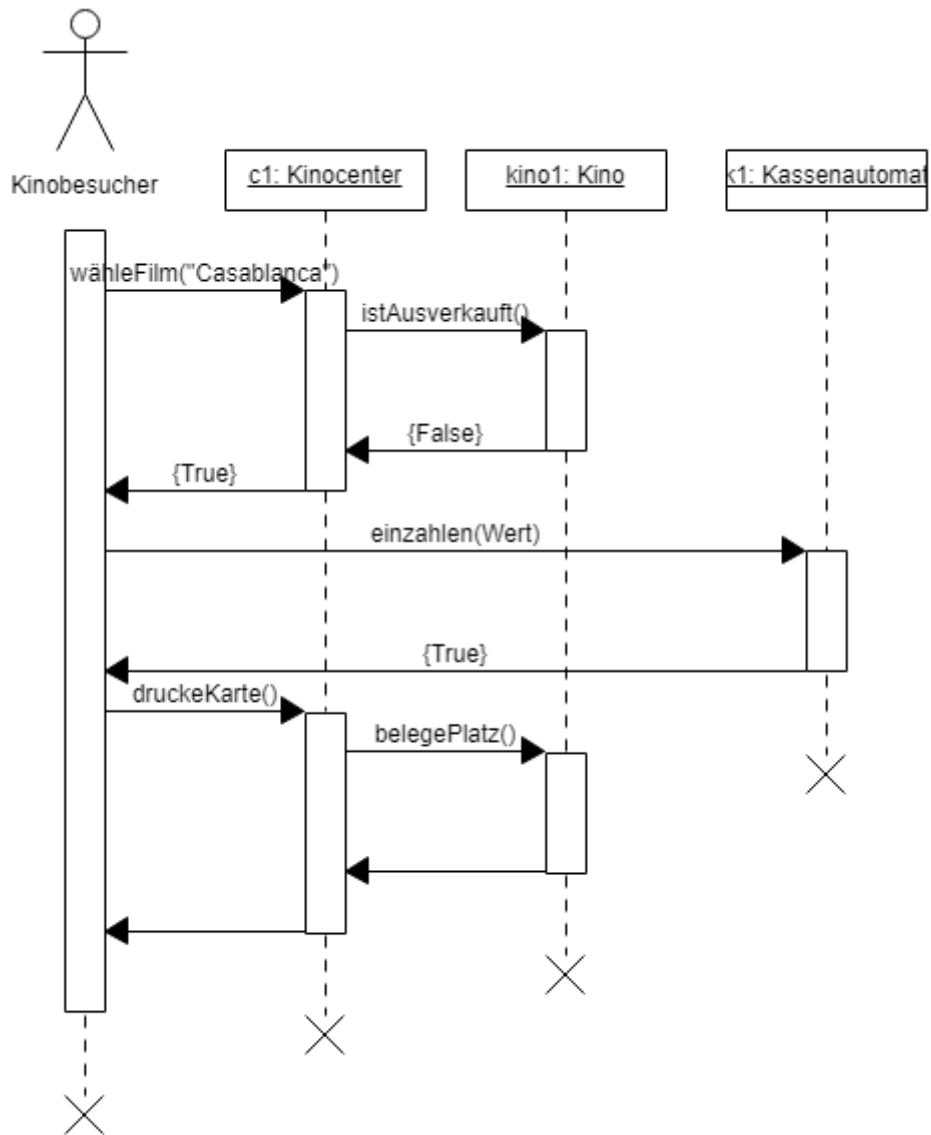


Die Attribute „Name“ und „Titel“ erhalten den Datentyp „Text“; „AnzahlPlätze“ und „Eintrittspreis“ den Datentyp „Zahl“. Die Methoden „druckeKarte()“ und „belegePlatz()“ benötigen weder Parameter noch Rückgabewerte, da alle benötigten Daten von den Klassen abgefragt werden können und keine Informationen an das Anwendungsprogramm geliefert werden müssen. Der Methode „wähleFilm()“ wird als Parameter der gewählte Filmtitel übergeben, sie liefert als Antwort „True“, wenn im Kino ein Platz frei ist. Der Methode *einzahlen()* wird der eingezahlte Betrag als Wert übergeben und sie liefert als Antwort „True“, wenn der Eintrittspreis eingezahlt wurde.

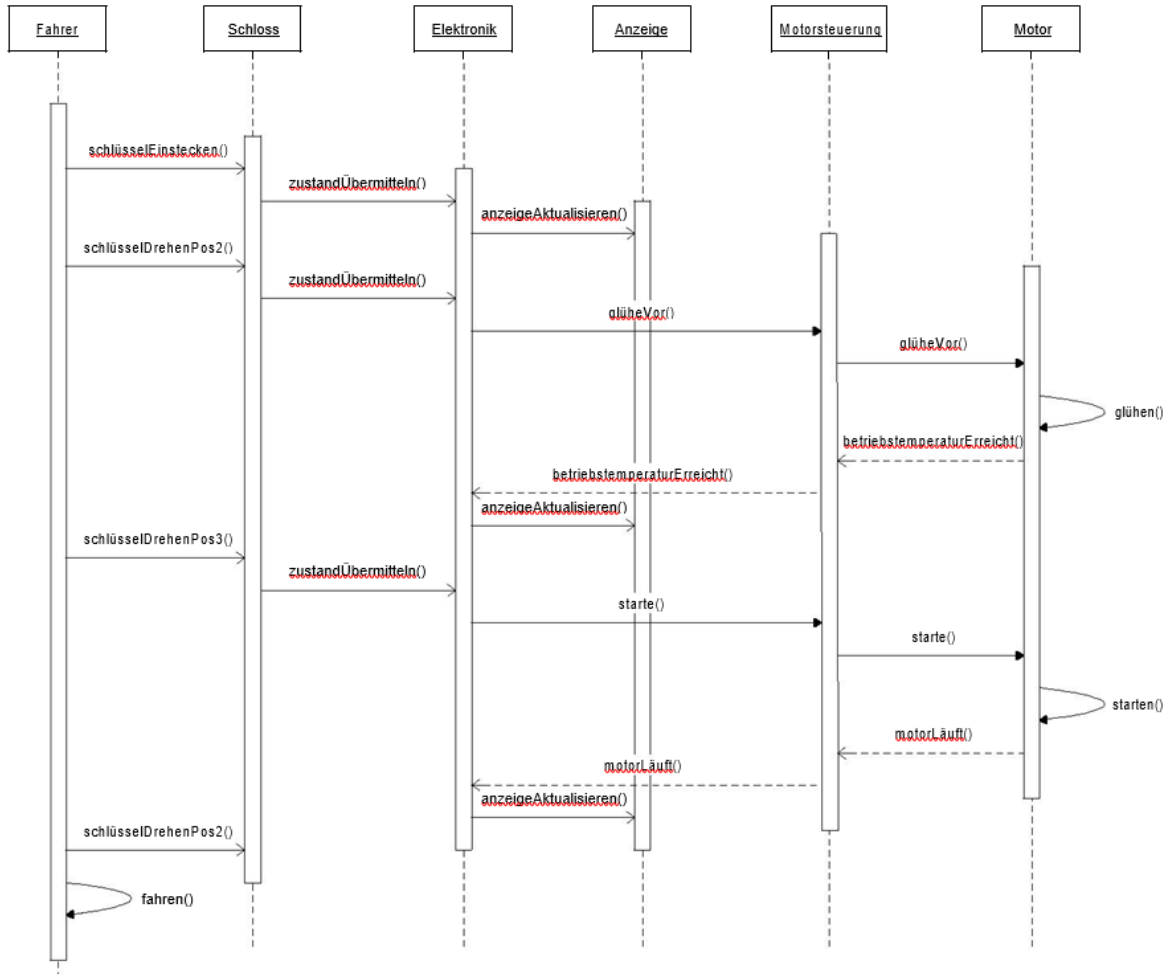
(c.)



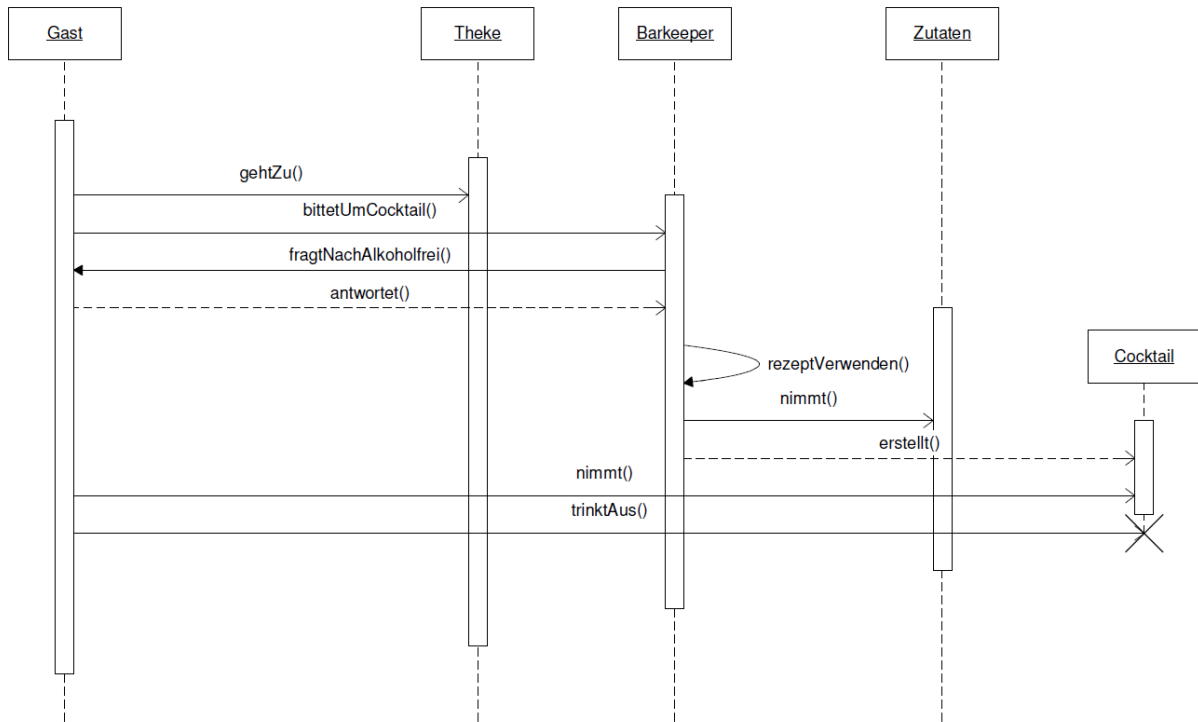
(d.)



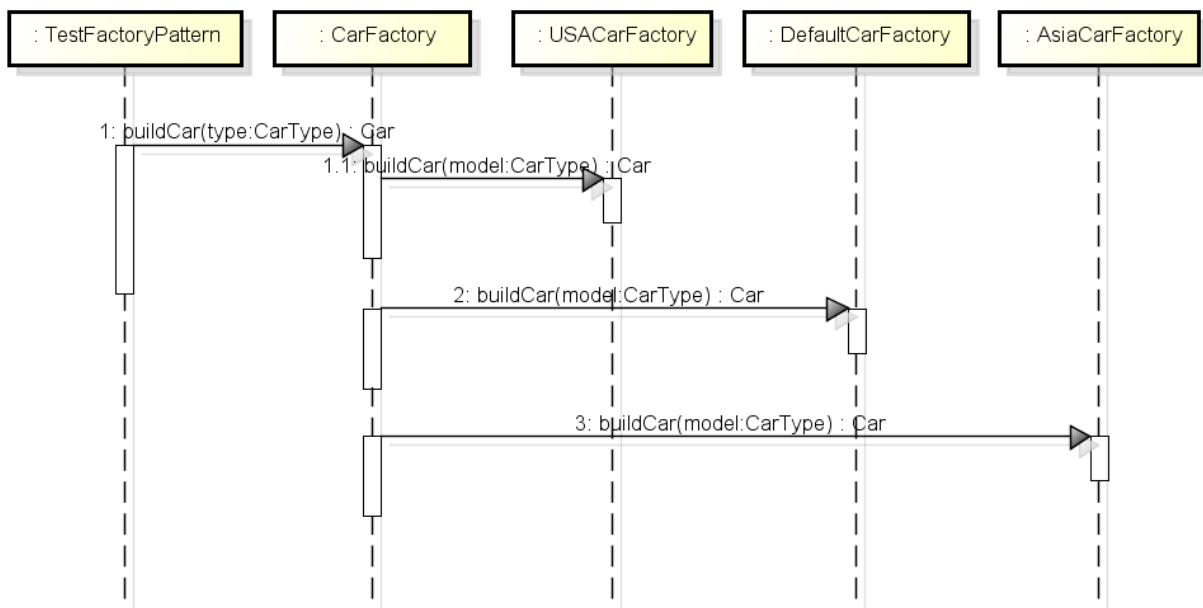
(5.)



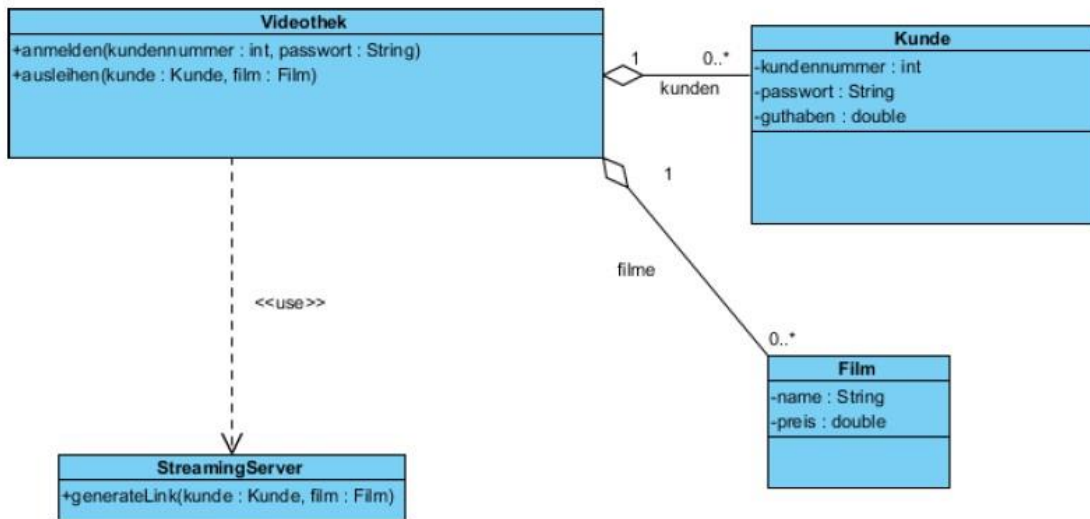
(6.)



(7.)



(8.) (a.)



(b.)

