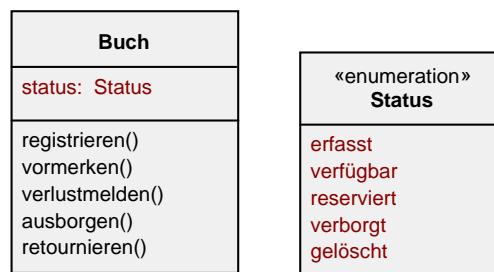


Aufgabe 1: Buch einer Bibliothek

Modellieren Sie die Zustände eines Buches in einer Bibliothek mittels UML Zustandsdiagramm.

Jedes Buch ist zu Beginn neu. Damit das Buch im System vorhanden ist, muss es erst einmal erfasst werden. Ist dies passiert, ist das Buch im System vorhanden und kann verliehen oder reserviert werden. Wurde ein Buch reserviert aber innerhalb von einer Woche nicht abgeholt, so verfällt die Reservierung und jemand anders kann es ausborgen. Wenn jemand den Verlust eines ausgeborgten Buches meldet, wird der Status des Buches auf gelöscht gesetzt. Um das Beispiel einfach zu halten nehmen wir an, dass lediglich verfügbare Bücher reserviert werden können (d.h. verborgte Bücher können nicht reserviert werden).

Gegeben ist folgender Ausschnitt eines Klassendiagramms der sämtliche Methoden der Klasse *Buch* zeigt. Weiters besitzt ein Buch einen Status – die einzelnen möglichen Stati sind ebenfalls abgebildet:



Erstellen Sie anhand des Klassendiagramms ein entsprechendes Zustandsdiagramm, das die verschiedenen Zustände und Zustandsübergänge eines Buches wiedergibt. Modellieren Sie die **Statusänderungen** des Buches explizit im Zustandsdiagramm.

Aufgabe 2: Fahrkartenautomat

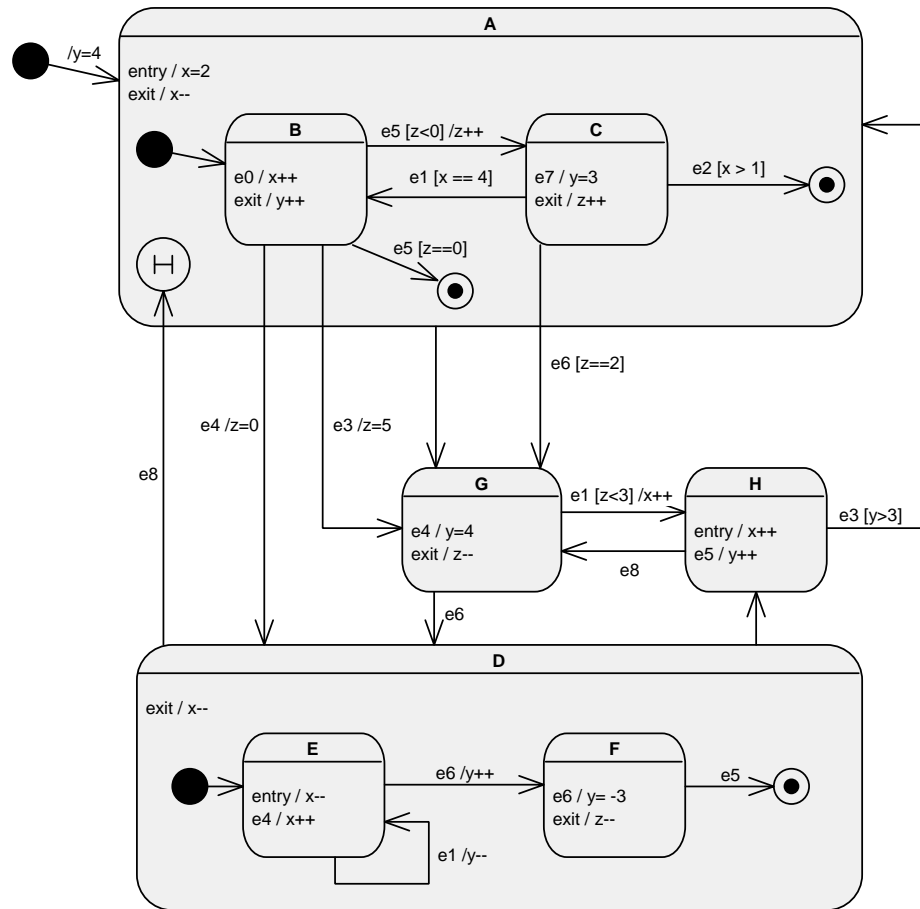
Modellieren Sie ein UML Zustandsdiagramm, das die Zustände eines Fahrkartenautomats (z.B. in einer Straßenbahn) aus der Sicht des Automaten abbildet.

Es stehen folgende Informationen zur Verfügung:

Zu Beginn wartet der Fahrkartenautomat auf eine neue Eingabe. Der Kunde wählt die gewünschte Karte aus und bekommt den offenen Betrag angezeigt. Nun wirft der Kunde solange Münzen in den Automaten ein, bis der offene Betrag erreicht oder überschritten wurde. Hat der Kunde mehr Münzen eingeworfen als erforderlich, bekommt er das Wechselgeld retour und anschließend kann er die gedruckte Karte entnehmen. Hat der Kunde genau den offenen Betrag bezahlt, wird die Karte sofort gedruckt. Damit ist der Kartenkauf beendet und der Automat für einen neuen Kartenkauf bereit. Hat der Kunde den offenen Betrag noch nicht vollständig eingeworfen, kann er jederzeit die Storno-Taste betätigen. Dies führt zum Abbruch des Kartenkaufs, der bisher eingeworfene Betrag wird ausgegeben und der Automat wartet auf den nächsten Kunden. Zu einem entsprechenden Abbruch kommt es auch, wenn der Kunde 3 Minuten lang keine neue Münze einwirft.

Aufgabe 3: Ereignisfolge 1

Gegeben ist das folgende Zustandsdiagramm:



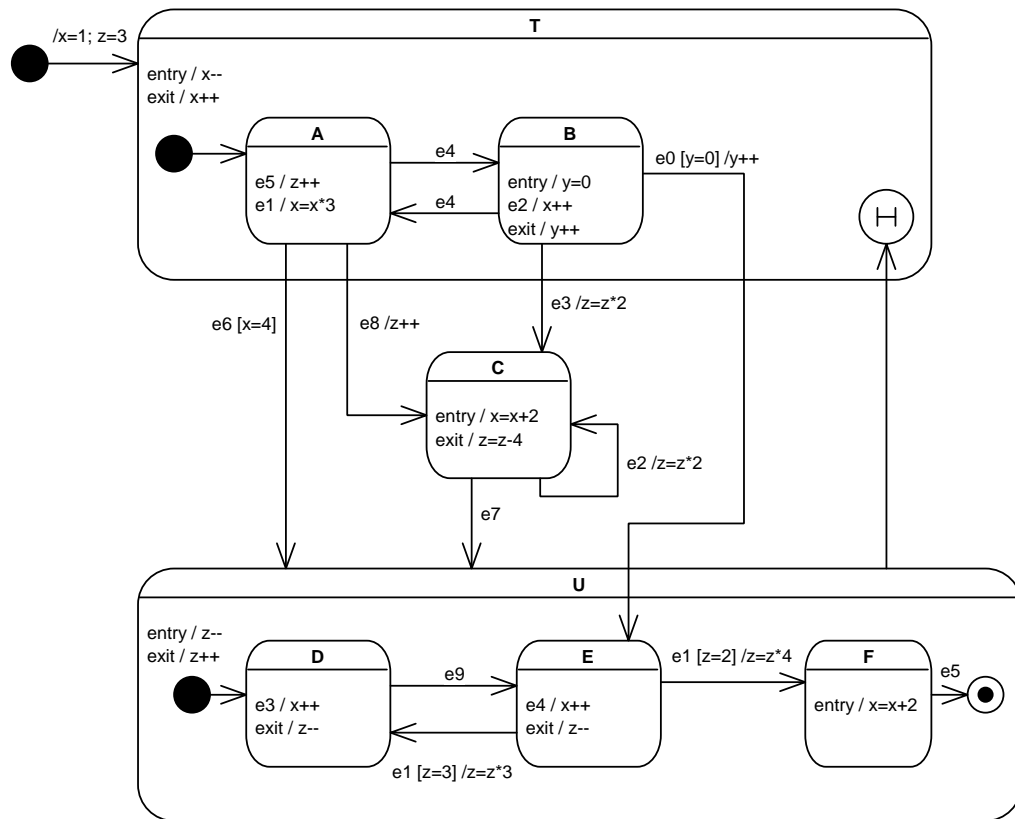
Vervollständigen Sie die folgende Tabelle, um zu veranschaulichen, welche Zustände und Aktionen bei der folgenden Ereignisfolge vorkommen.

Belegung der Variablen

Ereignis	Eingetr. Zustand	x	y	z
<i>Beginn</i>				
e4				
e6				
e8				
e5				
e2				
e4				
e1				
e8				
e6				

Aufgabe 4: Ereignisfolge 2

Gegeben ist das folgende Zustandsdiagramm:



Vervollständigen Sie die folgende Tabelle, um zu veranschaulichen, welche Zustände und Aktionen bei der folgenden Ereignisfolge vorkommen.

Belegung der Variablen

Ereignis	Eingetr. Zustand	x	y	z
<i>Beginn</i>				
e4				
e0				
e1				
e5				
e3				
e2				
e7				
e3				
e9				
e5				

Aufgabe 5: Ampel

Modellieren Sie ein Zustandsdiagramm für eine Ampelanlage:

Prinzipiell kann sich entweder die Nord-Süd Richtung oder die West-Ost Richtung auf GO befinden, während sich die jeweils andere Richtung dann in STOP befindet. Eine GO Phase dauert 30 Sekunden. Zwischen jedem Wechsel von GO und STOP wird für beide Richtungen eine Gelbphase für 5 Sekunden geschaltet (dabei sind alle Autoampel auf Gelb und alle Fußgängerampeln auf Rot). Innerhalb der GO Phase ist die Autoampel immer Grün, die Fußgängerampel jedoch nur 20 Sekunden Grün und dann Rot. Für die letzten 5 Sekunden wo Autoampel aber auch Fußgängerampel Grün sind, blinken diese. Während der STOP Phase sind sowohl Autoampel als auch Fußgängerampel auf Rot.

Modellieren Sie die Ampelanlage mit Hilfe einer UND-Verfeinerung um die Zustände der verschiedenen Ampeln darstellen zu können.